ARM Infrastructure PrimeCell
AXI-AHB Bridges (BP137)
Errata Notice

This document contains all errata known at the date of issue in releases up to and including revision r0p1 of AXIToAHB

Proprietary notice

Document confidentiality status

This document is Non Confidential.

Web address

http://www.arm.com/

Feedback on the product

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

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

Feedback on this document

If you have any comments on about this document, please send email to mailto: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.

Contents

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

05 Oct 2005: Changes in Document v2

Page	Status	ID	Cat	Summary
10	New	337799	Cat 3	OVLs required for unsupported TrustZone accesses
11	New	339001	Cat 3	HBURST encoding mismatch between RTL and DM
12	New	347897	Cat 3	Combinatorial path from HREADY to HTRANS on AxiToAhbLiteM.v
13	New	340289	Doc	Undocumented AHB protocol violation with certain combinations of sparse and aligned AXI transfers

16 Dec 2004: Changes in Document v1.0

No Errata in this document revision

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	r0p0-00REL0	r0p1-00REL0
337799	Cat 3	OVLs required for unsupported TrustZone accesses	X	
339001	Cat 3	HBURST encoding mismatch between RTL and DM	X	
347897	Cat 3	Combinatorial path from HREADY to HTRANS on AxiToAhbLiteM.v	X	
340289	Doc	Undocumented AHB protocol violation with certain combinations of sparse and aligned AXI transfers	X	

Errata - Category 1

There are no Errata in this Category

Errata - Category 2

There are no Errata in this Category

Errata - Category 3

337799: OVLs required for unsupported TrustZone accesses

Status

Affects: product AxiToAhb.

Fault status: Cat 3, Present in: r0p0-00REL0, Fixed in r0p1-00REL0. New in this document.

Description

TrustZone security extensions are not supported by any version AHB, and so an OVL assertion is included within AxiToAhbLiteMOVL.v to warn if an incoming AXI transaction has the security bit (ARPROT[1] or AWPROT[1]) asserted. The remaining versions of the bridge (AxiToA11AhbLite.v, AxiToAhbLiteM.v, AxiToAhbLiteS.v) should include the OVL assertion, but do not.

Additionally, the HPROT[6] output from AxiToA11AhbLite.v should be ignored - this is not part of the ARM11 AHB specification, and will be removed from future versions of the bridge.

Implications

Since TrustZone is not supported by AHB, the HPROT[6] bit should not be used in any existing systems, and any system which includes TrustZone security considerations should be designed such that security barriers are implemented in the AXI domain. Therefore no adverse implications are expected.

Workaround

If you want your simulations of AXI to AHB bridges other than AxiToAhbLiteMOVL to include a warning when a non-secure AXI transaction is received, then move the OVL assertion from AxiToAhbLiteMOVL.v to AxiToA11AhbLite.v

339001: HBURST encoding mismatch between RTL and DM

Status

Affects: product AxiToAhb.

Fault status: Cat 3, Present in: r0p0-00REL0, Fixed in r0p1-00REL0. New in this document.

Description

The AXI to AHB "Technical Overview" and "Deign Manual" documents state that: "incrementing bursts of length 1, wrapping bursts of length 2, and all fixed address transactions are converted to sequences of AHB SINGLE bursts"

The RTL implementation instead converts incrementing bursts of length 1 and wrapping bursts of length 2 into sequences of AHB bursts of type INCR and length 1.

Implications

The generated AHB bursts of type INCR are fully compliant with the AHB specification, and so do not cause any adverse effects in a system.

Workaround

None required

347897: Combinatorial path from HREADY to HTRANS on AxiToAhbLiteM.v

Status

Affects: product AxiToAhb.

Fault status: Cat 3, Present in: r0p0-00REL0, Fixed in r0p1-00REL0. New in this document.

Description

The AxiToAhbHtransSquelch block is used to combine the HREADYM, HSELM and HTRANSM outputs from the slave-gasket port of ExAcMnAhb into a single HTRANS output suitable for an AHB-Lite master port. This is required because HSEL might initially go high whilst HREADY (from a previous slave) is low - but that HREADY value does not propagate to the output master port and so HTRANS is squelched.

A feature of the ValidReg generation inside AxiToAhbHtransSquelch means that ValidReg is not held if the slave responds with wait states to the final transfer (indicated by the final cycle with HSEL high). If HSEL is re-asserted during these wait states then it is the HREADY from the slave which is used to enable the HTRANS output. This is undesirable because it causes a combinatorial timing path from an input to an output on an AMBA interface.

Implications

The presence of a combinatorial timing path from the HREADY input to the HTRANS output of AxiToAhbLiteM.v may cause problems when trying to achieve timing closure for a system containing this component.

Please note that this defect does not cause any functional errors, only timing is affected.

Workaround

If timing closure is proving difficult in a system design which instantiates this component, then the generation of ValidReg within AxiToAhbHtransSquelch can be altered by adding HREADYS as an enable term to the inferred register as follows:

(lines 112-119 of AxiToAhbHtransSquelch.v)

```
 // Valid register
 always @ (negedge HRESETn or posedge HCLK)
   begin : p_ValidReg
     if (!HRESETn)
       ValidReg <= 1'b0;
     else if (HREADYS)
       ValidReg <= Valid;
   end
```

Errata - Documentation

340289: Undocumented AHB protocol violation with certain combinations of sparse and aligned AXI transfers

Status

Affects: product AxiToAhb.

Fault status: Doc, Present in: r0p0-00REL0, Fixed in r0p1-00REL0. New in this document.

Description

This errata is issued to clarify a limitation of the PrimeCell® Infrastructure AMBA™ 3 AXI™ to AMBA 3 AHB™ Bridges (BP137) as documented in Revision r0p0 of the Design Manual.

The issue is that the AXI to ARM11 AHB-Lite bridge does not fully support sparse write bursts, i.e. where some bits of WSTRB are de-asserted in order to prevent the writing of certain bytes. The Design Manual correctly refers to a potential protocol violation on the output signal IHUNALIGN of this subcomponent of the AXI to AHB bridges. However, it omits to describe a possible consequent protocol violation on the AHB signals HTRANS, HBURST, and HSIZE at the outputs of the AxiToAhbLiteM or AxiToAhbLiteS variants of the AXI to AHB bridges.

The issue arises when an AXI write transaction begins with an aligned, non-sparse write transfer but one or more subsequent write transfers in the same transaction are sparse. This causes the internal signal IHUNALIGN to transition from low to high at the output of the module AxiToA11AhbLite, as described in the Design Manual. When IHUNALIGN goes high, the byte lane strobe converter (BLSCnv) converts each data transfer into series of byte transfers and the outputs HBURST, HTRANS, and HSIZE of BLScnv change. This violates the AHB protocol because the AHB transaction is mid-burst.

For example:

- A NONSEQ, SIZE32, INCR8, write transfer with an aligned address and full byte strobes on the first transfer at the input to BLScnv results in a NONSEQ, INCR8, SIZE32 transfer at the output.

- A second write transfer, SEQ, SIZE32, INCR8, with sparse byte strobes at the input to BLScnv results in a NONSEQ, INCR, SIZE8 transfer at the output.

This breaks the AHB protocol, because HTRANS must not change from SEQ to NONSEQ, HBURST must not change from INCR8 to INCR, and HSIZE must not change from SIZE32 to SIZE8 until the first burst completes or is cancelled due to an error response.

Implications

If the bridge is used with a master that can generate the type of write transaction described, then the break of AHB protocol may cause a slave to exhibit UNPREDICTABLE behavior. Note that the ARM PrimeCell L220 Level 2 Cache Controller (AC131) can cause this problem. The workaround described below should be employed in this case.

If the AHB slave uses the burst information, there may be a performance cost for write transactions if the workaround is implemented. This is dependant on the particular AHB slave. Many AHB slaves do not make use of the burst information and there would be no performance cost in this case.

Workaround

You should either ensure that the AXI master does not generate write transactions that begin with an aligned, non-sparse write transfer but subsequently include a sparse write transfer, or use the workaround documented in Section 2.2.12 of the Design Manual:

“Every AXI write burst must be converted into a string of single, length = 1, AHB bursts by forcing HBURST to SINGLE, 0b000, or INCR, 0b001, and forcing HTRANS[0] LOW.”

In AxiToAhbLiteM and AxiToAhbLiteS, override the HBURST input to the module uBLScnv to SINGLE (3’b000) or INCR (3’b001), and override the HTRANS[0] input to the module uBLScnv to LOW(1’b0) during write bursts. This is shown in Figure 1 below:

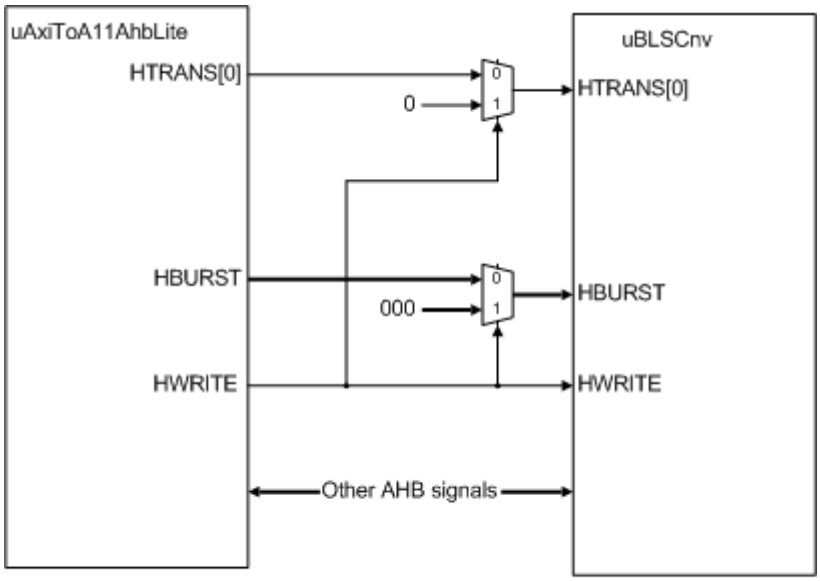

Figure 1 Override HTRANS[0] and HBURST to uBLSCnv for write bursts

A simpler version of this workaround is to ignore the direction of the burst and permanently override the HBURST and HTRANS[0] inputs to the module uBLScnv. This has the disadvantage of potentially impairing performance during read bursts. This is shown in Figure 2 below:

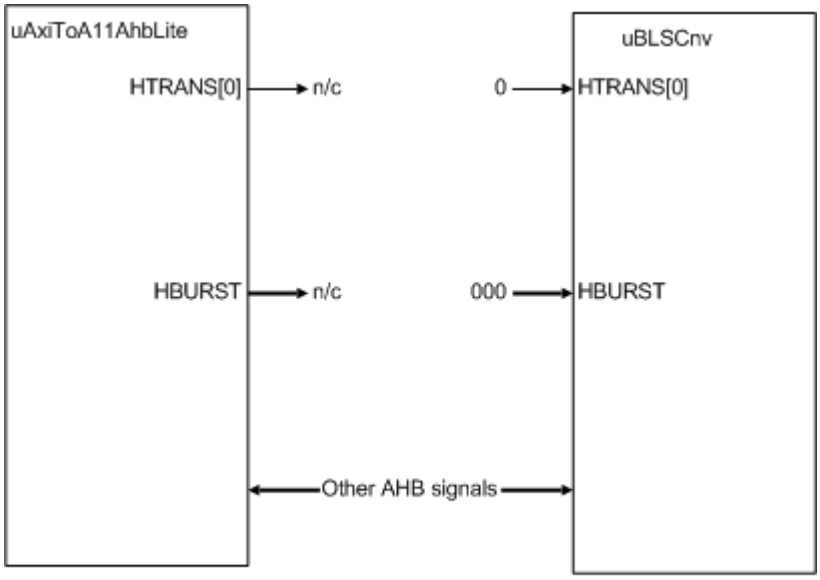

Figure 2 Permanently override HTRANS[0] and HBURST to uBLSCnv

Errata – Driver Software

There are no Errata in this Category