ARM
L220 Level 2 Cache Controller (AC131)
Errata Notice

This document contains all errata known at the date of issue in releases up to and including revision r1p7 of L220

© Copyright ARM Limited 2004-2009. All rights reserved.

Proprietary notice

Words and logos marked with ® or ™ are registered trademarks or trademarks of ARM Limited in the EU and other countries, except as otherwise stated below in this proprietary notice. Other brands and names mentioned herein may be the trademarks of their respective owners.

Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder.

The product described in this document is subject to continuous developments and improvements. All particulars of the product and its use contained in this document are given by ARM Limited in good faith. However, all warranties implied or expressed, including but not limited to implied warranties of merchantability, or fitness for purpose, are excluded.

This document is intended only to assist the reader in the use of the product. ARM Limited shall not be liable for any loss or damage arising from the use of any information in this document, or any error or omission in such information, or any incorrect use of the product.

Document confidentiality status

This document is Non Confidential.

Web address

http://www.arm.com/

© Copyright ARM Limited 2004-2009. All rights reserved.

Feedback on the product

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

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

Feedback on this document

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

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

General suggestion for additions and improvements are also welcome.

© Copyright ARM Limited 2004-2009. All rights reserved.

Contents

© Copyright ARM Limited 2004-2009. All rights reserved.

© Copyright ARM Limited 2004-2009. All rights reserved.

Introduction

Scope

This document describes errata categorised 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

Categorisation of Errata

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

Category 1 Behavior that is impossible to work around and that severely restricts the use of the product in all, or the majority of applications, rendering the device unusable.

Category 2 Behavior that contravenes the specified behavior and that might limit or severely impair the intended use of specified features, but does not render the product unusable in all or the majority of applications.

Category 3 Behavior that was not the originally intended behavior but should not cause any problems in applications.

Change Control

9 Mar 2009: Changes in Document v19

Page	Status	ID	Cat	Summary
38	Updated	484863	Cat 2	The Cache Sync operation does not guarantee that the Eviction Buffer is empty

14 May 2008: Changes in Document v18

Page	Status	ID	Cat	Summary
36	Updated	425331	Cat 2	Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations received
38	Updated	484863	Cat 2	The Cache Sync operation does not guarantee that the Eviction Buffer is empty
46	New	533267	Cat 3	Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations received

11 Dec 2007: Changes in Document v17

Page	Status	ID	Cat	Summary
38	New	484863	Cat 2	The Cache Sync operation does not guarantee that the Eviction Buffer is empty

02 Mar 2007: Changes in Document v16

Page	Status	ID	Cat	Summary
36	New	425331	Cat 2	Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations received
43	New	424665	Cat 3	ARSIDEBANDMx tied to 5’b00000 when L2 cache is enabled
44	New	425600	Cat 3	BRESPP may be incorrect on the first write access with WSTRBP != 4’b1111 received by the Peripheral port after reset

30 Aug 2006: Changes in Document v15

Page	Status	ID	Cat	Summary
18	Updated	401552	Cat 1	“mbist_ce_reg” and “mbist_we_reg” need to be reset

24 Aug 2006: Changes in Document v14

Page	Status	ID	Cat	Summary
18	New	401552	Cat 1	“mbist_ce_reg” and “mbist_we_reg” need to be reset

2 Aug 2006: Changes in Document v13

Page	Status	ID	Cat	Summary
35	New	400021	Cat 2	L220 AXI master interfaces do not maintain ARID or AWID during a locked sequence
47	Updated		Imp	Updated chapter to point to 391291
49	Updated	391291	Doc	TRM must precisely document the clock enable usage model implemented in the L220 AXI slave and master interfaces

© Copyright ARM Limited 2004-2009. All rights reserved.

17 Jul 2006: Changes in Document v12

Page	Status	ID	Cat	Summary
33	New	397385	Cat 2	In exclusive cache configuration, inner cacheable writeback data reads that are not linefills can deadlock or return incorrect data
34	New	397533	Cat 2	Data parity errors may not be reported with the first read data of a burst

20 June 2006: Changes in Document v11

Page	Status	ID	Cat	Summary

24 May 2006: Changes in Document v10

Page	Status	ID	Cat	Summary
41	New	391290	Cat 3	The L220 does not make correct distinction between data and instructions for cache lockdown
47	Updated		Imp	Updated chapter to point to 391291 and the "Clock Enable Usage Model in the L220 AXI interfaces" EAN
49	New	391291	Doc	TRM must precisely document the clock enable usage model implemented in the L220 AXI slave and master interfaces

04 May 2006: Changes in Document v9

Page	Status	ID	Cat	Summary
47	New	382986	Imp	Incorrect connections within the Peripheral port IEM register slice

04 Apr 2006: Changes in Document v8

Page	Status	ID	Cat	Summary
23	Updated	351241	Cat 2	L220 RTL problem with multi-cycle path into AXI master interface
32	New	379681	Cat 2	Aborted linefills are incorrectly allocated into the L2 cache

01 Mar 2006: Changes in Document v7

Page	Status	ID	Cat	Summary

03 Feb 2006: Changes in Document v6

Page	Status	ID	Cat	Summary
27	Updated	367118	Cat 2	A clean eviction received by L220 can incorrectly mark dirty data as clean
29	New	372114	Cat 2	Some imprecise error responses from L3 memory are not reported through interrupts
30	New	373382	Cat 2	Starting a background clean or clean-and-invalidate maintenance operation can prevent the allocation of a cache line
48	New	373383	Doc	TRM must state that any write to registers of ARM L220 must be preceded by an explicit cache-sync operation

06 Dec 2005: Changes in Document v5

Page	Status	ID	Cat	Summary
25	New	364369	Cat 2	L220 can issue concurrent read and write transactions to the same address on its AXI master ports
27	New	367118	Cat 2	A clean eviction received by L220 can incorrectly mark dirty data as clean

© Copyright ARM Limited 2004-2009. All rights reserved.

28	New	367714	Cat 2	In exclusive cache configuration, a read targetting data in a dirty cache line can corrupt L220

12 Oct 2005: Changes in Document v4

Page	Status	ID	Cat	Summary
23	Updated	351241	Cat 2	L220 RTL problem with multi-cycle path into AXI master interface
24	New	362460	Cat 2	Read-after-write hazards can be incorrectly handled between write allocate linefills and evictions

02 Jun 2005: Changes in Document v3

Page	Status	ID	Cat	Summary
22	New	350705	Cat 2	Data parity error can incorrectly occur to the slave port following a cached read
23	New	351241	Cat 2	L220 RTL problem with multi-cycle path into AXI master interface

18 Feb 2005: Changes in Document v2

Page	Status	ID	Cat	Summary
14	New	333649	Cat 1	In an obscure occurence of transactions the master port will provide incorrect address control data on a linefill
15	New	336152	Cat 1	In an IEM configuration of the L220 the CLKEN signals to the P, M0 & M1 ports are incorrectly tied to 1'b1
16	New	338093	Cat 1	Cache maintenance operations may fail in a multicore system, when a LOCK and peripheral access occurs concurrently
17	New	342249	Cat 1	In some circumstances the LRB is invalidated in the middle of a read transaction stopping the read from completing.
20	New	333646	Cat 2	Exclusive access when cache disabled gives incorrect address control data (ARLOCK & ARSIDEBAND)
21	New	333647	Cat 2	WA events occur when all ways are locked
40	New	331224	Cat 3	Contains no functionality to disable RAM interface during SCAN shift

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata Summary Table

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

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

ID	Cat	Summary of Erratum	r0p0	r1p0	r1p1	r1p2	r1p4-00rel0	r1p4-01rel0	r1p5	r1p6	r1p7-00rel0	r1p7-01rel0
333649	Cat 1	In an obscure occurence of transactions the master port will provide incorrect address control data on a linefill	X									
336152	Cat 1	In an IEM configuration of the L220 the CLKEN signals to the P, M0 & M1 ports are incorrectly tied to 1'b1		X								
338093	Cat 1	cache maintenance operations may fail in a multicore system, when a LOCK and peripheral access occurs concurrently			X							
342249	Cat 1	In some circumstances the LRB is invalidated in the middle of a read transaction stopping the read from completing.			X							
401552	Cat 1	“mbist_ce_reg” and “mbist_we_reg” need to be reset					X		X	X	X	
333646	Cat 2	Exclusive access when cache disabled gives incorrect address control data (ARLOCK & ARSIDEBAND)	X									
333647	Cat 2	WA events occur when all ways are locked	X									

© Copyright ARM Limited 2004-2009. All rights reserved.

ID	Cat	Summary of Erratum	r0p0	r1p0	r1p1	r1p2	r1p4-00rel0	r1p4-01rel0	r1p5	r1p6	r1p7-00rel0	r1p7-01rel0
350705	Cat 2	Data parity error can incorrectly occur to the slave port following a cached read.				X						
351241	Cat 2	L220 RTL problem with multi-cycle path into AXI master interface				X						
362460	Cat 2	Read-after-write hazards can be incorrectly handled between write allocate linefills and evictions				X						
364369	Cat 2	L220 can issue concurrent read and write transactions to the same address on its AXI master ports				X						
367118	Cat 2	A clean eviction received by L220 can incorrectly mark dirty data as clean				X						
367714	Cat 2	In exclusive cache configuration, a read targetting data in a dirty cache line can corrupt L220				X						
372114	Cat 2	Some imprecise error responses from L3 memory are not reported through interrupts				X						
373382	Cat 2	Starting a background clean or clean-and-invalidate maintenance operation can prevent the allocation of a cache line				X						

© Copyright ARM Limited 2004-2009. All rights reserved.

ID	Cat	Summary of Erratum	r0p0	r1p0	r1p1	r1p2	r1p4-00rel0	r1p4-01rel0	r1p5	r1p6	r1p7-00rel0	r1p7-01rel0
379681	Cat 2	Aborted linefills are incorrectly allocated into the L2 cache					X					
397385	Cat 2	In exclusive cache configuration, inner cacheable writeback data reads that are not linefills can deadlock or return incorrect data					X	X	X	X		
397533	Cat 2	Data parity errors may not be reported with the first read data of a burst					X	X	X	X		
400021	Cat 2	L220 AXI master interfaces do not maintain ARID or AWID during a locked sequence					X	X	X	X		
425331	Cat 2	Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations are received					X	X	X	X	X	X
484863	Cat 2	The Cache Sync operation does not guarantee that the Eviction Buffer is empty	X	X	X	X	X	X	X	X	X	X
331224	Cat 3	Contains no functionality to disable RAM interface during SCAN shift	X									
391290	Cat 3	The L220 does not make correct distinction between data and instructions for cache lockdown					X	X	X			

© Copyright ARM Limited 2004-2009. All rights reserved.

ID	Cat	Summary of Erratum	r0p0	r1p0	r1p1	r1p2	r1p4-00rel0	r1p4-01rel0	r1p5	r1p6	r1p7-00rel0	r1p7-01rel0
424665	Cat 3	ARSIDEBANDMx tied to 5’b00000 when the L2 cache is enabled					X	X	X	X	X	X
425600	Cat 3	BRESPP may be incorrect on the first write access with WSTRBP != 4’b1111 received by the Peripheral port after reset							X	X	X	X
533267	Cat 3	Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations received							X	X	X	X

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata - Category 1

333649: In an obscure occurence of transactions the master port will provide incorrect address control data on a linefill

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 1, Present in: r0p0-00REL0, Fixed in r1p0-00REL0. Unchanged in this document.

Description

This problem occurs when the CLKEN functionality is used and can be seen to cause the linefill buffer stall signal to transition from an inactive to active state before finally returning to inactive state between two CLKEN pulses on the master port.

Implications

The solution requires a fix to the master block of the L220 cache controller to provide logic to protect the lfb_stall signal from possible CLKEN ratios.

Workaround

There is no work around to this problem.

© Copyright ARM Limited 2004-2009. All rights reserved.

336152: In an IEM configuration of the L220 the CLKEN signals to the P, M0 & M1 ports are incorrectly tied to 1'b1

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 1, Present in: r1p0, Fixed in r1p1. Unchanged in this document.

Description

The issue occurs when the IEM slices are configured and instantiated in the L220_noram RTL. When the slices are used in a synchronous mode of operation there is no CLKEN logic in the slices to deal with different ratios of synchronous clocks. In the IEM configuration of L220, the CLKEN logic within the Peripheral port, M0 and M1 master ports has been incorrectly tied off. This does not allow the synchronous mode of operation to work correctly with the slices enabled.

The modes of operation that are not affected.

- All synchronous modes of operations, where the IEM slices are absent.
- Asynchronous operation of the IEM slices.

Implications

The solution requires a fix to the L220_noram level of RTL, where a level shifted version of the CLKEN needs connecting to the Peripheral port, and M0 and M1 masters. This fix only affects functionality when the IEM slices are enabled.

Workaround

There is no work around to this problem if IEM is being used. L220 functionality is not affected when the slices are not configured.

© Copyright ARM Limited 2004-2009. All rights reserved.

338093: Cache maintenance operations may fail in a multicore system, when a LOCK and peripheral access occurs concurrently

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 1, Present in: r1p1, Fixed in r1p2. Unchanged in this document.

Description

If the L220 peripheral port can receive a background operation at the same time as a slave receives a LOCK operation, then an issue could occur where by a background operation may not complete, particularly if the locked operation occurs at a specific time during the background operation. If the L220 is used with a single ARM core (1176/1156) this issue will not occur as a background operation can not occur at the same time as a locked sequence.

Implications

To add an additional register to the peripheral port to capture the data value on the bus so that when the locked access completes the background operation can resume cache maintenance operation.

Workaround

If a system connected to the slave ports and the peripheral port can issue both a Lock and background operation at the same time, there is no workaround to this problem. In the case where the system driving the L220 is a single ARM1176 or ARM1156 this problem cannot occur.

© Copyright ARM Limited 2004-2009. All rights reserved.

342249: In some circumstances the LRB is invalidated in the middle of a read transaction stopping the read from completing.

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 1, Present in: r1p1, Fixed in r1p2. Unchanged in this document.

Description

If a write starts or crosses a cache line while a read is in progress in the same slave port and an outstanding invalidate of the Line Read Buffer (LRB) is waiting to occur e.g. due to a cache sync, the LRB is incorrectly cleared. Thus preventing the read from completing.

Implications

A change is required to the RTL so that a change in the write index does not activate the invalidation enable, only a change to the read index should.

Workaround

There is no work around to this problem.

© Copyright ARM Limited 2004-2009. All rights reserved.

401552: “mbist_ce_reg” and “mbist_we_reg” need to be reset

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 1, Present in: r1p4-00rel0, r1p5, r1p6 and r1p7-00rel0, Fixed in: r1p4-01rel0 and r1p7-01rel0. Unchanged in this document.

Description

The registers mbist_ce_reg[10:0] and mbist_we_reg declared in the L220_bist_if.v verilog file are not reset. As a result, they have an unknown value (1’bX in simulation, VDD or VSS on silicon) after the L220 cache controller has been reset. However, the output of these registers is used to control some RAM output signals in L220_cache_ctl.v, even if the MBIST mode is not enabled (i.e. in functional mode).

Implications

On silicon, the value of uninitialised registers is unpredictable. If the output of the mbist_ce_reg[10:0] and mbist_we_reg registers is not VSS when the power is turned on, failures (incorrect values driven to the RAMs) occur in functional mode. Under these circumstances, this means that L220 cannot be enabled.

In netlist simulations, failures also occur in functional mode and are characterised by X propagation on the RAM interface.

Note that this erratum cannot be seen on RTL simulations.

Workaround

There is one possible workaround for this erratum. Before L220 is enabled, e.g. during the reset sequence of L220 (nRESET asserted LOW), MTESTON can be driven HIGH for at least one clock cycle while MBISTCE[10:0] and MBISTWE are driven LOW. This allows propagating known and inactive values to the output of the mbist_ce_reg[10:0] and mbist_we_reg registers.

This workaround can be implemented as follows:

- if the L220 MBIST interface is used

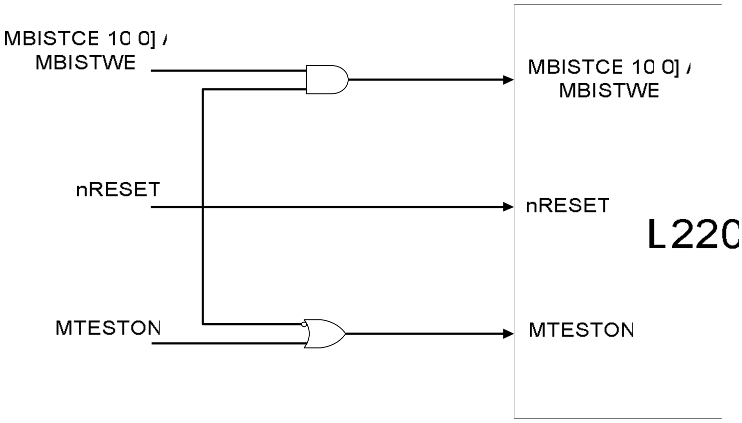


Figure 1: Implementation workaround for 401552 erratum when the L220 MBIST interface is used

© Copyright ARM Limited 2004-2009. All rights reserved.

- if the L220 MBIST interface is not used

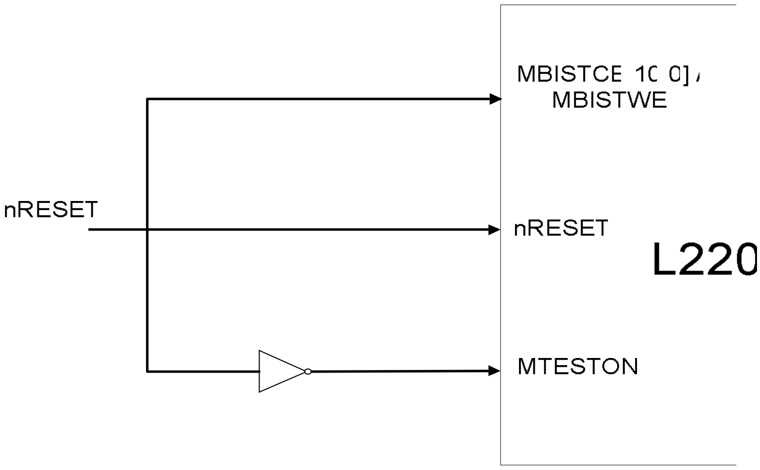


Figure 2: Implementation workaround for 401552 erratum when the L220 MBIST interface is not used

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata - Category 2

333646: Exclusive access when cache disabled gives incorrect address control data (ARLOCK & ARSIDEBAND)

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r0p0-00REL0, Fixed in r1p0-00REL0. Unchanged in this document.

Description

The issue arises when the cache control register is set to disabled and a cacheable exclusive transaction is passed through the L220.

Implications

The issue causes the transaction to set the ARLOCK and ARSIDEBAND signals to 'zero'

Workaround

There is no work around to this problem

© Copyright ARM Limited 2004-2009. All rights reserved.

333647: WA events occur when all ways are locked

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r0p0-00REL0, Fixed in r1p0-00REL0. Unchanged in this document.

Description

The L220 provides WA events when all ways are locked on the event monitor and the event counters if enabled.

Implications

WA events are not gated if all ways are locked.

Workaround

There is no work around to this problem

© Copyright ARM Limited 2004-2009. All rights reserved.

350705: Data parity error can incorrectly occur to the slave port following a cached read.

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

The problem occurs due to differences in the pipelining of a Data RAM read access and the generation of the Data parity error signal. Due to these differences, under certain circumstances the logic within the slave port can incorrectly generate a "slave error" response on the read response channel.

Implications

The product requires a fix to the slave port (issue common for S0 and S1 ports) so that the data parity error signal can be evaluated within the slave port with the same timing as the read data.

Workaround

This problem will only occur with parity enabled, there is no workaround to this defect if parity is enabled.

© Copyright ARM Limited 2004-2009. All rights reserved.

351241: L220 RTL problem with multi-cycle path into AXI master interface

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

In the L220 masters, it is possible to use the ACLKENMx clock enable inputs to allow the L2 cache controller communicating with slaves running synchronously at lower clock frequency. The fundamental requirement for the usage of these clock enables is that the inputs of the L220 masters must only be sampled when their corresponding ACLKENMx clock enable is HIGH. The requirement on the outputs of the L220 masters is not exactly the same: they can be initially asserted on any rising edge of the L220 clock but they can then only be updated when ACLKENMx is HIGH. Note that the usage of the ACLKENMx clock enables do not infer any multi-cycle paths from a synthesis or static timing analysis point of view between the L220 master ports and the slaves running at lower clock frequency.

The requirements listed above on the inputs are not correctly implemented in the L220 master ports. More particularly, the RVALIDMx and RLASTMx input signals are used to drive the RREADYMx output signal without taking into account the value of ACLKENMx.

Conditions

ACLKENMx is used to make L220 communicate with slaves running synchronously at lower clock frequency.

Implications

L220 cannot communicate with the slaves running at lower clock frequency.

Workaround

There are two possible workarounds for this erratum:

- If the IEM register slices are implemented, use them as asynchronous interfaces between the L220 master ports and the slaves running at lower clock frequency to remove the need for the clock enables (ACLKENMx tied HIGH).

- If they are not implemented, run the whole L220 cache controller at lower clock frequency to remove the need for the clock enables (ACLKENMx tied HIGH).

© Copyright ARM Limited 2004-2009. All rights reserved.

362460: Read-after-write hazards can be incorrectly handled between write allocate linefills and evictions

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

In L220, a write allocation linefill and dirty data eviction to the same half or full line occurring at the same time can cause the L2 cache controller to be allocated with out-of-date data.

Conditions

- Two cache allocation linefills cause two half lines, HL0 and HL1, to be transferred to the eviction buffer. HL0 is processed first by master M1.
- The HL1 evicted half line address corresponds to a memory region defined as Outer Write-Back Write-Allocate (OWBWA) or Outer Write-Back Non Write allocate (OWBNWA) with the "Write allocate override" bit of Auxiliary Control register (bit 23) set.
- Then, a write buffer slot containing a write access to the same address as HL1 is processed. The write buffer sends a lookup request to the cache controller and receives a cache miss answer as the half line is now in the eviction buffer.
- The write buffer slot is then transferred to the write-allocate buffer that, if some data is missing in the line, will request a linefill transaction to master M1.
- In the case where master M1 has not yet started the HL1 half line eviction transaction, the linefill will occur first in the L3 memory system, fetching out of date data.

Implications

As the write-allocation linefill transaction can start on master port M1 before the dirty data eviction transaction starts, the data allocated in L2 cache can be out of date.

Workaround

There are two possible workarounds for this erratum:

- Set the "Force write allocate" bits of Auxiliary Control register (bit 24 and 23) to 2'b01, ie. "Force no allocate"
- Change memory regions attributes so that "allocate on read miss only" policy is used for all outer write-back regions. In this case, it must also be checked that the "Force write allocate" bits of Auxiliary Control register (bit 24 and 23) are not equal to 2'b10.

© Copyright ARM Limited 2004-2009. All rights reserved.

364369: L220 can issue concurrent read and write transactions to the same address on its AXI master ports

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

The AMBA AXI Protocol specification (ARM IHI 0022) defines the following behaviour for masters: "There are no ordering restrictions between read and write transactions and they are allowed to complete in any order. If a master requires a given relationship between read and write transaction then it must ensure that the earlier transaction is complete before issuing the later transaction. [...] In the case of writes the transaction can only be considered complete when the write response is received by the master, it is not acceptable to consider the write transaction complete when all the write data is sent."

As a consequence of this, it is the resposibility of L220 to manage the hazards between all the requests it handles to make sure that the order of transactions is respected.

This erratum describes a scenario where L220 does not comply with the AMBA AXI Protocol specification on this aspect.

When the eviction of a full or half cache line is being serviced, L220 issues a write request on its master 1 AXI port. If a read request targetting data in this evicted cache line is received by one the L220 slave ports at this time, it will miss in the cache. A linefill will then occur on one of the L220 master ports. This read access will be incorrectly issued as soon as the last data of the eviction is sent by the master, and thus before the write response is received, which violates the AXI ordering model between read and write transactions mentioned above.

Conditions

- A full or half cache line is being evicted on the L220 master 1 port because of natural replacement or cache maintenance operation;
- a read request targetting data in the evicted cache line is received on one of the two slave ports of L220;
- this request is serviced by the cache controller and misses in the cache;
- a linefill occurs on one of the two master ports of L220 as soon as the last data of the eviction is sent by the master 1 AXI port.

Implications

Under the conditions described above, L220 issues concurrent read and write transactions on its master ports. As these accesses can complete in any order according to the AXI specification, read data cannot be relied upon.

Workaround

There are two workarounds for this erratum:

© Copyright ARM Limited 2004-2009. All rights reserved.

- use write-through memory attributes instead of write-back;
- force the write-through behaviour of L220 by setting the bit [1] of the Debug Control Register.

© Copyright ARM Limited 2004-2009. All rights reserved.

367118: A clean eviction received by L220 can incorrectly mark dirty data as clean

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

On the two main slave ports of L220, the WRITEBACKSx[1:0] (where x=1, resp. 0, indicates slave 1, resp. slave 0) input pins are used to indicate if the write access is an eviction (WRITEBACKSx[0]=1) and, in this case, if the eviction is clean (WRITEBACKSx[1]=1) or dirty.

For a write transaction with AWCACHESx indicating write-back and WRITEBACKSx[0]=1, if the access misses in the cache, the behaviour of L220 depends on the value of WRITEBACKSx[1]. If WRITEBACKSx[1]=0 (dirty eviction), the data is allocated in the cache and marked as dirty. If WRITEBACKSx[1]=1 (clean eviction), the data is allocated in the cache and marked as clean.

For a write transaction with AWCACHESx indicating write-back and WRITEBACKSx[0]=1, if the access hits in the cache, the behaviour of L220 also depends on the value of WRITEBACKSx[1]. If WRITEBACKSx[1]=0 (dirty eviction), the data is written to the cache and marked as dirty. If WRITEBACKSx[1]=1 (clean eviction), no action is required. This last behaviour is not correctly implemented in L220: the data is written to the cache and erroneously marked as clean even if the line was previously marked as dirty.

Conditions

- A write access is received by L220 with write-back memory attributes and WRITEBACKSx=11;
- the write request hits in the cache;
- the dirty bits are overwritten to indicate clean data.

Implications

Under the conditions described above, dirty data can erroneously be marked as clean and thus data is lost.

In order to better assess the impact of this erratum, it has to be noticed that the only ARM processor able to drive HIGH the WRITEBACKSx[1] signal is the ARM11 MPCore and that, in this case, it can only be driven HIGH when MPCore is configured to work in exclusive cache. So, when connected to an ARM processor core, L220 will only exhibit the faulty behaviour when configured in exclusive cache even if the WRITEBACKSx[1:0] inputs are evaluated in all configurations.

Workaround

Several workarounds are possible for this erratum:

- Use write-through memory attributes instead of write-back;
- Force the write-through behaviour of L220 by setting the bit [1] of the Debug Control Register;
- When L220 is connected to MPCore, do not enable the exclusive cache configuration.

© Copyright ARM Limited 2004-2009. All rights reserved.

367714: In exclusive cache configuration, a read targetting data in a dirty cache line can corrupt L220

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

When L220 is configured to work in exclusive cache, if a read access with inner write-back memory attributes (indicated by ARSIDEBANDSx[4:1] inputs) hits in the cache, the corresponding cache line is cleaned (if dirty) and invalidated after data is returned to the L1 memory system.

ARSIDEBANDSx[4:1], refered as "inner" as opposed to "outer" (cf. ARCACHESx[3:0]), reflect the memory attributes taken into account by the Level 1 memory system. ARSIDEBANDSx[4:1]=1111 or ARSIDEBANDSx[4:1]=0111 indicate inner write-back memory attributes.

In this context, if the targetted cache line is dirty or partially dirty, the cache controller sends a request to the eviction buffer and must stall all other lookup requests if any till the eviction buffer sends an acknoledgement. This behaviour is not correctly implemented as the cache controller accepts to service other pending lookup requests before having received the acknoledgement. This results in an incorrect behaviour of the cache controller and can lead to data corruption.

Conditions

- Exclusive cache configuration is enabled;
- A read access with inner write-back memory attributes hits in the cache;
- The corresponding cache line is dirty and thus needs to be cleaned before being invalidated;
- Other lookup requests are pending at the entry of the cache controller.

Implications

Under the conditions described above, the subsequent behaviour of L220 cannot be relied upon.

Workaround

Several workarounds are possible for this erratum:

- Do not enable the exclusive cache configuration;
- If the exclusive cache configuration is enabled, use inner write-through memory attributes.

© Copyright ARM Limited 2004-2009. All rights reserved.

372114: Some imprecise error responses from L3 memory are not reported through interrupts

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

L3 memory error responses received by the master ports of L220 are reported via the AXI error response signals of the slave ports or the DECERRINTR and SLVERRINTR cache interrupts. A non-secure write-allocate cacheable write transaction with the Override Security Check bit set or a secure write-allocate cacheable write transaction causes an L3 memory read access if the data to be written does not fill a complete cache line. In this context, a potential error response resulting from the L3 read access is only meant to be reported through the cache interrupts. Due to the erratum described here, this error response reporting is not done in some cases.

Conditions

This problem occurs when the following conditions are met:

- A non-secure cacheable write transaction with the Override Security Check bit (cf. bit [25] of the Auxiliary Control Register) set or a secure cacheable write transaction is received by one of the slave ports (S0 or S1) of L220.
- This write access goes into the write buffer and it misses in the cache.
- It is a write allocate transaction, either due to its initial memory attributes (cf. AWCACHE) or due to the Force write allocate feature (cf. bits [24:23] of the Auxiliary Control Register).
- The write access goes to the write allocate buffer.
- The resulting cache line waiting to be allocated only misses eight or less bytes within a doubleword, then a single doubleword read access is sent to the L3 memory by the M1 master.
- The read response for this access is SLVERR, resp. DECERR.
- This error response is forwarded to the write allocate buffer, which invalidates the write transaction so that the allocation is not done. However the error response is not forwarded to the event monitor module so that the cache interrupt SLVERRINTR, resp. DECERRINTR, is not asserted.

Implications

Under the conditions above, an L3 error response is lost and not reported to the system. This has only an impact on a system that handles the DECERRINTR and SLVERRINTR cache interrupts issued by L220.

Workaround

A workaround for this erratum is only required when L220 is implemented in a system that handles the DECERRINTR and SLVERRINTR cache interrupts. In this case, the following workarounds can be applied:

- Set the bits [24:23] of the Auxiliary Control Register to 2'b01 so that write allocate is disabled.
- Do not use the write-allocate memory attribute.

© Copyright ARM Limited 2004-2009. All rights reserved.

373382: Starting a background clean or clean-and-invalidate maintenance operation can prevent the allocation of a cache line

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 2, Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

When a background clean or clean-and-invalidate cache maintenance operation is requested to the L220, all the targeted ways are considered as being locked by the cache controller. If such a cache maintenance operation is requested on all non-locked available ways, or more ways, between the time a data line has been transferred to the write allocate buffer and the time that the buffer request line allocation to the cache controller has occurred, the cache controller won't allocate the line as all ways are seen as locked.

Conditions

This problem occurs when the following conditions are met:

- A write transaction to a write-back write allocate memory region, or to a cacheable memory with Force Write-allocate bit set in Auxiliary Control Register, is pending in one of the write buffer slot.
- This write buffer slot is drained because of a new write transaction to the write buffer, a hazard or a non-cacheable transaction.
- The write transaction misses in the cache, causing the line to be transferred from the write buffer to the write allocate buffer. That can only happen if some ways are not locked. If all ways are locked when the write buffer receives the cache miss information, the write buffer forces the write to L3 memory system.
- A background clean or clean-and-invalidate cache maintenance operation is requested which targets at least all of the non-locked ways before the write-allocate buffer is requesting data line allocation to the cache controller. An automatic cache sync is triggered by the cache maintenance request, causing the write-allocate buffer to be drained.
- When the write allocate buffer is requesting allocation to the cache controller, the cache controller already considers all cache maintenance targeted ways to be locked, so that if all ways are seen as locked, the allocation is not performed.
- The write access that has been transferred from the write buffer to the write-allocate buffer is subsequently lost.

Implications

Performing a background clean or clean-and-invalidate cache maintenance request on all non-locked available ways can cause data corruption because the write transaction sitting in the write allocate buffer at that time can be lost.

Even if the Invalidate By Way operation has similar behaviour in terms of way lockdown, no work-around needs to be applied for that operation as any updated data in dirty cache line will be lost anyway when the line will be invalidated.

© Copyright ARM Limited 2004-2009. All rights reserved.

Workaround

The software work-around for this erratum is to perform an explicit Cache Sync operation before initiating a background clean or clean-and-invalidate cache maintenance operation. By doing so, the write-allocate buffer is empty when the background cache operation is initiated, therefore removing the erratum conditions.

That work-around relies on the atomicity of the Cache Sync and cache maintenance operations from cache controller perspective so that it is ensured that no write transactions are performed between them. If it appears to be impossible to prove atomicity of the operations, cache maintenance operations should be performed one way at a time so that at least one way is always seen as non-locked. This has no or little performance impact for Clean By Way and Clean-And-Invalidate By Way operations as they are performed one way at a time anyway by the cache controller.

© Copyright ARM Limited 2004-2009. All rights reserved.

379681: Aborted linefills are incorrectly allocated into the L2 cache

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p4, Fixed in r1p5. Unchanged in this document.

Description

When a cacheable read access misses in the L2 cache, a linefill is requested and a read burst of four 64-bit data is sent on one of the L220 master ports. If an error response (SLVERR or DECERR) is then received on at least one piece of data (including the case where all data are aborted), it is reported by the L220 slave port if the originating read access targeted the aborted data. It is also reported via the SLVERRINTR or DECERRINTR interrupt signals if they are not masked. Additionally, the line must not be allocated into the cache. Due to this erratum, the aborted line is allocated into the L2 cache.

Conditions

This problem occurs when the following conditions are met:

1. A cacheable read access is received by one of the L220 slave ports.
2. A lookup request is sent to the cache controller.
3. The request misses in the L2 cache.
4. A linefill request is sent to the L220 master port linked to the originating slave port.
5. A read burst of four 64-bit data is sent by the master.
6. At least one piece of data is returned with an error response (SLVERR or DECERR).
7. At the end of the burst, the data is erroneously allocated into the L2 cache.

Implications

If no error response is generated by the L3 memory system, this erratum does not have any impact.

If the L3 memory system can generate error responses with a granularity of at least one cache line, it is guaranteed that the error response received on the linefill will be immediately forwarded on the slave side of L220.

If the L3 memory system can generate error responses with a granularity smaller than one cache line, the error response is not guaranteed to be forwarded on the slave side of L220. However, the error response will trigger the SLVERRINTR or DECERRINTR interrupts signals if they are not masked.

In the two cases above where L3 can generate error responses, a subsequent access to the same cache line will hit in the L2 cache and will not report any abort, potentially causing some system failures.

Workaround

If the system driving L220 cannot resume normal execution after having received the error response information (eg. the system needs to apply a reset), this erratum does not have any impact.

In all other cases, there is no workaround for this erratum.

© Copyright ARM Limited 2004-2009. All rights reserved.

397385: In exclusive cache configuration, inner cacheable writeback data reads that are not linefills can deadlock or return incorrect data

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p4, r1p5, r1p6. Fixed in r1p7. Unchanged in this document.

Description

In exclusive cache configuration, cache lines containing data are located either in L1 data cache or L2 cache, but not both. When an inner cacheable writeback (ie. ARSIDEBANDSx[4:1] = 4’b0111 or 4’b1111) data read request misses in the L2 cache, data is returned to L1 but not allocated in L2 cache. This erratum occurs when the read request received by the L220 AXI slave interfaces is not a linefill (ie. a burst of four 64-bit accesses) and misses in the cache. In this case, as the slave interface always expects the read request to be a linefill, the L2 cache controller can deadlock or return incorrect data.

Conditions

This problem occurs when the following conditions are met:

1. Exclusive cache configuration is enabled.
2. An inner cacheable writeback data read request is received by one of the L220 AXI slave interfaces.
3. The request misses in the L2 cache.
4. A linefill request is sent to the L220 master port linked to the originating slave port.
5. A read burst of four 64-bit data is sent by the master port.
6. Data is received by the master port and forwarded to the slave port.
7. If the length of the original read request was not four, a deadlock can occur. If the size of the read request was not 64-bit, incorrect data can be returned by the slave port.

Implications

If the above conditions are met, either a deadlock can occur or incorrect data can be returned.

Note that only linefills are expected to be sent by the L1 memory system as inner cacheable writeback reads if the L1 data cache is enabled. So this erratum can only occur if the L2 cache is enabled in exclusive cache configuration while the L1 data cache is disabled.

Workaround

This erratum can be avoided by making sure that, in exclusive cache configuration:

- The L1 data cache is enabled before the L2 cache.
- The L2 cache is disabled before the L1 data cache.

© Copyright ARM Limited 2004-2009. All rights reserved.

397533: Data parity errors may not be reported with the first read data of a burst

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p4, r1p5, r1p6. Fixed in r1p7. Unchanged in this document.

Description

When a cacheable read request hits in the L2 cache and parity is enabled, the cache line returned by the Data RAM is checked against the parity bits returned by the Data Parity RAM. If the parity is not the same, a data parity error is returned to the L220 AXI slave interface one clock cycle after the cache line. The slave interface is then responsible for indicating SLVERR via the RRESPSx[1:0] read response signals. Due to this erratum, SLVERR is not reported with the first data of the read burst.

Conditions

This problem occurs when the following conditions are met:

1. Parity is implemented and enabled.
2. A cacheable read request is received by one of the L220 AXI slave interfaces.
3. The request hits in the L2 cache.
4. Data is returned from the Data RAM to the slave port.
5. A data parity error is detected and returned to the slave port.
6. The first read data is returned on the AXI bus with an OKAY read response.
7. If any, following data are returned with a SLVERR read response.

Implications

As the first data is not aborted, the data parity error may not be directly reported to the processor core and the latter may incorrectly use the read data.

It is likely that the cacheable read request is actually a linefill, ie. a burst of four 64-bit accesses, issued by the L1 memory system. In this case, as three data out of four are aborted, the cache line will not be allocated in the L1 cache.

Note that the data parity error is also reported via the L220 PARRDINTR and L2CCINTR interrupt lines. If these interrupts are not masked, this ensures that the data parity error is reported to the system.

Workaround

If the PARRDINTR or L2CCINTR is propagated to the system and the latter cannot recover from a data parity error, then this erratum does not have any impact.

Otherwise, there is no workaround for this erratum.

© Copyright ARM Limited 2004-2009. All rights reserved.

400021: L220 AXI master interfaces do not maintain ARID or AWID during a locked sequence

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p4, r1p5, r1p6. Fixed in r1p7. Unchanged in this document.

Description

The AMBA AXI Protocol Specification (ARM IHI 0022B) requires that “the master must ensure that all transactions within a locked sequence have the same ARID or AWID value”. This feature is not correctly implemented in the L220 AXI master interfaces. The ARID or AWID value associated with the transaction responsible for unlocking the bus is not guaranteed to be the same as the rest of the locked sequence.

Conditions

This problem occurs when the following conditions are met:

1. A non-cacheable locked transaction is received by one of the L220 AXI slave interfaces with AxID[5:0] equal to ID0.
2. It is forwarded to the relevant L220 AXI master interface, which issues a locked transaction with the following ID: {1’b0, ID0, 1’b1}.
3. At this point, the bus is locked on the master side of L220.
4. A non-locked transaction is received by the originating slave interface with AxID[5:0] equal to ID0.
5. This transaction is forwarded to the same master interface, which issues a non-locked transaction with the following ID: 8’b0000_0000.

Implications

If the AXI system on the master side of L220 relies on the AXI ID to correctly handle locked sequences, then this erratum can cause some wrong behaviour when locked sequences occur. However, this is not expected to be the case in the majority of systems.

Workaround

If the AXI system on the master side of L220 ignores the AXI ID during locked sequences, then no workaround is required. On the contrary, there is no workaround for this erratum.

© Copyright ARM Limited 2004-2009. All rights reserved.

425331: Potential deadlock in certain system configurations if Write Buffer not empty when specific maintenance operations are received

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r1p4, r1p5, r1p6, r1p7. Unchanged in this document.

Description

The L220 TRM describes several system configurations for connecting L220 to an ARM processor (see sections 1.2 and 1.3). These configurations are particularly important for showing how the L220 Peripheral port is accessed. This erratum concerns the system configuration described in section 1.3 where the accesses to the Peripheral port first go through L220 and an L3 interconnect before going back to the Peripheral port. This is for example the system configuration that is usually implemented with the ARM11 MPCore processor.

Certain L2 cache maintenance operations and accesses to the Peripheral port only start when the L220 Write Buffer is empty. Their corresponding write response is only sent on AXI when the operation is complete or the access has taken effect.

A deadlock can occur if the Write Buffer is not empty when such an operation or access is received by the L220 Peripheral port.

Conditions

The problem occurs when the following conditions are met:

1. The Core and L220 are implemented in a system using a similar connectivity as described in section 1.3 of the L220 TRM.
2. The L220 Write Buffer is not empty and contains valid data that needs to be sent to L3 (eg. non-cacheable or write-through or write-back miss no write-allocate).
3. One of the following write accesses targeting the L220 Peripheral port is received by the S1 L220 AXI slave port in 2 slave 2 master configuration or by S0 or S1 in 2 slave 1 master configuration:
   - Cache Sync operation (0x730)
   - Clean and/or Invalidate by PA operations (0x770, 0x7B0 and 0x7F0)
   - Clean and/or Invalidate by Index/Way operations (0x7B8 and 0x7fF)
4. The write access is forwarded to the M1 L220 AXI master port, is issued on AXI and goes through the L3 interconnect.
5. It is received by the L220 Peripheral port.
6. The Peripheral port sends a drain request to the Write Buffer.
7. The Write Buffer (WB) starts draining and sends a request to M1.
8. M1 cannot issue the WB request on AXI as it is busy with the Peripheral port write access.

Implications

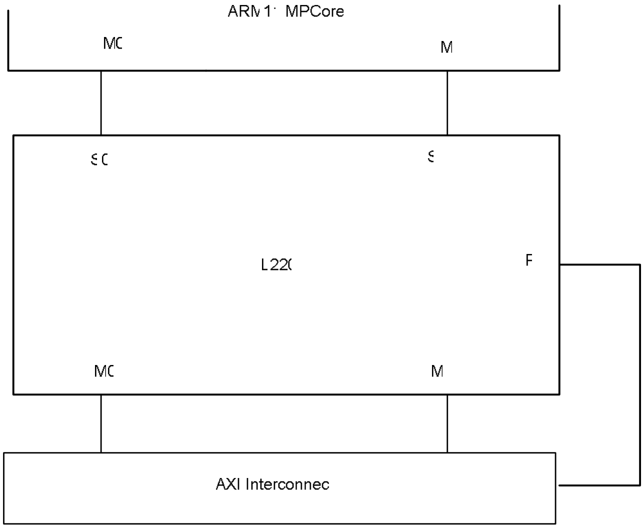
Under the conditions listed above, a deadlock occurs.

© Copyright ARM Limited 2004-2009. All rights reserved.

Workaround

No workaround is required if the path from the Core to the L220 peripheral port is not through the L220.

If the system connectivity is exactly as described in the figure below (note the M0 → S0 and M1 → S1 connections between ARM11 MPCore and L220 and the 2slaves/2masters configuration for L220),



then a workaround for this erratum is to run the following pseudocode sequence to launch the maintenance operations:

```
Disable interrupts
LDREX to maintenance operation register
STREX to maintenance operation register
Re-enable interrupts
```

Note that the L220 maintenance operation registers are expected to be mapped with Shared Device memory attributes.

Note also that the L220 Peripheral port does not support AXI exclusive accesses so that it will answer OKAY to the STREX. As a consequence, bit 0 of the STREX destination register may be set at the end of the store. This value should not be checked as it does not reflect the actual result of the write.

If the connection from the Core to L220 peripheral port is through the L220, but the system connectivity is not exactly the same as in the figure above, the workarounds are the following ones:

- Replace the Cache Sync operation by a dummy SWP to a cacheable or non-cacheable memory area that will go through one of the two main L220 slave ports (S0 or S1). The AXI locked transactions received by the L220 slave ports are only treated once all buffers are empty, thus acting like a Cache Sync. This dummy SWP must not target the L220 Peripheral port, see erratum 533267 for more information.
- Replace the Clean and/or Invalidate by PA or Index/Way operations by Clean and/or Invalidate by Way operations.

© Copyright ARM Limited 2004-2009. All rights reserved.

484863: The Cache Sync operation does not guarantee that the Eviction Buffer is empty

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 2, Present in: r0p0, r1p0, r1p1, r1p2, r1p4, r1p5, r1p6, r1p7. Updated in this document.

Description

According to the L220 TRM (see Table 3-17), the Cache Sync operation drains the Write Buffer and Eviction Buffer to L3 main memory and drains the Linefill Buffers and the Write Allocate Buffer to the L2 Data RAM. The actual behaviour of the L220 is that the Cache Sync operation only ensures that the Write Buffer and the Write Allocate Buffer are empty.

Conditions

The problem occurs when the following conditions are met:

1. The L220 cache is holding Write Back data.
2. A Clean or Clean and Invalidate operation is initiated.
3. Dirty data is transferred into the Eviction Buffer, which automatically sends a write request to the L220 M1 AXI master port.
4. The maintenance operation completes and this is confirmed by a read to the corresponding L220 register.
5. A Cache Sync operation is initiated.
6. The Write Buffer is empty. If the Write Allocate Buffer is not empty, it drains to the L2 cache.
7. The Cache Sync operation completes irrespective of the state of the Eviction Buffer write request, which might still be pending between the Eviction Buffer and the master port or issued on AXI, waiting for its buffered response.

Implications

This erratum has implications where cache maintenance has been performed in order to ensure that memory updates present in the L220 cache need to be written back to L3 memory. In those situations, the Cache Sync operation is not sufficient to ensure that the writebacks have completed.

Two situations are commonly encountered where this is necessary:

- Cache maintenance is performed to ensure that a set of memory updates are visible to an external observer, such as a DMA, as part of software managed coherency.
- Cache maintenance is performed to clean (and possibly invalidate) the cache as part of powering down the L2 cache.

© Copyright ARM Limited 2004-2009. All rights reserved.

Workaround

Several workarounds exist for this erratum:

- Replace the Cache Sync operation by a dummy SWP to an L1 non-cacheable and L2 cacheable or non-cacheable memory area that will go through one of the two main L220 slave ports (S0 or S1). The AXI locked transactions received by these slave ports are only treated once all buffers are empty, including the Eviction Buffer. This dummy SWP must not target the L220 Peripheral port.
- Do a dummy STR to L1 and L2 Normal Memory Non-cacheable memory area that will go into the L220 Write Buffer before launching the Cache Sync operation. This guarantees that the last eviction resulting from the clean operation has been written to L3 when the Cache Sync operation completes.

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata - Category 3

331224: Contains no functionality to disable RAM interface during SCAN shift

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Cat 3, Present in: r0p0-00REL0, Fixed in r1p0-00REL0. Unchanged in this document.

Description

The L220 IG Documentation states: (DII0104A r0p0)

"To ease any issues with scan testing the clocks and chip selects, driving the RAMs are gated with SCANENABLE, when scan shifting the RAMs are isolated from the serial shifting of logic values."

This functionality has not been implemented within the verilog RTL provided with r0p0 release, it is recommended that the end user take necessary measures on the outputs of L220 RAM interfaces to ensure that the clock and chip selects are disabled when SCANENABLE is high (shifting). Gating these outputs (RAM clocks / Chip selects) when SCANENABLE is high will also help ATPG tools test shadow logic between the registers and RAMs.

Implications

The RAMs are not isolated during scan shift, this may cause a loss of test coverage

Workaround

Add gating to *CS (DATACS, TAGCS & DIRTYCS and CLKOUT) so that when SCANENABLE is high these signals are disabled.

© Copyright ARM Limited 2004-2009. All rights reserved.

391290: The L220 does not make correct distinction between data and instructions for cache lockdown

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 3, Present in: r1p4, r1p5, Fixed in r1p6. Unchanged in this document.

Description

The Data and Instruction Lockdown registers permit to lock the L2 cache ways differently for instructions and data. If ARPROT[2] = 1'b0 (data access) for a linefill, the Data Lockdown Register is taken into account to choose the way in which the line can be allocated or not. If ARPROT[2] = 1'b1 (instruction access) for a linefill, the Instruction Lockdown Register is taken into account to choose the way in which the line can be allocated or not. This behaviour is not correctly implemented in L220. ARPROT[0] (normal/privileged) is used for lockdown purpose, instead of ARPROT[2].

Conditions

This problem occurs when the following conditions are met:

1. The Instruction and Data Lockdown registers have different programmed values.
2. An instruction (resp. data) linefill marked with ARPROTMx[2] = 1'b1 (resp. 1'b0) is issued by one L220 AXI master port.
3. When the AXI read transaction is complete, the master sends a request to the cache controller for allocation.
4. The master erroneously gives the inverted value of ARPROT[0] to the cache controller instead of ARPROT[2].
5. The cache controller might thus consider the wrong Lockdown register for allocation and incorrectly choose the victim way.
6. The cache controller might then erroneously allocate the line in a locked way.

Implications

This is not possible to configure L220 to make a distinction between instructions and data as far as the cache lockdown mechanism is concerned. The two Data and Instruction Lockdown registers must be programmed with the same value.

Workaround

There is no software workaround for this erratum.

There is a possible implementation workaround for this erratum. Indeed, ARPROT[0] indicates if the read transaction is normal or privileged. This information is not used within L220 and can only be relied upon at L3 level in case of Strongly Ordered or Device accesses. If ARPROT[0] is not used by the L3 memory system in that context, it is possible to workaround the erratum described above by connecting ARPROT[2] to the ARPROT[0] input on the L220 slave ports, as described in the figure below.

© Copyright ARM Limited 2004-2009. All rights reserved.

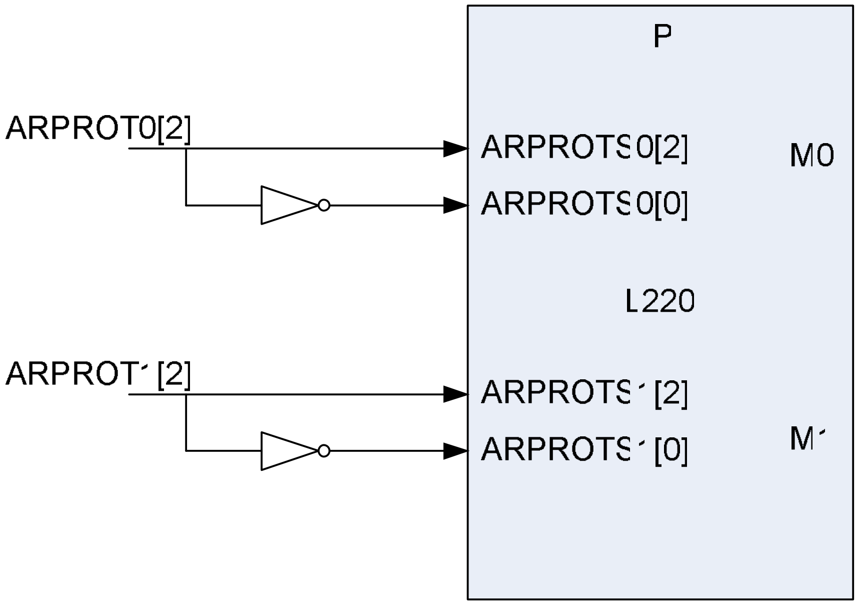


Figure 3: Implementation workaround for 391290 erratum

© Copyright ARM Limited 2004-2009. All rights reserved.

424665: ARSIDEBANDMx tied to 5’b00000 when the L2 cache is enabled

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 3, Present in: r1p4, r1p5, r1p6, r1p7. Unchanged in this document.

Description

In the L220 cache controller, the ARSIDEBAND[4:0] signals hold the shared attribute (bit [0]) and the inner cacheable attributes (bits [4:1]) of AXI read transactions. The L220 TRM says that, for non-cacheable read transactions, ARSIDEBANDMx[4:0] take the same value as the original one given by ARSIDEBANDSx[4:0] in all cases (see Table 2-8). This feature is not correctly implemented. When the L2 cache is enabled, ARSIDEBANDMx[4:0] outputs are constantly tied to 5'b00000, no matter the original values of ARSIDEBANDSx[4:0].

Conditions

The problem occurs when the following conditions are met:

1. The L2 cache is enabled.
2. A non-cacheable read transaction is received by one of the L220 AXI slave interfaces with some values on ARSIDEBANDSx[4:0], potentially different from 5'b00000.
3. The transfer is forwarded to the relevant L220 master interface.

Implications

Due to this erratum, the L3 memory system cannot use the ARSIDEBANDMx[4:0] signals sent by L220.

Note that the ARSIDEBANDMx[4:0] signals are not expected to be used in the majority of the systems implementing L220.

Note also that the ARSIDEBANDSx[4:0] signals are correctly used within the L220 cache controller. As a consequence, this erratum does not affect the correct behaviour of the exclusive cache configuration and the management of shared accesses.

Workaround

There is no workaround for this erratum.

© Copyright ARM Limited 2004-2009. All rights reserved.

425600: BRESPP may be incorrect on the first write access with WSTRBP != 4’b1111 received by the Peripheral port after reset

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 3, Present in: r1p5, r1p6, r1p7. Unchanged in this document.

Description

The L220 TRM (see section 2.2.4) says that single (AWLENP = 4'b0000) 32-bit (AWSIZEP = 3'b010) aligned (AWADDRP[1:0] = 2'b00) write accesses to the Peripheral port with WSTRBP != 4'b1111 return a SLVERR write response on BRESPP. This feature is not correctly implemented. Since reset, if no other write access to the Peripheral port has generated a SLVERR or DECERR response, then the first write access with the above characteristics may have an incorrect response on BRESPP.

Conditions

The problem occurs when the following conditions are met:

1. Since reset, no write access to the Peripheral port has generated SLVERR or DECERR responses.
2. There is no background maintenance operation running.
3. A write access is received by the L220 with the following characteristics:
    - AWSIZEP = 3'b010 (32-bit)
    - AWADDRP[1:0] = 2'b00 (32-bit aligned)
    - AWLENP = 4'b0000 (single)
    - It does not access the Auxliary Control Register or the L2 cache is disabled
    - The access has permission to access the target register from a security point of view
4. As expected, the write does not occur within the Peripheral port.
5. BRESPP is returned with the following value:
    - 2'bxx in RTL and netlist simulations
    - Any value (2’b00, 2’b01, 2’b10 or 2’b11) on silicon, depending on the state of uninitialised flip-flops.

Implications

As write accesses with WSTRBP != 4'b1111 normally return a SLVERR write response, this type of accesses are not expected to be received by the L220 Peripheral port. They are potentially expected to occur during system development for verification purposes, but not in real products.

In RTL or netlist simulations, the 2'bxx value on BRESPP can lead to X propagation issues.

On silicon, several cases are possible:

- The write response is 2'b00 (OKAY). This means that the invalid write does not generate any error even if it was invalid and did not update any register within L220. It is not expected to lead to any particular problem.

© Copyright ARM Limited 2004-2009. All rights reserved.

- The write response is 2'b01 (EXOKAY). This is not expected to lead to any particular problem as the original write access is not exclusive.
- The write response is 2'b10 (SLVERR). There is no problem in this case as 2'b10 is the expected response for the invalid write.
- The write response is 2'11 (DECERR). In this case, the response can lead to an incorrect abort case decoding in the abort handler and unnecessary resulting actions.

The erratum does not affect the security of the system. All write accesses that do not have the correct security level are properly detected in the Peripheral port and DECERR is returned.

The erratum only affects the first invalid write access after reset. All subsequent writes will have a correct SLVERR error response.

Workaround

A workaround is not expected to be necessary for this erratum. However, if a workaround appears to be mandatory, an invalid write to the Peripheral port can be issued from a known trusted software code (for example the secure reset handler) in order to clear the issue in the L220 Peripheral port.

© Copyright ARM Limited 2004-2009. All rights reserved.

533267: Potential deadlock in certain system configurations when SWP is used to launch specific maintenance operations

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Cat 3, Present in: r1p5, r1p6, r1p7. Unchanged in this document.

Description

The L220 TRM describes several system configurations for connecting L220 to an ARM processor (see sections 1.2 and 1.3). These configurations are particularly important for showing how the L220 Peripheral port is accessed. This erratum concerns the system configuration described in section 1.3 where the accesses to the Peripheral port first go through L220 and an L3 interconnect before going back to the Peripheral port. This is for example the system configuration that is usually implemented with the ARM11 MPCore processor.

A deadlock can occur if a SWP instruction is used to launch certain maintenance operations in this system configuration.

Conditions

The problem occurs when the following conditions are met:

1. L220 is implemented in a system using a similar connectivity as described in section 1.3 of the L220 TRM.
2. One of the following write accesses issued due to a SWP instruction and targeting the L220 Peripheral port is received by the one L220 AXI slave port:
   - Cache Sync operation (0x730)
   - Clean and/or Invalidate by PA operations (0x770, 0x7B0 and 0x7F0)
   - Clean and/or Invalidate by Index/Way operations (0x7B8 and 0x7FF)
3. The write access is forwarded to the one L220 AXI master port, is issued on AXI and goes through the L3 interconnect.
4. It is received by the L220 Peripheral port.

Implications

Under the conditions listed above, a deadlock occurs.

Workaround

The workaround for this erratum is not to use SWP instruction to launch maintenance operations in the system configuration where the accesses to the L220 Peripheral port are derived from an AXI interconnect connected to the L220 master port(s).

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata – Implementation

Note that the documentation erratum 391291 details the clock enable usage model in the L220 AXI slave and master interfaces. This is important from an implementation point of view.

Note also that there is an implementation workaround for the Cat 1 erratum 401552 and the Cat 3 erratum 391290.

382985: Incorrect connections within the Peripheral port IEM register slice

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Imp, Present in: r1p4, Fixed in r1p5. Unchanged in this document.

Description

The two parts of the Peripheral port IEM register slice, L220_VCORESliceP and L220_VSOCSliceP, are connected via wires going through level shifters. A wire coming from L220_VCORESliceP, resp. L220_VSOCSliceP, must have a name with the suffix “Core”, resp. “Soc”, go through the L220_LSCVCoreUpVSocP, resp. L220_LSCVSocDownVCoreP, level shifter placeholder and then have a name with the suffix “Soc”, resp. “Core”. This naming convention is not correctly implemented. The “Core” and “Soc” suffixes and the placeholders are reversed for all signals except for the following ones: SYNCMODEREQP and SYNCMODEACKP.

Implications

Due to this erratum, there is no consistency of connections between the two parts of the Peripheral port IEM register slice. This prevents from building a correct IEM implementation of L220 with the Peripheral port IEM register slice.

Workaround

There is no workaround for this erratum.

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata - Documentation

373383: TRM must state that any write to registers of ARM L220 must be preceded by an explicit cache-sync operation

Status

Affects:	product L220 Level-2 Cache Controller.
Fault status:	Present in: r1p2, Fixed in r1p4. Unchanged in this document.

Description

A statement of software integration kit README must be made clear in the TRM.

The README provided with the software integration kit contains the following statement:

```
However, writes to any of the other registers should be preceded by an explicit
cache-sync request. This is especially important when the cache is enabled and
changes to how the cache allocates new cache-lines are to be made, such as, cache
lockdown and change of debug control attributes.
```

This statement is very important and must be made visible in the L220 TRM. For example, changing the data cache lockdown register while a line is in the write allocate buffer can cause the allocation of this line in the cache to fail.

Conditions

Not Applicable.

Implications

NA.

Workaround

NA.

© Copyright ARM Limited 2004-2009. All rights reserved.

391291: TRM must precisely document the clock enable usage model implemented in the L220 AXI slave and master interfaces

Status

Affects: product L220 Level-2 Cache Controller.

Fault status: Present in: r1p4, r1p5, r1p6. Fixed in r1p7. Unchanged in this document.

Description

The AMBA AXI Protocol Specification (ARM IHI 0022B) does not specify how a clock enable must be implemented within an AXI master (or slave) when communicating with a module running synchronously at a different clock frequency. As a consequence, this must be clearly documented in the L220 Technical Reference Manual.

Find below the definition of the clock enable as used in the L220 r1p4, r1p5 and r1p6 revisions and valid for all L220 AXI interfaces (S0, S1, Peripheral, M0 and M1):

- AXI inputs are sampled on L220 CLK rising edges only when the corresponding clock enable ACLKEN* is HIGH.
- AXI outputs can be updated on any CLK rising edge with the requirement that the value is kept stable till the next CLK rising edge with the clock enable ACLKEN* HIGH.

The impact of this definition is that, for synthesis and static timing analysis, no multi-cycle path can be defined between L220 and the module to which L220 is connected via AXI. For example, there is only one CLK clock cycle to go from the register driving RREADYM0, through the AXI interconnect (if any), to the Level 3 slave sampling the ready signal, even if the slave runs at a slower ACLK clock frequency.

For more details, see the Engineering Advice Note (PR156-PRDC-007653) called “Clock Enable Usage Model in the L220 AXI interfaces”. This EAN only applies to the following revisions of L220: r1p4, r1p5 and r1p6.

In all other revisions of L220, including r1p7, the clock enable is implemented as follows:

- AXI inputs are sampled on CLK rising edges only when the corresponding clock enable ACLKEN* is HIGH.
- AXI outputs can be updated on CLK rising edges only when the corresponding clock enable ACLKEN* HIGH.

With this definition, it is possible to define multi-cycle paths between L220 and the module to which L220 is connected via AXI and which runs at a lower clock frequency.

Conditions

Not Applicable.

Implications

NA.

Workaround

NA.

© Copyright ARM Limited 2004-2009. All rights reserved.

Errata – Driver Software

There are no Errata in this Category

© Copyright ARM Limited 2004-2009. All rights reserved.