LogiCORE IP AXI Master Lite (axi_master_lite) (v1.00a)

DS836 March 1, 2011 **Product Specification**

Introduction

The AXI Master Lite is an AXI4-compatible LogiCORE™ IP product. It provides an interface between a user-created IP core and an AXI4-Lite interface. The AXI4-Lite Master IP supports AXI4-Lite compatible bus mastering operations which are single 32-bit wide read or write data transfers.

Features

- AXI4-Lite Master interface
  - Fixed 32-bit data width
  - Supports single beat read and write data transfers of up to 4 bytes (32-bits)
- IPIC back-end interface for PLBV46 Master Single migration

LogiCORE IP Facts Table					
Core Specifics					
Supported Device Family[1]	Virtex-6, Spartan-6				
Supported User Interfaces	AXI4-Lite, IPIC Master				
	Resources				
	LUTs	**FFs**	**DSP Slices**	**Block RAMs**	**Frequency**
	See Table 5 and Table 6.				
Provided with Core					
Documentation	Product Specification				
Design Files	VHDL				
Example Design	Not Provided				
Test Bench	Not Provided				
Constraints File	Not Provided				
Simulation Model	Not Provided				
Tested Design Tools					
Design Entry Tools	EDK 13.1, XPS				
Simulation	Mentor Graphics ModelSim v6.6d				
Synthesis Tools	XST				
Support					
Provided by Xilinx, Inc.					

1. For a complete listing of supported devices, see the release notes for this core.

Applications

The AXI Master Lite provides a AXI4-Lite mastering capability that has the legacy IPIC User interface suitable for updating to AXI4 those legacy plbv46 designs that used the plbv46_master_single module.

The backend IPIC interface connects directly to the legacy User IPIC interface of the plbv46_master_single without modification. However, the plbv46 signal set has been replaced with AXI4-Lite master signal set. To migrate from a plbv46_master_single core:

- HDL Changes
  - Modify the instantiation to use the axi_master_lite core.
  - Replace the PLB ports with AXI4-Lite ports.
- PAO File Changes
  - Change the PAO file instantiation to use the axi_master_lite library.
- MPD File Changes
  - Replace the PLB ports with AXI4-Lite ports
  - Add AXI Interconnect related parameters. See DS768, *LogiCORE AXI Interconnect IP Data Sheet*, for more information.

Functional Description

'The AXI Master Lite provides a quick way to implement a light-weight mastering interface between user logic and an AXI4-Lite interface. Figure 1 is a block diagram of the AXI Master Lite. The port references and groupings are detailed in Table 1. The design is natively 32 bits and supports read and write transfers of 1 to 4 bytes. Transfer request protocol between the AXI4 and the User Logic is provided by the Read and Write Controller block.

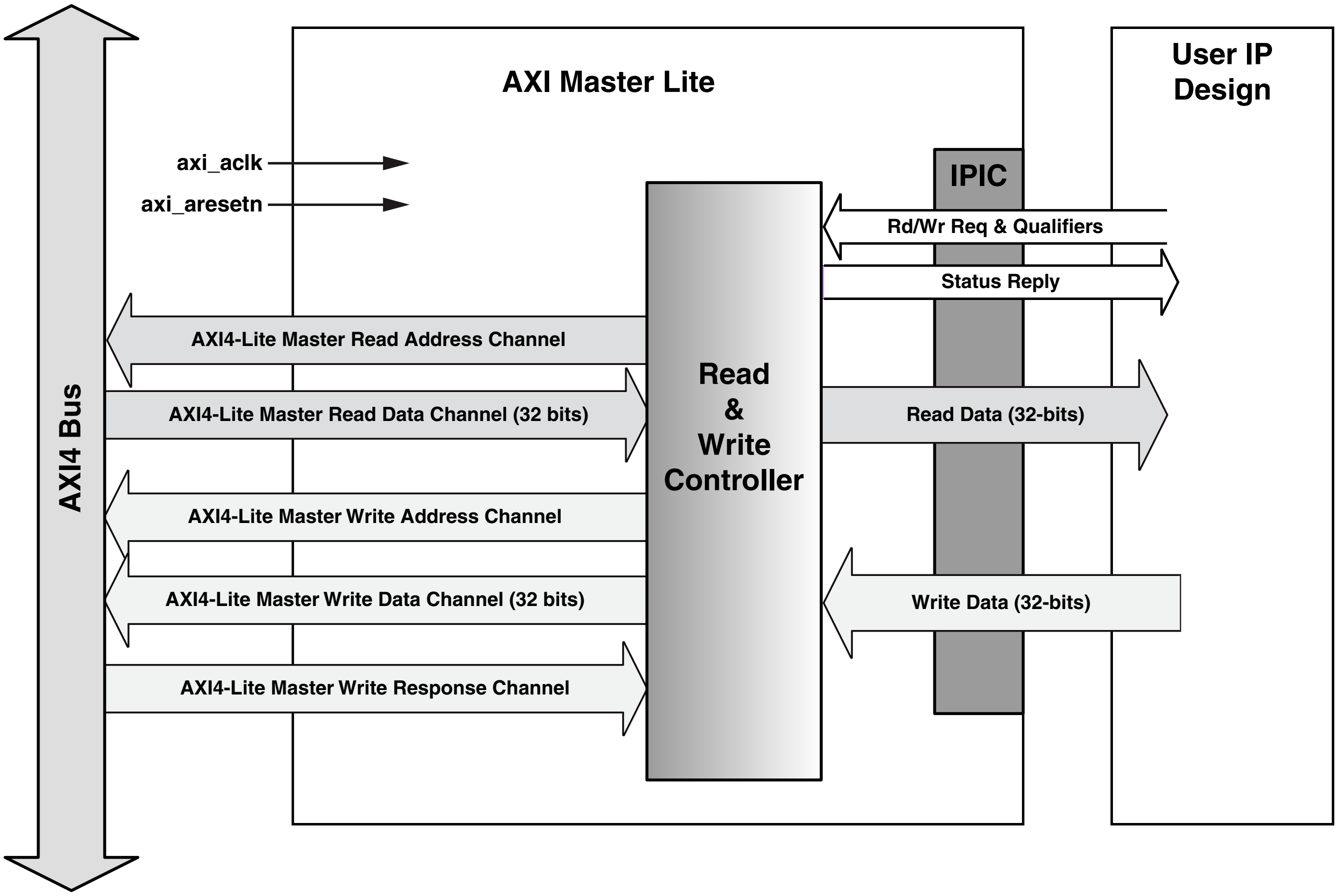
AXI4 Bus
AXI Master Lite
User IP Design
axi_aclk
axi_aresetn
IPIC
Rd/Wr Req & Qualifiers
Status Reply
AXI4-Lite Master Read Address Channel
AXI4-Lite Master Read Data Channel (32 bits)
Read & Write Controller
Read Data (32-bits)
AXI4-Lite Master Write Address Channel
AXI4-Lite Master Write Data Channel (32 bits)
Write Data (32-bits)
AXI4-Lite Master Write Response Channel

Figure 1: **AXI Master Lite Block Diagram**

Typical System Interconnect

A typical use case of the AXI Master Lite is shown in Figure 2. The AXI Interconnect Block allows the User IP to access AXI4 slaves (AXI4 and AXI4-Lite) via the AXI4-Lite interface.

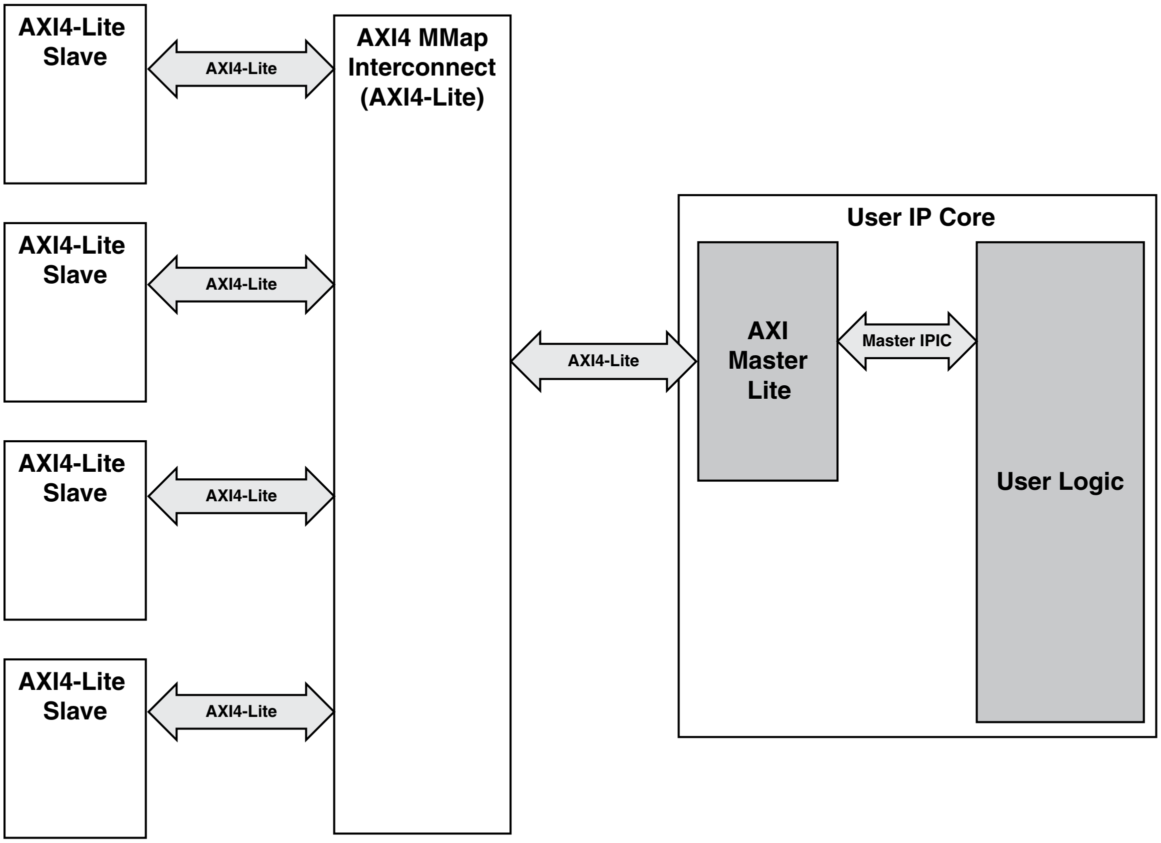

Figure 2: **AXI Interconnect Block (AXI_Interconnect)**

I/O Signals

The AXI Master Lite signals are described in Table 1.

Table 1: **AXI Master Lite I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
System Signals				
m_axi_lite_aclk	Clock	I		AXI Master Lite synchronization Clock.
m_axi_lite_aresetn	Reset	I		AXI Master Lite Reset. When asserted low, the AXI Master Lite core is put into hard reset. This signal must be synchronous to m_axi_aclk.
Master Detected Error Discrete				
md_error	Discrete Out	O		Master Detected Error. Active high master detected error output discrete. This bit is sticky once set and is only cleared by a hardware reset.
AXI4-Lite Master Read Address Channel				
m_axi_lite_araddr (C_M_AXI_LITE_ADDR_WIDTH-1: 0)	M_AXI_LITE	O	zeros	AXI Master Lite Read Address Channel Address Bus.
m_axi_lite_arprot(2:0)	M_AXI_LITE	O	000b	AXI Master Lite Read Address Channel Protection. This is always driven with a constant output of 000b.
m_axi_lite_arvalid	M_AXI_LITE	O	0	AXI Master Lite Read Address Channel Read Address Valid. Indicates if m_axi_lite_araddr is valid. • 1 = Read Address is valid. • 0 = Read Address is not valid.
m_axi_lite_arready	M_AXI_LITE	I		AXI Master Lite Read Address Channel Read Address Ready. Indicates target is ready to accept the read address. • 1 = Target read to accept address. • 0 = Target not ready to accept address.
AXI4-Lite Master Read Data Channel				
m_axi_lite_rdata (C_M_AXI_LITE_DATA_WIDTH-1: 0)	M_AXI_LITE	I		AXI Master Lite Read Data Channel Read Data.
m_axi_lite_rresp(1:0)	M_AXI_LITE	I		AXI Master Lite Read Data Channel Response. Indicates results of the read transfer. • 00b = OKAY - Normal access has been successful. • 01b = EXOKAY - Not supported. • 10b = SLVERR - Slave returned error on transfer. • 11b = DECERR - Decode error, transfer targeted unmapped address.
m_axi_lite_rvalid	M_AXI_LITE	I		AXI Master Lite Read Data Channel Data Valid. Indicates m_sg_aximry_rdata is valid. • 1 = Valid read data. • 0 = Not valid read data.
m_axi_lite_rready	M_AXI_LITE	O	0	AXI Master Lite Read Data Channel Ready. Indicates the read channel is ready to accept read data. • 1 = Is ready. • 0 = Is not ready.
AXI4-Lite Master Write Address Channel				

Table 1: **AXI Master Lite I/O Signal Description *(Cont'd)***

Signal Name	Interface	Signal Type	Init Status	Description
m_axi_lite_awaddr (C_M_AXI_LITE_ADDR_WIDTH-1: 0)	M_AXI_LITE	O	zeros	AXI Master Lite Write Address Channel Address Bus.
m_axi_lite_awprot(2:0)	M_AXI_LITE	O	000b	AXI Master Lite Write Address Channel Protection. This is always driven with a constant output of 000b.
m_axi_lite_awvalid	M_AXI_LITE	O	0	AXI Master Lite Write Address Channel Write Address Valid. Indicates if m_axi_lite_awaddr is valid. • 1 = Write Address is valid. • 0 = Write Address is not valid.
m_axi_lite_awready	M_AXI_LITE	I		AXI Master Lite Write Address Channel Write Address Ready. Indicates target is ready to accept the write address. • 1 = Target ready to accept address. • 0 = Target not ready to accept address.
AXI4-Lite Master Write Data Channel				
m_axi_lite_wdata (C_M_AXI_LITE_DATA_WIDTH-1 : 0)	M_AXI_LITE	O	zeros	AXI Master Lite Write Data Channel Write Data Bus.
m_axi_lite_wstrb (C_M_AXI_LITE_DATA_WIDTH/8 - 1: 0)	M_AXI_LITE	O	1111b	AXI Master Lite Write Data Channel Write Strobe Bus. All strobe bytes asserted for SG write address channel transfer requests.
m_axi_lite_wvalid	M_AXI_LITE	O	0	AXI Master Lite Write Data Channel Data Valid. Indicates the Write Data Channel has a valid data beat on the bus. • 1 = Valid write data. • 0 = Not valid write data.
m_axi_lite_wready	M_AXI_LITE	I		AXI Master Lite Write Data Channel Ready. Indicates the SG Write Data Channel target slave is ready to accept write data. • 1 = Target slave is ready. • 0 = Target slave is not ready.
AXI4-Lite Master Write Response Channel				
m_axi_lite_bresp(1:0)	M_AXI_LITE	I		AXI Master Lite Write Response Channel Response. Indicates results of the write transfer. • 00b = OKAY - Normal access has been successful. • 01b = EXOKAY - Not supported. • 10b = SLVERR - Slave returned error on transfer. • 11b = DECERR - Decode error, transfer targeted unmapped address.
m_axi_lite_bvalid	M_AXI_LITE	I		AXI Master Lite Write Response Channel Response Valid. Indicates response, m_axi_lite_bresp, is valid. • 1 = Response is valid. • 0 = Response is not valid.
m_axi_lite_bready	M_AXI_LITE	O	0	AXI Master Lite Write Response Channel Ready. Indicates source is ready to receive response. • 1 = Ready to receive response. • 0 = Not ready to receive response.
IPIC Command Interface Signals				

Table 1: **AXI Master Lite I/O Signal Description *(Cont'd)***

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstRd_Req	IPIC	I		IP to Bus Master Read Request. Active high read request initiation control signal.
IP2Bus_MstWr_Req	IPIC	I		IP to Bus Master Write Request. Active high write request initiation control signal.
IP2Bus_Mst_Addr(C_M_AXI_LITE_ADDR_WIDTH-1: 0)	IPIC	I		Bus Master Address. Address to be used for the specified read or write command
IP2Bus_Mst_BE(C_M_AXI_LITE_DATA_WIDTH/8 - 1: 0)	IPIC	I		IP to Bus Master Byte Enables. Input command qualifiers (active high byte enables) used to indicate the valid bytes for the specified write transfer. This input is ignored for read commands.
IP2Bus_Mst_Lock	IPIC	I		IP to Bus Master Lock. This input command qualifier is ignored by the AXI Master Lite.
IP2Bus_Mst_Reset	IPIC	I		IP to Bus Master Reset. Active high reset input used to reset the AXI Master Lite logic.
Bus2IP_Mst_CmdAck	IPIC	O	'0'	Bus to IP Master Command Acknowledge. Active high signal indicating that the Read or Write Address Channel has successfully posted the request to the AXI4 bus.
Bus2IP_Mst_Cmplt	IPIC	O	'0'	Bus to IP Master Command Complete. Active high signal indicating the requested transfer has completed and the associated status bits are valid to sample.
Bus2IP_Mst_Error	IPIC	O	'0'	Bus to IP Master Error. Active high signal indicating an error response was received from the AXI4 bus during the requested transfer.
Bus2IP_Mst_Rearbitrate	IPIC	O	'0'	Bus to IP Master Rearbitrate. Not part of AXI4. This signal is always set to '0'.
Bus2IP_Mst_Timeout	IPIC	O	'0'	Bus to IP Master Timeout. Not part of AXI4. This signal is always set to '0'.
IPIC Read Data Interface Signals				
Bus2IP_MstRd_d(C_M_AXI_LITE_DATA_WIDTH - 1: 0)	IPIC	O	zeros	Bus to IP Master Read Data. Read data output.
Bus2IP_MstRd_src_rdy_n	IPIC	O	'1'	Bus to IP Master Read Source Ready. Active low signal indicating that the data value asserted on the Bus2IP_MstRd_d output bus is valid.
IPIC Write Data Interface Signals				
IP2Bus_MstWr_d(C_M_AXI_LITE_DATA_WIDTH - 1: 0)	IPIC	I		IP to Bus Master Write Data. Write data input.
Bus2IP_MstWr_dst_rdy_n	IPIC	O	'1'	Bus to IP Master Write Destination Ready. Active low signal indicating that the data value asserted on the IP2Bus_MstWr_d input bus has been accepted by the AXI4 bus.

Design Parameters

The AXI Master Lite design parameters are described in Table 2.

Table 2: **AXI Master Lite Design Parameter Description**

Feature/Description	Parameter Name	Allowable Values	Default Values	VHDL Type
AXI Master Lite General Parameters				
Specifies the target FPGA family	C_FAMILY	virtex6, spartan6	virtex6	String
AXI Master Lite AXI4-Lite Parameters				
Address width (in bits) of AXI4-Lite Interface. This is currently fixed at 32 bits.	C_M_AXI_LITE_ADDR_WIDTH	32	32	integer
Data width (in bits) of AXI4-Lite Interface. This is currently fixed at 32 bits.	C_M_AXI_LITE_DATA_WIDTH	32	32	integer

Parameter - I/O Signal Dependencies

Table 3: **Parameter - I/O Signal Dependencies**

Parameter Name	Affects Signal	Depends on Parameter	Relationship Description
C_M_AXI_LITE_DATA_WIDTH	m_axi_lite_rdata m_axi_lite_wdata m_axi_lite_wstrb IP2Bus_Mst_BE Bus2IP_MstRd_d IP2Bus_MstWr_d		The setting of the parameter sets the vector width of the port.
C_M_AXI_LITE_ADDR_WIDTH	m_axi_lite_awaddr m_axi_lite_araddr		The setting of the parameter sets the vector width of the port.

Parameter Descriptions

C_FAMILY

- **Type:** String
- **Allowed Values:** Spartan®-6 and Virtex®-6 and later
- **Definition:** Indicates the target FPGA device family for the design
- **Description:** This parameter is set by the EDK tools to a value reflecting the FPGA device family selected for the EDK project.

C_M_AXI_LITE_ADDR_WIDTH

- **Type:** Integer
- **Allowed Values:** 32 (default = 32)
- **Definition:** Address bus width of attached AXI on the AXI AXI Master Lite interface
- **Description:** This integer parameter is used to size the Read Address and Write Address Channels of the AXI Master Lite AXI4 Interface. The EDK tool suite will assign this parameter a fixed value of 32.

C_M_AXI_LITE_DATA_WIDTH

- **Type:** Integer
- **Allowed Values:** 32 (default = 32)
- **Definition:** Read Data bus width of attached AXI4 on the AXI Master Lite interface
- **Description:** This integer parameter is used to size the Read Data and Write Data Channels of the AXI Master Lite interface. The EDK tool suite will assign this parameter a fixed value of 32.

Clocking

AXI Master Lite uses a single clock for logic synchronization. This clock is input on the `m_axi_lite_aclk` input port. All interfaces for the core are required to be synchronized to this clock. The AXI Master Lite has been simulation tested with an `m_axi_lite_aclk` frequency range of 10 MHz to 200 MHz. Actual F_{max} achieved in a hardware implementation may vary. See Performance for more details.

Resets

An active low reset assertion on the AXI Master Lite `m_axi_lite_aresetn` input will reset the entire AXI Master Lite core. This is considered a hardware reset, and there are no graceful completions of AXI4 transfers in progress. A hardware reset initializes all AXI Master Lite internal logic to power on conditions. It is required that the `m_axi_lite_aresetn` input is synchronous to the `m_axi_lite_aclk` master clock input and is asserted for the minimum number of clocks stated in Table 4. Table 4 also indicates the stabilization time for AXI Master Lite outputs reacting to a reset condition.

Table 4: **Reset Assertion/Deassertion Stabilization Times**

Description	Value	Applicable Signal
Minimum assertion time	8 clocks (m_axi_lite_aclk)	axi_resetn input
Reset assertion to output signals in reset state (maximum)	3 clocks (m_axi_lite_aclk)	All output signals
Reset deassertion to normal operation state (maximum)	3 clocks (m_axi_lite_aclk)	All output signals

Performance

The targeted design clock frequencies of AXI Master Lite for the Spartan-6 and Virtex-6 FPGA families are shown below. The maximum achievable clock frequency and resource utilization estimates may be different from those shown due to tool options, FPGA speed and logic utilization, and other factors.

- Spartan-6 FPGA: 120 MHz
- Virtex-6 FPGA: 180 MHz

Throughput

The AXI Master Lite only performs single data beat transfers of 1 to 4 bytes. It is not designed for high throughput applications. A typical read or write operation should take four `m_axi_lite_aclk` clock cycles from `IP2Bus_MstRd_Req` or `IP2Bus_MstWr_Req` assertion to `Bus2IP_Mst_Cmplt` assertion.

Transaction Timing Examples

This section shows timing relationships for AXI4-Lite and the IPIC interface signals during read and write transfers. Only single data beat transfers of 1 to 4 bytes are supported by this Master.

Single Data Beat Read Operation

Two single beat read cycles are shown in Figure 3. The first cycle shows the AXI Slave accepting the read address and qualifiers in one clock cycle and presenting the read data in the next clock cycle. The second read transfer indicates a delayed address acceptance sequence and a delayed read data valid by the AXI Slave device.

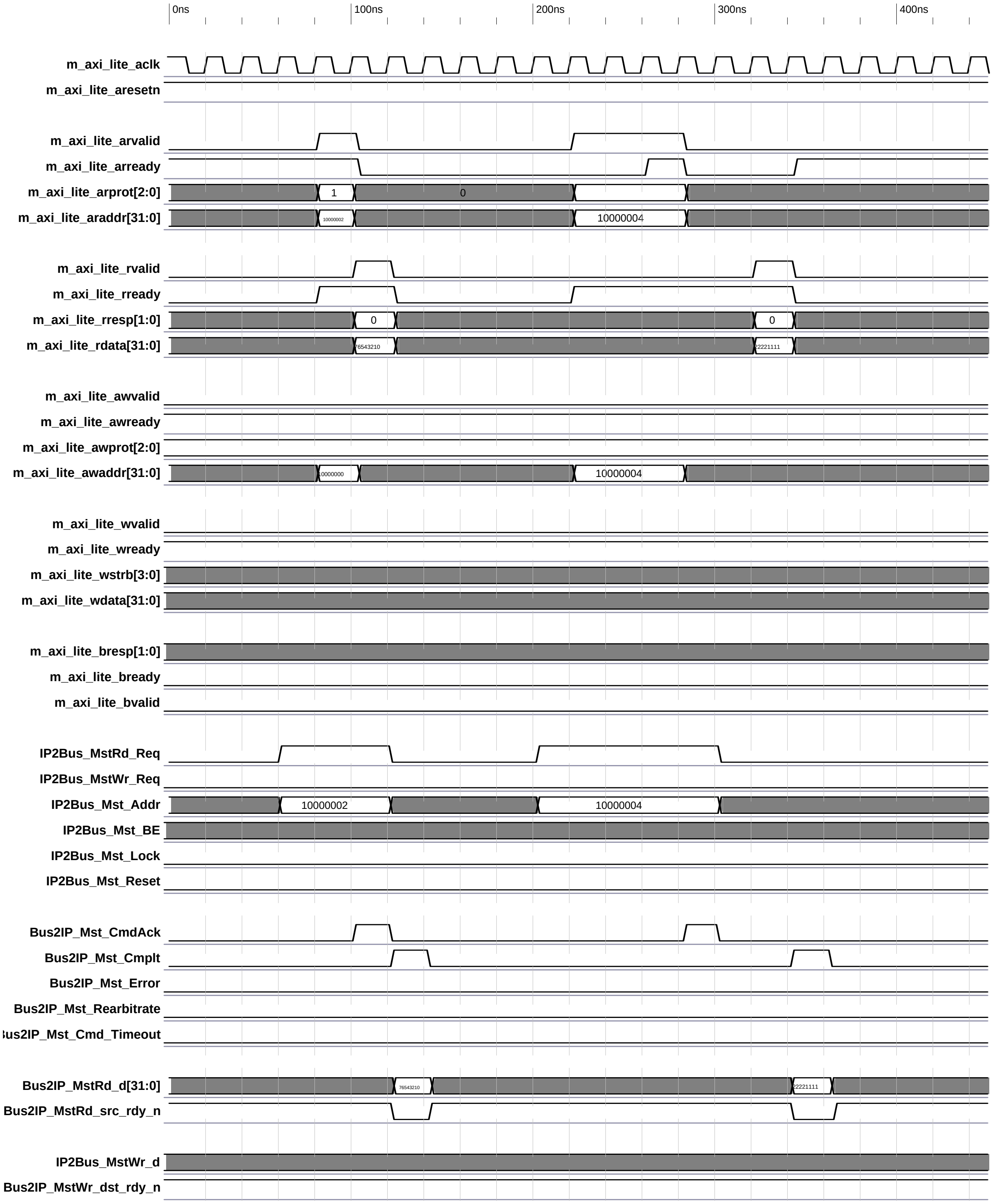

Figure 3: **Example Read Transfer Timing**

Single Data Beat Write Operation

Two single beat write cycles are shown in Figure 4. The first cycle shows the AXI Slave accepting the address and data in one clock cycle followed by a one clock later write response. The second write transfer shows a delayed address acceptance, a delayed write data acceptance, and a delayed write response by the AXI Slave device.

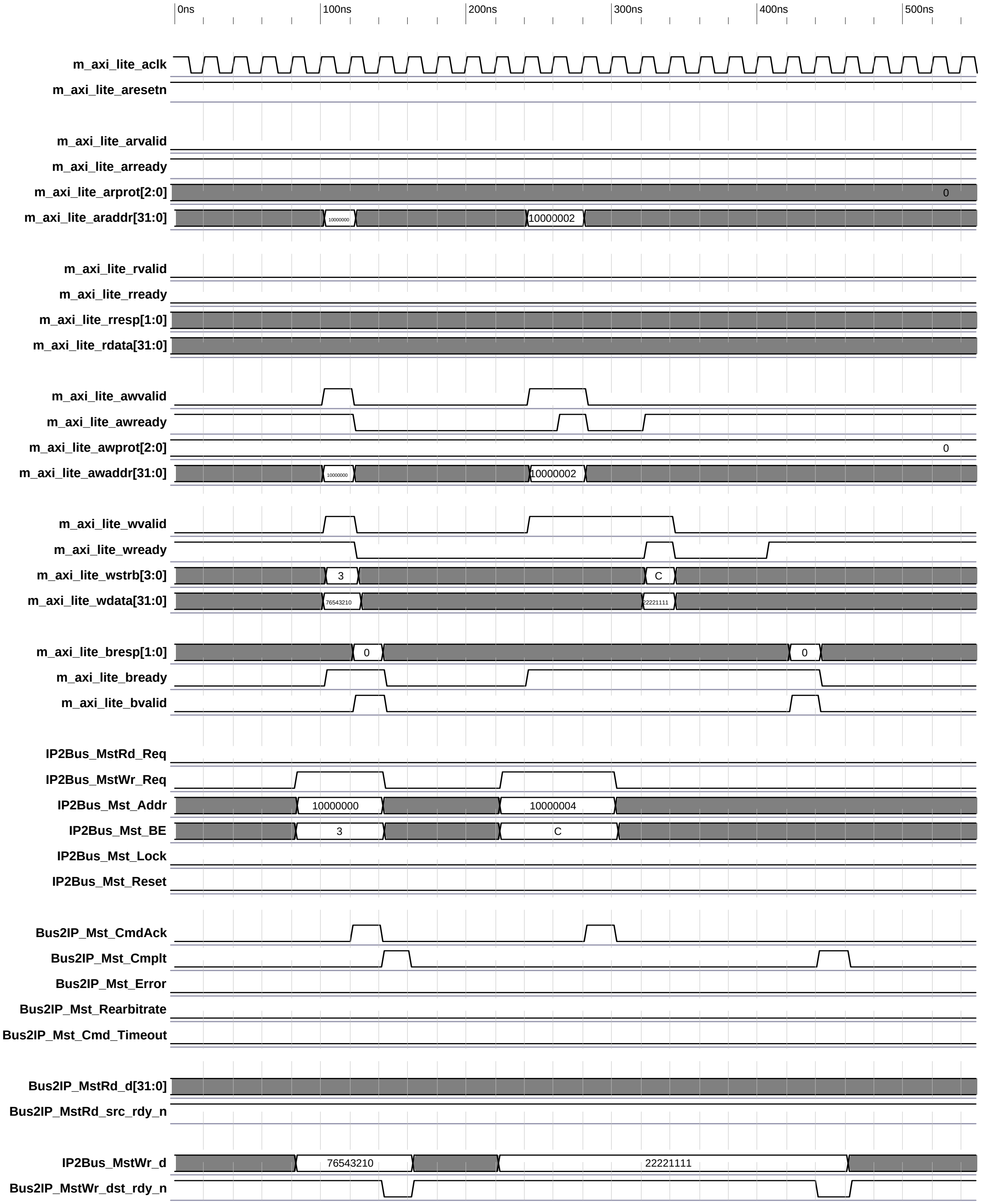

Figure 4: **Example Write Transfer Timing**

Single Data Beat Read Operation with Error

Single data beat Read transfers with a Slave reported error is shown in Figure 5. A Slave data channel response error is reported, and the Master's `md_error` output is asserted and held. The assertion of `md_error` is cleared by the input from the AXI4 interface. The `IP2Bus_Mst_Reset` can also used to clear the `md_error` if desired.

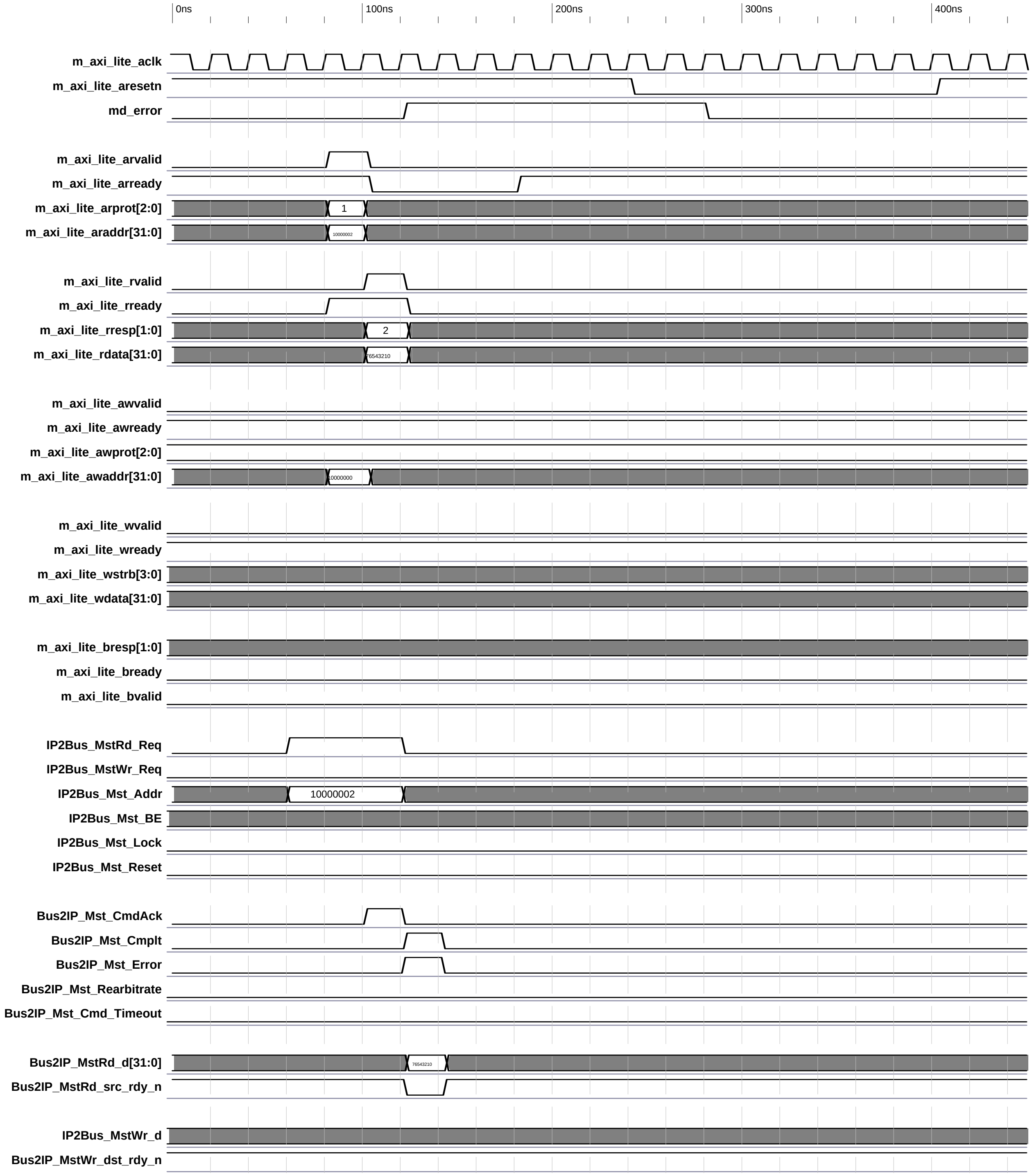

Figure 5: **Example Read Transfer Timing With Error**

Single Data Beat Write Operation with Error

Two single beat write cycles are shown in Figure 6. For both transfers, a Slave data error is reported and the Master's `md_error` output is asserted and held. The assertion of `md_error` is cleared by the input from the AXI4-Lite interface. The `IP2Bus_Mst_Reset` can also used to clear the `md_error` if desired.

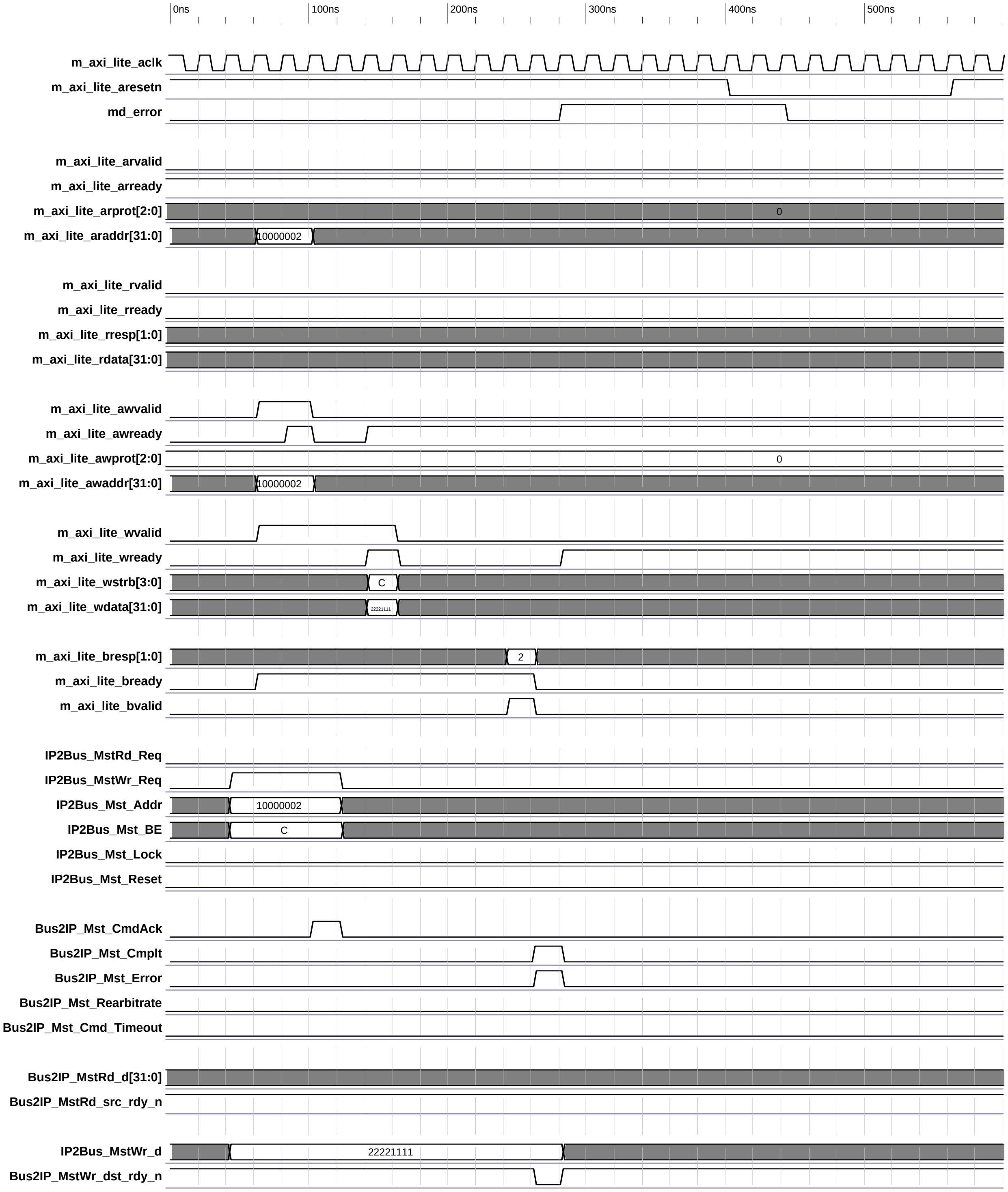

Figure 6: **Example Write Transfer Timing With Error**

Resource Utilization

Resource utilization numbers for the AXI Master Lite core are shown for the Spartan-6 FPGA family in Table 5 and the Virtex-6 FPGA family in Table 6. These values have been generated using the Xilinx EDK and ISE® tools for version 13.1.

Table 5: **Spartan-6 FPGA Resource Estimates**

C_M_AXI_DATA_WIDTH	C_M_AXI_ADDR_WIDTH	Slices	Slice Reg	Slice LUTs	Block RAM
32	32	38	127	59	0

Table 6: **Virtex-6 FPGA Resource Estimates**

C_M_AXI_DATA_WIDTH	C_M_AXI_ADDR_WIDTH	Slices	Slice Reg	Slice LUTs	Block RAM
32	32	36	125	30	0

Support

Xilinx provides technical support for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Ordering Information

This Xilinx LogiCORE IP module is provided at no additional cost with the Xilinx ISE Design Suite Embedded Edition software under the terms of the Xilinx End User License. The core is generated using the Xilinx ISE Embedded Edition software (EDK).

Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and software, please contact your local Xilinx sales representative.

Reference Documents

The following document contains reference information important to understanding the design:

- AXI4 AMBA® AXI Protocol Version: 2.0 Specification

List of Acronyms

Acronym	Spelled Out
AXI	Advanced eXtensible Interface
EDK	Embedded Development Kit
FF	Flip-Flop
FPGA	Field Programmable Gate Array
I/O	Input/Output
IP	Intellectual Property
LUT	Lookup Table
MHz	Mega Hertz
R/W	Read/Write
RAM	Random Access Memory
HW	Hardware
SW	Software
VHDL	VHSIC Hardware Description Language (VHSIC an acronym for Very High-Speed Integrated Circuits)
XPS	Xilinx Platform Studio (part of the EDK software)
XST	Xilinx Synthesis Technology

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
3/01/11	1.0	Initial Xilinx release.

Notice of Disclaimer

Xilinx is providing this product documentation, hereinafter "Information," to you "AS IS" with no warranty of any kind, express or implied. Xilinx makes no representation that the Information, or any particular implementation thereof, is free from any claims of infringement. You are responsible for obtaining any rights you may require for any implementation based on the Information. All specifications are subject to change without notice. XILINX EXPRESSLY DISCLAIMS ANY WARRANTY WHATSOEVER WITH RESPECT TO THE ADEQUACY OF THE INFORMATION OR ANY IMPLEMENTATION BASED THEREON, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OR REPRESENTATIONS THAT THIS IMPLEMENTATION IS FREE FROM CLAIMS OF INFRINGEMENT AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Except as stated herein, none of the Information may be copied, reproduced, distributed, republished, downloaded, displayed, posted, or transmitted in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx.