ARM® CoreLink™ AXI4 to AHB-Lite XHB-400 Bridge

Revision: r0p0

Technical Reference Manual

ARM®

ARM DDI 0523A (ID020315)

ARM CoreLink AXI4 to AHB-Lite XHB-400 Bridge
Technical Reference Manual

Release Information

The *Change history* table lists the changes made to this book.

Change history

Date	Issue	Confidentiality	Change
11 December 2014	A	Non-Confidential	First release for r0p0

Proprietary Notice

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

Contents
ARM CoreLink AXI4 to AHB-Lite XHB-400 Bridge Technical Reference Manual

Preface

This preface introduces the CoreLink AXI4 to AHB-Lite XHB-400 Bridge *Technical Reference Manual* (TRM). It contains the following sections:

About this book

This book describes the technical features of the XHB-400.

Product revision status

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

r*n* Identifies the major revision of the product.

p*n* Identifies the minor revision or modification status of the product.

Intended audience

This book is written for system designers who are designing a System-on-Chip that uses the XHB-400.

Using this book

This book is organized into the following chapters:

Chapter 1 *Introduction*

Read this for an introduction to the XHB-400 and its features.

Chapter 2 *Functional Description*

Read this for a description of the major interfaces and components of the XHB-400. This chapter also describes how the components operate.

Chapter 3 *Programmers Model*

Read this for a description of the programmers model.

Appendix A *Signal Descriptions*

Read this for a description of the XHB-400 signals.

Appendix B *Revisions*

Read this for a description of the technical changes between released issues of this book.

Glossary

The *ARM® Glossary* is a list of terms used in ARM documentation, together with definitions for those terms. The *ARM® Glossary* does not contain terms that are industry standard unless the ARM meaning differs from the generally accepted meaning.

See the *ARM® Glossary*
http://infocenter.arm.com/help/topic/com.arm.doc.aeg0014-/index.html.

Conventions

This book uses the conventions that are described in:

- *Typographical conventions* on page vi.
- *Timing diagrams* on page vi.
- *Signals* on page vii.

Typographical conventions

The following table describes the typographical conventions.

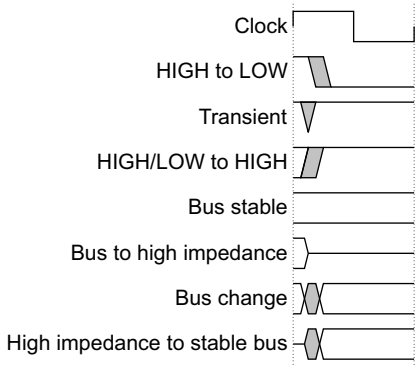
Typographical conventions

Style	Purpose
italic	Introduces special terminology, denotes cross-references, and citations.
bold	Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.
`monospace`	Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.
`mono`space	Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
`monospace italic`	Denotes arguments to monospace text where the argument is to be replaced by a specific value.
`monospace bold`	Denotes language keywords when used outside example code.
<and>	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0 <Rd>, <CRn>, <CRm>, <Opcode_2>`
SMALL CAPITALS	Used in body text for a few terms that have specific technical meanings, that are defined in the *ARM® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Timing diagrams

The figure *Key to timing diagram conventions* explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are UNDEFINED, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.



Key to timing diagram conventions

Timing diagrams sometimes show single-bit signals as HIGH and LOW at the same time and they look similar to the bus change shown in *Key to timing diagram conventions*. If a timing diagram shows a single-bit signal in this way then its value does not affect the accompanying description.

Signals

The signal conventions are:

Signal level The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:
- HIGH for active-HIGH signals.
- LOW for active-LOW signals.

Lower-case n At the start or end of a signal name denotes an active-LOW signal.

Additional reading

This section lists publications by ARM and by third parties.

See Infocenter http://infocenter.arm.com for access to ARM documentation.

ARM publications

This book contains information that is specific to this product. See the following documents for other relevant information:

- *ARM® CoreLink™ AXI4 to AHB-Lite XHB-400 Bridge Integration and Implementation Manual* (ARM DIT 0061).
- *ARM® AMBA® AXI and ACE Protocol Specification* (ARM IHI 0022).
- *ARM® AMBA® 3 AHB-Lite Protocol Specification* (ARM IHI 0033).

Feedback

ARM welcomes feedback on this product and its documentation.

Feedback on this product

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

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

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

- The title.
- The number, ARM DDI 0523A.
- The page numbers to which your comments apply.
- A concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Note

ARM tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Chapter 1
Introduction

This chapter introduces the XHB-400 and its features. It contains the following sections:

1.1 About the XHB-400

The XHB converts AXI4 protocol to AHB-Lite protocol and has an AXI4 slave interface and an AHB-Lite master interface. For information about how AXI4 transactions to AHB-Lite are bridged by the XHB, see Table 2-1 on page 2-2.

AXI4 slave interface

This connects to either the AXI4 master interface of a processor or to an AXI interconnect.

AHB-Lite master interface

This implements an AHB-Lite master to drive AHB-Lite subsystems.

Figure 1-1 shows some example systems that include the XHB.

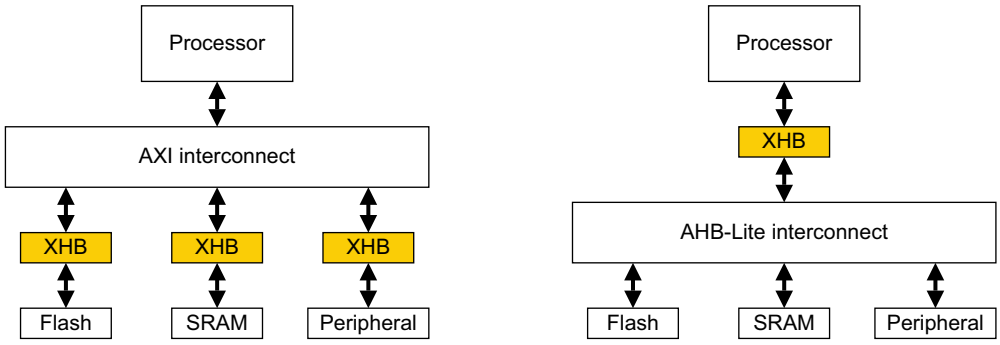


Figure 1-1 Example XHB system diagrams

1.1.1 XHB features

The main XHB features are:

- Full support of the AXI4 protocol.
- Efficient conversion of the AXI4 transactions to AHB-Lite.
- Conversion of sparse write transactions to AHB-Lite.
- Read acceptance capability is 2 transactions.
- Write acceptance capability is 2 transactions.
- Combined acceptance capability is 4 transactions.
- Configurable data width options.
- Zero latency conversion to AHB-Lite.
- Two entry FIFOs for buffering read data and write response channels.
- Processor performance improved by prioritizing read transactions over write transactions, when AXI read and write address channels are valid in the same cycle. Also read transactions might interrupt sparse write transactions. Write transactions are guaranteed to be accepted every 1 in 8 transactions when back-to-back read transactions occur.
- Support of exclusive accesses on the AHB-Lite interface using the **EXREQ** and **EXRESP** signals.

1.2 Compliance

The XHB-400 complies with the following protocols:

- AMBA 4 AXI4 protocol. See the *ARM® AMBA® AXI and ACE Protocol Specification*.
- AMBA 3 AHB-Lite protocol. See the *ARM® AMBA® 3 AHB-Lite Protocol Specification*.

This document complements the protocol specifications. It does not duplicate information from these sources.

1.3 Configurable options

The XHB provides a configurable data bus width option. For information about how to change the configuration and value ranges, see the *ARM® CoreLink™ AXI4 to AHB-Lite XHB-400 Bridge Integration and Implementation Manual*.

1.4 Product documentation and design flow

This section describes the XHB-400 books and how they relate to the design flow. It includes:

- *Documentation*.
- *Design flow*.

See *Additional reading* on page vii for more information about the books described in this section. For information about the relevant architectural standards and protocols, see *Compliance* on page 1-3.

1.4.1 Documentation

The XHB documentation is as follows:

Technical Reference Manual

The *Technical Reference Manual* (TRM) describes the functionality and the effects of functional options on the behavior of the XHB. It is required at all stages of the design flow. The choices made in the design flow can mean that some behavior described in the TRM is not relevant.

Integration and Implementation Manual

The *Implementation and Integration Manual* (IIM) describes:

- The available build configuration options and related issues in selecting them.
- How to configure the RTL code with the build configuration options.
- How to integrate the XHB into a SoC. This includes describing the signals that the integrator must tie off to configure the macrocell for the required integration.
- The processes to sign off the integration and implementation of the design.

The ARM product deliverables include reference scripts and information about using them to implement your design. Reference methodology flows supplied by ARM are example reference implementations. Contact your EDA vendor for EDA tool support.

Reference methodology documentation from your EDA tools vendor complements the IIM.

The IIM is a confidential book that is only available to licensees.

1.4.2 Design flow

The XHB is delivered as synthesizable RTL. Before it can be used in a product, the integrator must configure the RTL and then connect the design into a SoC.

Note

This manual refers to implementation-defined features that are applicable to build configuration options. Reference to a feature that is included means that the appropriate build configuration options are selected.

1.5 Product revisions

This section describes the differences in functionality between product revisions:

r0p0 First release.

Chapter 2
Functional Description

This chapter describes the major logic blocks and operation of the XHB-400. It contains the following sections:

2.1 XHB behavior

This section describes the behavior of the XHB and how AXI4 transactions convert to AHB-Lite transfers. It contains the following sections:

- *Burst conversions*.
- *1KB boundary crossing*.
- *Protection control* on page 2-3.
- *Exclusive accesses* on page 2-3.
- *Address alignment* on page 2-4.
- *Sparse write byte-strobes* on page 2-4.
- *User sideband signals* on page 2-6.

2.1.1 Burst conversions

Table 2-1 shows the mapping of AXI4 burst types to AHB-Lite burst types.

Table 2-1 AXI4 burst type to AHB-Lite burst type mapping

AxBURST	Number of transfers in AXI4 transaction	HBURST	Notes
FIXED	1-16	SINGLE	The number of SINGLE bursts depends on the **AxLEN** value
INCR	1	SINGLE	-
	4	INCR4	-
	8	INCR8	-
	16	INCR16	-
	2, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 17-256	INCR	Undefined length
WRAP	2	SINGLE	Two transfers
	4	WRAP4	-
	8	WRAP8	-
	16	WRAP16	-

2.1.2 1KB boundary crossing

The AHB-Lite protocol requires that bursts do not cross 1KB boundaries. Since AXI4 transactions can cross a 1KB boundary, the XHB converts the AHB-Lite bursts as follows:

- For AXI transactions that convert to INCR4, INCR8, or INCR16 bursts, the XHB changes the INCRx burst that crosses the boundary to a series of SINGLE AHB-Lite bursts.
- For AXI transactions that convert to undefined length INCR bursts, the XHB sets **HTRANS** to NONSEQ on the first address after a 1KB boundary is crossed.

Note

To simplify the logic, the conversion to SINGLE bursts can occur for bursts that are close to a 1KB boundary but do not actually cross it.

2.1.3 Protection control

Table 2-2 shows how the protection control information passes from AXI4 to AHB-Lite.

Table 2-2 Protection control mapping

Description	AXI4 signal	AHB-Lite signal
Modifiable (AXI4), Cacheable (AHB-Lite)	**AxCACHE[1]**	**HPROT[3]**
Bufferable	**AxCACHE[0]**	**HPROT[2]**
Privileged	**AxPROT[0]**	**HPROT[1]**
Data or instruction access	**AxPROT[2]**	**HPROT[0]** **Note** **HPROT[0]** is the inverse of **AxPROT[2]**.

2.1.4 Exclusive accesses

The XHB AHB-Lite master interface supports exclusive accesses using the **EXREQ** and **EXRESP** signals. **EXREQ** is an address phase signal and **EXRESP** is a data phase signal. The XHB asserts **EXREQ** on every AHB-Lite transfer that derives from an AXI4 transaction with **AxLOCK** asserted.

Table 2-3 shows the mapping of the **EXRESP** signal to the AXI4 **RRESP** and **BRESP** signals.

Table 2-3 Mapping exclusive response

EXRESP	RRESP[0], BRESP[0]	Description
HIGH	OKAY	Indicates exclusive access has failed
LOW	EXOKAY	Indicates exclusive access has been successful

For an exclusive read the **EXRESP** signal must be:

LOW If a system monitor is implemented that covers the access address.

HIGH If a system monitor is not implemented that covers the access address.

For an exclusive write the **EXRESP** signal must be:

LOW If a system monitor is implemented that covers the access address and the exclusive check passes.

HIGH If a system monitor is implemented that covers the access address and the exclusive check fails, or a system monitor is not implemented that covers the access address.

If **EXRESP** is HIGH on any beat of an exclusive write transaction, then the XHB sets **BRESP[0]** LOW to indicate that the exclusive access failed.

If **HRESP** is HIGH, indicating an access error, this value overrides the **EXRESP** value and the XHB returns SLVERR to the AXI master on the appropriate AXI4 **xRESP** signal.

Note

- AXI4 exclusive write transactions must not have sparse write strobes. If an AXI4 exclusive write transaction contains sparse write strobes, then the XHB generates the AHB-Lite transfers but it also returns SLVERR on **BRESP[1:0]**.

- The **AWSPARSE** signal is ignored when **AWLOCK** is asserted. See *Sparse write byte-strobes*.
- The XHB does not generate locked AHB-Lite transactions so **HMASTLOCK** is always LOW.

For more information about the **EXREQ** and **EXRESP** signals, see Table A-2 on page A-4.

2.1.5 Address alignment

The AHB-Lite protocol does not support unaligned transfers so the XHB performs address alignment.

For read transactions, if the XHB receives an unaligned **ARADDR** value, then it aligns the address to the **ARSIZE** value before it generates the AHB-Lite transfer.

For write transactions, if the XHB receives an unaligned **AWADDR** value when:

- **AWSPARSE** is not asserted, then it aligns the address to the **AWSIZE** value before it generates the AHB-Lite transfer. This system error might corrupt the memory locations that are accessed so the XHB returns SLVERR on **BRESP[1:0]**.
- **AWSPARSE** is asserted, then the XHB processes the transaction as *Sparse write byte-strobes* describes.

2.1.6 Sparse write byte-strobes

The XHB has the **AWSPARSE** signal to support efficient conversion of write transactions containing sparse write byte-strobes. This signal is an extension to the *ARM® AMBA® AXI and ACE Protocol*. To transfer the **AWSPARSE** signal through an AXI interconnect you can use an **AWUSER** signal bit.

A master must assert **AWSPARSE** if a write transaction contains sparse write byte-strobes or if the transaction is unaligned. This causes the **HBURST** mappings that Table 2-1 on page 2-2 shows to be overridden to undefined length INCR. The XHB decodes the **WSTRB** signal to generate the least number of AHB-Lite transfers possible that correspond to the strobe bits set. If the write strobe bits set match the size of the AXI4 transaction, then only one AHB-Lite transfer is generated for the AXI4 beat.

Normally, for a sparse write, each AHB-Lite transfer has an **HTRANS** value of NONSEQ. When the current and the previous beat of an AXI4 transaction do not have sparse write strobes and **AWBURST** is INCR, then **HTRANS** is SEQ for the current beat. This means that if a transaction is marked as sparse pessimistically, and the **WSTRB** values are not sparse, the transaction is converted to a sequential burst of the same length on the AHB-Lite interface with little or no impact on performance.

Note

If **AWSPARSE** is not asserted for a write transaction that has a beat with a sparse **WSTRB** value or the write transaction is unaligned, then the XHB sets **BRESP** to SLVERR. The XHB converts the transaction to AHB-Lite transfers, as shown in Table 2-1 on page 2-2, which might corrupt the memory locations that are accessed.

Figure 2-1 on page 2-5 shows an example of how the XHB-400 processes an AXI sparse write transaction when **AWSPARSE** is HIGH.

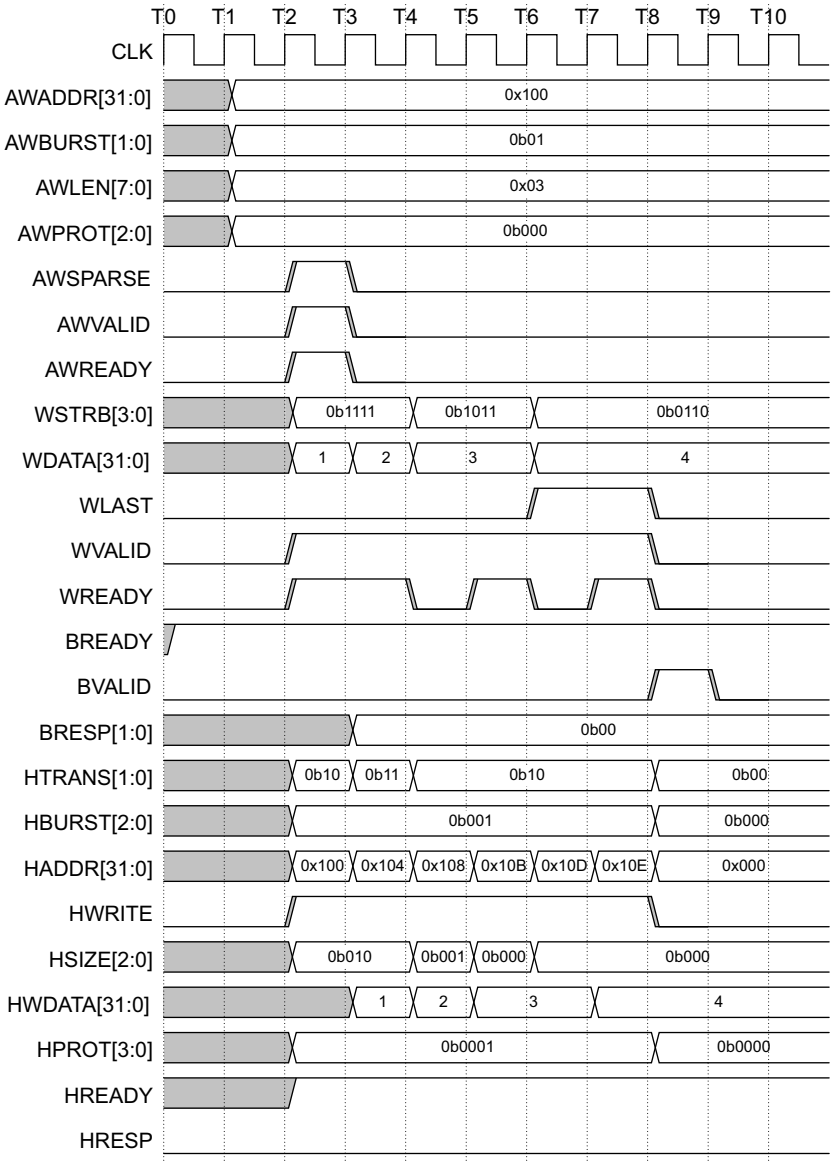


Figure 2-1 Sparse write byte-strobe timing example

In Figure 2-1, at time:

T3 The XHB samples the arrival of a new write transaction.

T4 **WSTRB[2]** goes LOW so the XHB splits the word transfer into:

- A halfword transfer (**HSIZE**=`0b001`) to address `0x108`.
- A byte transfer (**HSIZE**=`0b000`) to address `0x10B`, at time T5.

T6 **WSTRB** changes to `0b0110` so only data bits[23:8] are valid. Therefore, to maintain the alignment requirements of AHB-Lite, the XHB performs:

- A byte transfer (**HSIZE**=`0b000`) to address `0x10D`.
- A byte transfer (**HSIZE**=`0b000`) to address `0x10E`, at time T7.

T8 The XHB returns a single write response, to complete the AXI transaction.

2.1.7 User sideband signals

The XHB supports four AXI4 sideband signals and three AHB-Lite sideband signals. Table 2-4 shows how these signals are mapped between the interfaces.

Table 2-4 User sideband signals mapping

AXI4		AHB-Lite		Phase	Comment
Signal	**Direction**	**Signal**	**Direction**		
ARUSER	Input	**HAUSER**	Output	Address	Read transfers. The same value is output on each transfer associated with an AXI4 transaction.
AWUSER					Write transfers. The same value is output on each transfer associated with an AXI4 transaction.
WUSER	Input	**HWUSER**	Output	Data	Only valid on write transfers. The value might be different on each transfer except for sparse writes where the same value is output for transfers associated with the same AXI4 beat.
RUSER	Output	**HRUSER**	Input		Only valid on read transfers. The value might be different on each transfer.

Note

The XHB does not provide a **BUSER** signal because the AHB-Lite protocol does not have a write response user signal.

2.2 Interfaces

Figure 2-2 shows the XHB-400 external interfaces.

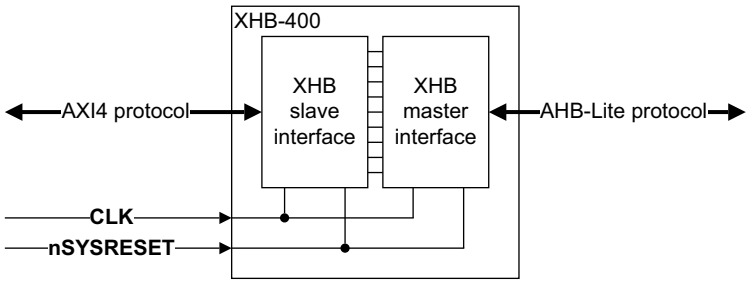


Figure 2-2 XHB-400 interfaces

The XHB-400 interfaces are:

- AMBA protocols that include some additional extension signals. See *AXI4 slave interface signals* on page A-2 and *AHB-Lite master interface signals* on page A-4.
- Signals whose requirements are linked:
  - Clock signal. See *Clocks* on page 2-8.
  - Reset signal. See *Resets* on page 2-9.

2.3 Clocks

The XHB-400 has a single clock domain that is driven by the **CLK** signal.

2.4 Resets

The XHB-400 has a single active-LOW reset, **nSYSRESET**.

Chapter 3
Programmers Model

This chapter describes the programmers model. It contains the following section:

- *About the programmers model* on page 3-2.

3.1 About the programmers model

The XHB-400 has no programmable registers so the XHB is transparent to a programmer.

Appendix A
Signal Descriptions

This appendix describes the signals that the XHB-400 provides. It contains the following sections:

A.1 AXI4 slave interface signals

The AXI4 slave interface does not use the following signals:

- **AxQOS**.
- **AxREGION**.
- **BUSER**.
- **CSYSREQ**.
- **CSYSACK**.
- **CACTIVE**.

Table A-1 shows the AXI4 slave interface signals.

Table A-1 AXI4 slave interface signals

Signal name	Direction	Description
AW channel signals:		
AWADDR[31:0]	Input	See the *ARM® AMBA® AXI and ACE Protocol Specification*
AWBURST[1:0]		
AWCACHE[3:0]		
AWID[15:0]		
AWLEN[7:0]		
AWLOCK		
AWPROT[2:0]		
AWREADY	Output	
AWSIZE[2:0]	Input	
AWUSER[15:0]		
AWVALID		
AWSPARSE		Indicates that a transaction might use sparse writes strobes, see *Sparse write byte-strobes* on page 2-4. This signal is an extension to the AXI protocol.
AR channel signals:		

Table A-1 AXI4 slave interface signals (continued)

Signal name	Direction	Description
ARADDR[31:0]	Input	See the *ARM® AMBA® AXI and ACE Protocol Specification*
ARBURST[1:0]		
ARCACHE[3:0]		
ARID[15:0]		
ARLEN[7:0]		
ARLOCK		
ARPROT[2:0]		
ARREADY	Output	
ARSIZE[2:0]	Input	
ARUSER[15:0]		
ARVALID		
W channel signals:		
WDATA[DATA_WIDTH–1:0][a]	Input	See the *ARM® AMBA® AXI and ACE Protocol Specification*
WLAST		
WREADY	Output	
WSTRB[(DATA_WIDTH/8)–1:0][a]	Input	
WUSER[15:0]		
WVALID		
R channel signals:		
RDATA[DATA_WIDTH–1:0][a]	Output	See the *ARM® AMBA® AXI and ACE Protocol Specification*
RID[15:0]		
RLAST		
RREADY	Input	
RRESP[1:0]	Output	
RUSER[15:0]		
RVALID		
B channel signals:		
BID[15:0]	Output	See the *ARM® AMBA® AXI and ACE Protocol Specification*
BREADY	Input	
BRESP[1:0]	Output	
BVALID		

a. The signal width is configurable. See the *ARM® CoreLink™ AXI4 to AHB-Lite XHB-400 Bridge Integration and Implementation Manual* for information about the signal widths that the XHB supports.

A.2 AHB-Lite master interface signals

Table A-2 shows the AHB-Lite master interface signals.

Table A-2 AHB-Lite master interface signals

Signal name	Direction	Description
HADDR[31:0]	Output	See the *ARM® AMBA® 3 AHB-Lite Protocol Specification*.
HBURST[2:0]		
HMASTLOCK		
HPROT[3:0]		
HRDATA[DATA_WIDTH–1:0][a]	Input	
HREADY		
HRESP		
HSIZE[2:0]	Output	
HTRANS[1:0]		
HWDATA[DATA_WIDTH–1:0][a]		
HWRITE		
Exclusive access signals:		
EXREQ[b]	Output	This is an address phase signal that indicates if a transfer is part of an exclusive transaction: **LOW** Non-exclusive transaction. **HIGH** Exclusive transaction. During IDLE transfers **EXREQ** is LOW.
EXRESP[b]	Input	This is a data phase signal. It indicates whether the exclusive request was granted or failed: **LOW** Exclusive access granted. **HIGH** Exclusive access failed. **EXRESP** is only valid when **EXREQ** is HIGH in the address phase.
User sideband signals:		
HAUSER[15:0][c]	Output	This is an address phase signal.
HRUSER[15:0][c]	Input	This is a data phase signal that is only valid for read transfers when **HREADY** is HIGH.
HWUSER[15:0][c]	Output	This is a data phase signal that is only valid for write transfers.

a. The signal width is configurable. See the *ARM® CoreLink™ AXI4 to AHB-Lite XHB-400 Bridge Integration and Implementation Manual* for information about signal widths that the XHB supports.

b. Exclusive access signals are extensions to the AHB-Lite protocol. See *Exclusive accesses* on page 2-3.

c. User sideband signals are extensions to the AHB-Lite protocol. See *User sideband signals* on page 2-6.

Appendix B
Revisions

This appendix describes the technical changes between released issues of this book.

Table B-1 Issue A

Change	Location	Affects
First release	-	-