PLBV46 Master (v1.00a)

Introduction

The PLB Master is a continuation of the Xilinx family of IBM CoreConnect compatible LogiCORE products. It provides a bi-directional Bus Mastering interface between a User IP logic (also called the Client IP) and the PLB V4.6 bus standard.

Features

- Compatible with IBM CoreConnect 128-bit PLB V4.6. with Xilinx Simplifications
  - Leverages PLB Split Bus architecture which allows simultaneous Read and Write Data Phases.
  - Supports initiation of Single data beat and Fixed Length Burst operations via the User IP Client interface.
  - Support for transfers with narrower Slaves through the incorporation of Conversion Cycle and Burst Length Expansion logic.
- Support of unaligned burst transfers. Read and Write Controllers perform automatic burst to single conversions for starting and ending bursts with non-native data width aligned boundaries.
- Incorporates the Core Generator™ system synchronous or asynchronous Read and Write FIFOs in the interface logic to the IP Client. These have parameterized depth.
  - Xilinx LocalLink interface used in as the data transfer protocol in the IP Client interface

LogiCORE™ Facts		
Core Specifics		
Supported Device Family	See EDK Supported Device Families.	
Version of core	plbv46_master	v1.00a
Resources Used		
	Min	Max
Slices	512	2166
LUTs	869	3438
FFs	502	1940
Block RAMs	0	130(36K) or 260(18K)
Special Features	None	
Provided with Core		
Documentation	Product Specification	
Design File Formats	VHDL	
Constraints File	None	
Verification	EDK Simulation Support	
Instantiation Template	None	
Reference Designs & application notes	None	
Additional Items	None	
Design Tool Requirements		
Xilinx Implementation Tools	See Tools for requirements.	
Verification		
Simulation		
Synthesis		
Support		
Provided by Xilinx, Inc.		

Functional Description

The PLBV46 Master with LocalLink is a VHDL design targeting high data rate applications needing a PLB Master interface. It incorporates features that are designed to meet the needs of the DMA/Scatter Gather function. The design incorporates three major interface groups; the Request Controller, the Read Controller, and the Write Controller. The block diagram of the PLBV46 Master design is shown in Figure 1. The following is a list of high level features of the design:

- Provide a design with high data transfer rate potential.
  - Parameterized Native Data Width of 32, 64, and 128 bits.
  - Leverage the PLB Split Bus Architecture which provides the capability to perform simultaneous Read and Write PLB data phases.
  - Support Single Data Beat and Fixed Length Burst protocol.
  - Incorporates the Xilinx LocalLink interface protocol for the data transfer mechanism between the Master and the User IP Client. This protocol was developed for high bandwidth interfaces between Xilinx networking IP.
  - The Master's Architecture is such that it can be used in conjunction with a separate PLB Slave IPIF to provide a Master/Slave solution or separately as a stand alone PLB Master only interface.
    - No interdependencies between the PLB IPIF Slave Attachment and the PLB Master functions.
- Incorporates three selectable FIFO implementations for the Read and Write FIFOs in the LocalLink interfaces.
  - Synchronous SRL based FIFO (uses SRL16 primitives instead of BRAM).
  - Synchronous Core Genator system FIFO using BRAM.
  - Asynchronous Core Generator system FIFO using BRAM.
- Enhanced unaligned transfer support
  - Both the Read and Write Controllers of the Master have added functionality for automatically processing burst transfers that may start and/or end on non-native width aligned address boundaries.
- Support for transfers with narrower Slaves
  - Conversion Cycle logic included for Single Data beat completions with narrow Slaves.
  - Burst Length Expansion for supporting fixed length burst completions with narrower Slaves.

.

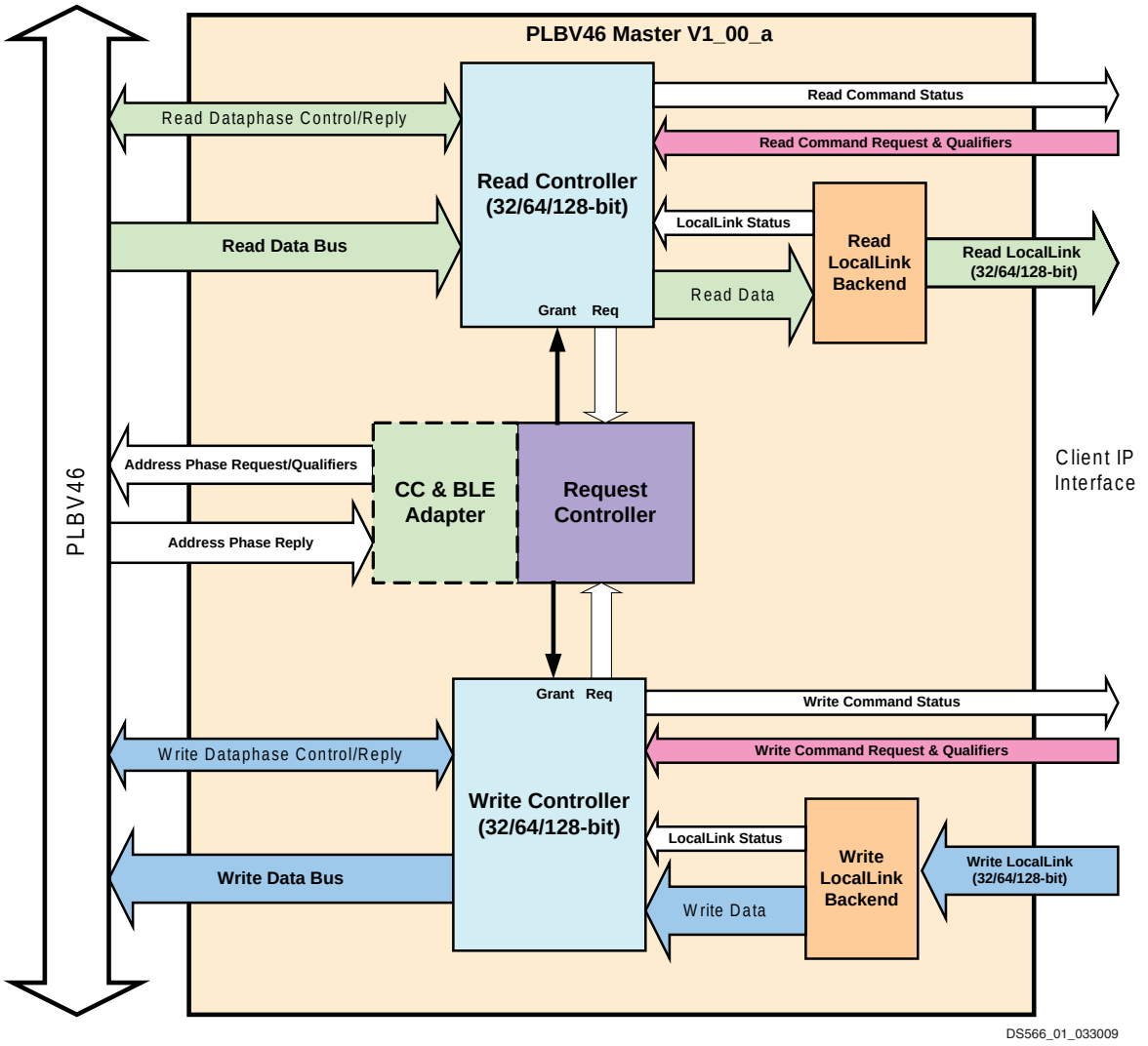

Figure 1: **PLBV46 Master Block Diagram**

Request Controller

The Request Controller is designed to manage the Address Phase of PLB requests. The Address Phase is defined as the time from the assertion of M_request by the Master to either the receipt of PLB_addrAck from the PLB. A transaction request is initiated by the IP Client via the Write Controller or the Read Controller. As part of its responsibility, the Request Controller must arbitrate between the Read and Write Controllers when requests are being posted simultaneously. The Read and Write Controllers are provided with a Grant signal when it is being given access to the PLB by the Request Controller.

The Request Controller Arbiter monitors requests from the Read and the Write Controllers. When simultaneous requests occur and arbitration is required, the Arbiter is designed to give priority to Write requests over Reads. However, the Arbiter is designed such that access starvation cannot occur. At the completion of a request arbitration cycle, a pending request from a different Controller is given priority over the Controller that has already been given a grant and is completing it's resulting data phase.

A special arbitration modification is used in the case of PLB rearbitration of a Master's request. If a Read request is rearbitrated, then the Arbiter checks for a pending Write from the Write Controller. If a Write is pending, the Arbiter will grant PLB access to the pending Write. However, if a Write Request is rearbitrated, the Arbiter will ignore any pending Read request and continue to grant the Write access to the PLB. This scheme is required for PCIE Bridge applications.

Conversion Cycle and Burst Length Expansion Adapter (CC & BLE Adapter)

The CC/BLE Adapter adds the ability for the Master to support Conversion Cycles and Burst Length Expansion. These functions are required if the Master accesses a PLB Slave that has a Native Data Width that is smaller than the Native Data Width of the Master. The CC/BLE Adapter is automatically included or omitted from the Master implementation depending on the assigned values for the C_MPLB_NATIVE_DWIDTH and C_MPLB_SMALLEST_SLAVE. If C_MPLB_SMALLEST_SLAVE is smaller than C_MPLB_NATIVE_DWIDTH, then the CC/BLE Adapter will be included automatically. However, if a User knows that the Master will never be accessing any Slave that is narrower than the Master's Native Data Width, then the adapter can be omitted manually by assigning the parameter C_INHIBIT_CC_BLE_INCLUSION a value of 1.

Read Controller

The Read Controller provides the IP Client with the capability to perform PLB read transactions in the form of Singles and Fixed Length Bursts. The IP Client initiates a read transaction via the Read Controller's Command interface. The Read Command signal set is graphically shown in Figure 2. The interface is a a mix of request, qualifier, and status reply signals and buses. These signals are detailed in Table 1, "PLBV46 Master I/O Signal Description," on page 9.

The Read Controller orchestrates the initiation of requests to the Master Request Controller and the associated read data transfer from the PLB to the Read LocalLink Backend. A Client IP may request a large burst read transfer of hundreds or thousands of bytes. The Read Controller must examine the start and end conditions of the burst request to determine if a single data beat is required to start and end the request and breaks up the intermediate burst operations into fixed length burst requests of 2 to 16 data beats. At the same time the Read Controller must monitor the Read LocalLink Backend status and adjust it's PLB Read request generation as that status changes due to rate at which the Client IP is consuming the Read Data. All operations and interfaces on the Read Controller are synchronous with the PLB clock.

Read LocalLink Backend

Read data transfer from the Master to the IP Client is handled by the Read LocalLink Backend. It utilizes the Xilinx LocalLink transfer protocol. The LocalLink protocol has been developed to support high data rate transmissions required by networking type IP. It dovetails well with the use of FIFOs as an intermediate data storage element. This protocol is documented in the Xilinx document SP006 titled LocalLink Interface Specification. These signals are also detailed in Table 1, "PLBV46 Master I/O Signal Description," on page 9.

Via parameterization, the module can be customized to incorporate an asynchronous Core Generator FIFO, a synchronous Core Generator system FIFO, or an SRL based FIFO. This FIFO is referred to as the Read FIFO. The Read FIFO may be used as a time domain transform mechanism and as an intermediate storage device. Data is written into the RdFIFO by the Read Controller synchronous to the PLB clock.

The Read LocalLink Backend provides status information to the Read Controller so that the Read Controller can adapt it's PLB Read request generation to the state of the Read LocalLink Backend. A

good example of this is if the Read FIFO is approaching a full condition, the Read Controller may not be able to post a burst read request on the PLB because the incoming read data would overrun the Read FIFO.

If asynchronous Read LocalLink operation has been specified (see parameter C_RD_LLINK_IS_ASYNC), the Read LocalLink interface to the IP Client is synchronized to an input clock (IP2Bus_MstRd_clk) that is provided by the IP Client. If synchronous Read LocalLink interface operation has been specified, the Read LocalLink interface utilizes the PLB Bus clock for the timing standard and the IP2Bus_MstRd_clk is ignored by the Master. On the write side of the Read FIFO, operations are coordinated by the Read Controller which is always synchronous to the PLB clock.

Write Controller

The Write Controller provides the IP Client with the capability to perform PLB write transactions in the form of Singles and Fixed Length Bursts. The IP Client initiates a write transaction via the Write Controller's Command interface. The Write Command interface is separate from the Read Command interface bus has the same signal set. This is graphically shown in Figure 3. The interface is a a mix of request, qualifier, and status reply signals and buses. These signals are detailed in Table 1, "PLBV46 Master I/O Signal Description," on page 9.

Like the Read Controller, the Write Controller orchestrates the initiation of write requests to the Master Request Controller and the associated write data transfer to the PLB from the Write LocalLink Backend. A Client IP may request a large burst write transfer of hundreds or thousands of bytes. The Write Controller must examine the start and end conditions of the burst request to determine if a single data beat write is required to start and end the request and breaks up the intermediate burst operations into fixed length burst write requests of 2 to 16 data beats. At the same time, the Write Controller must monitor the Write LocalLink Backend status and adjust it's PLB Write request generation as the status changes due to rate at which the Client IP is providing the Write Data. All operations and interfaces on the Read Controller are synchronous with the PLB clock.

Write LocalLink Backend

Write data transfer from the IP Client is handled by the Write LocalLink Backend. The module utilizes the Xilinx LocalLink transfer protocol. The module incorporates either an asynchronous Core Generator system FIFO or a synchronous Core Generator system FIFO for the WrFIFO. The Write FIFO is used as a time domain transform mechanism and as an intermediate storage device. Data is written into the WrFIFO by the Write LocalLink Controller.

If asynchronous Write LocalLink operation has been specified (see parameter C_WR_LLINK_IS_ASYNC), the Write LocalLink to WrFIFO operations are synchronized to an input clock (IP2Bus_MstWr_clk) that is provided by the IP Client. If synchronous Write LocalLink interface operation has been specified, the Write LocalLink interface utilizes the PLB Bus clock for the timing standard and the IP2Bus_MstWr_clk is ignored. On the read side of the Write FIFO, operations are coordinated by the Write Controller which is always synchronous to the PLB clock.

Read and Write Command Interfaces

The Read and Write Controllers within the PLBV46 Master serve the IP Client via a command interface. The basic command interface is symmetrical between Read and Write Controllers. The command interface is how the Client IP makes request for data transfer to the Master and how the Master Service provides high level status of the transfer back to the IP Client, The basic command interface consists of the following:

- Inputs to the Master Service (from IP Client)
  - Command Request
  - Command Starting PLB address
  - Command Transfer Length limit (in bytes)
  - Command BE designator (for single data beat requests)
  - Command Size field (designates number of bytes transferred for each data beat of a burst transfer)
  - Command Type field (specifies Single Data Beat or Fixed Length Burst)
  - Command Bus Lock (function not currently supported)
  - Command Address Increment
  - Reset Request
- Status outputs from the Master to the IP Client
  - Command Acknowledge
  - Command Complete
  - Command Error
  - Command PLB Rearbitrate
  - Command Timeout
  - Command Length Limit Count is Zero

The Command interface protocol requires that the IP Client drive the Request and the associated qualifiers until the Command Acknowledge is received from the Master. Upon the receipt of the Command Acknowledge the IP Client may stop asserting the Request and the qualifiers for the command. If a problem occurs during the PLB Address Phase of a request, it is possible that a Command Acknowledge will not occur. Instead the Master may reply with the Command Abort or command Error status asserted in conjunction with assertion of the Command Complete status. This is an indication to the IP Client that a non-recoverable error occurred and the transfer did not complete. If a Command Acknowledge is asserted by the Master Service, then the Master has successfully negotiated a PLB Address Phase and the corresponding data phase is in progress. When the Data Phase completes, the Master will assert the Command Complete signal. If a Data Phase error is received from the PLB Slave by the Master, the Command Error status will be also be asserted when the Command Complete is asserted. The duration of the Command Complete assertion is one PLB clock period.

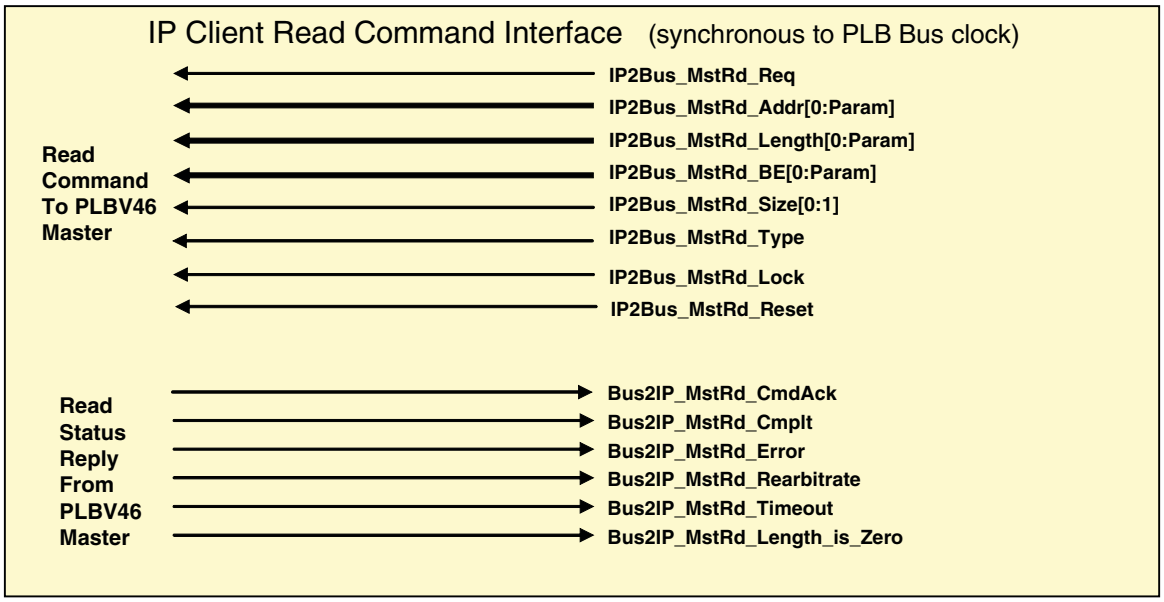

Figure 2: **IP Client Read Command Interface Signal Set**

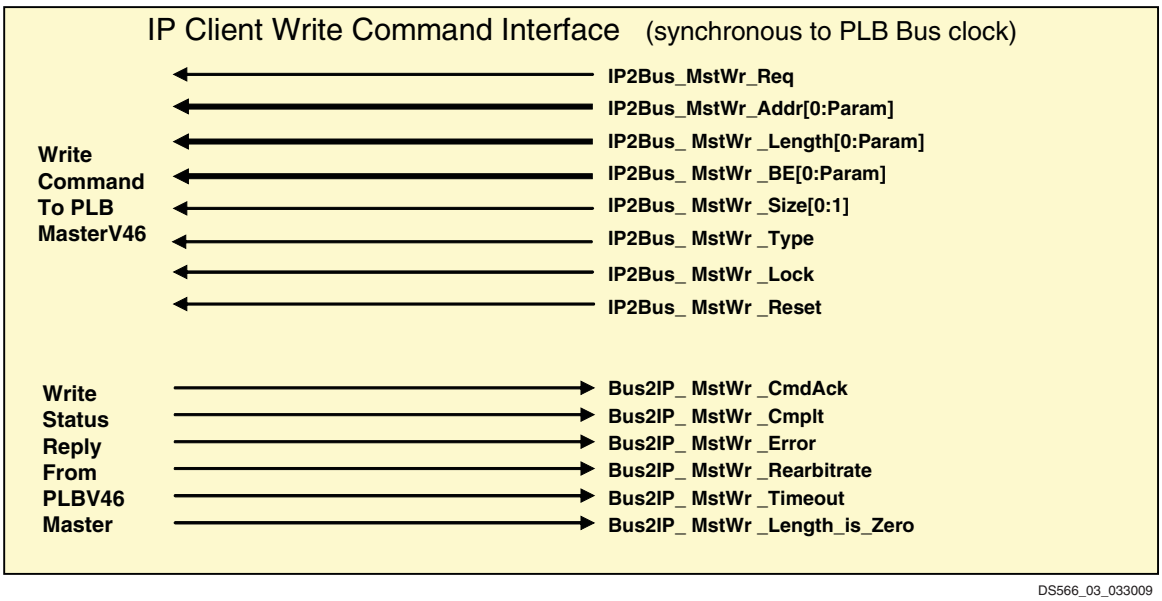

Figure 3: **IP Client Write Command Interface Signal Set**

Xilinx LocalLink Interface Summary

The Client IP receives data from and transmits data to the PLB Master via the Xilinx LocalLink Interface protocol. LocalLink is a point-to-point, synchronous interface intended for high data rate applications. Because data flow is unidirectional, the PLB Master employs two LocalLink interfaces, one for IP Client data read operations and one for IP Client data write operations. The Read and Write interfaces are independent from each other and thus enable the PLB Master to support simultaneous Read and Write data transfers with the IP Client. This enables the PLB Master to leverage the split bus architecture of the PLB and eliminates any need for PLB IPIF support logic to decode addresses and mechanize the data transfers.

LocalLink is based upon the concept of a Source device transmitting data to a Destination device. Data flow is unidirectional; always from the Source to the Destination. Both Source and Destination can throttle transfers as well as choose to discontinue the transfer. In order for a transfer data beat to complete., both the Source and the Destination must signal that they are ready at the rising edge of the transfer synchronization clock (clk). The Source indicates a ready condition by asserting the src_rdy_n signal. The Destination indicates ready by asserting the dst_rdy_n signal.

Data (d[n:0]) is transferred in a delimited group otherwise known as a packet. The start of a packet is delimited with the assertion of the Start-of-Frame signal (sof_n) by the Source. The assertion of End-of-Frame by the Source (eof_n) delimits the last data beat of a packet. A single data beat transfer is delimited with simultaneous assertion of sof_n and eof_n.

Transfer acknowledge/throttling is accomplished with the assertion of src_rdy_n and dst_rdy_n. De-assertion of either signal will throttle the transfer. If the Destination device can no longer transfer data or no longer needs data, it may assert the dst_dsc_n to discontinue the transfer. Conversely, the Source may terminate transmission prematurely with the assertion of the src_dsc_n signal.

The rem[0:n] signal (short for remainder) is set by the Source during each data beat in which a delimiter flag is set (sof_n, sop_n, eop_n, eof_n). The value asserted specifies the valid bytes in that data beat and are somewhat application specific depending on the needs of the source and destination devices. The rem can be either an encoded value or a masked value and is set via parameterization. The active assertion level is parameterizable active high or active low depending on the User needs. Byte lane ordering follows PLB byte lane ordering (0 is MSB).

A basic LocalLink data transfers are shown in Figure 4. The data packet consists of 16 data beats of 64 bits wide. The diagram shows both the Source and Destination throttling the transfer.

Note: This PLB Master follows the IBM CoreConnect convention of left-to-right bit ordering and Big Endian byte ordering.

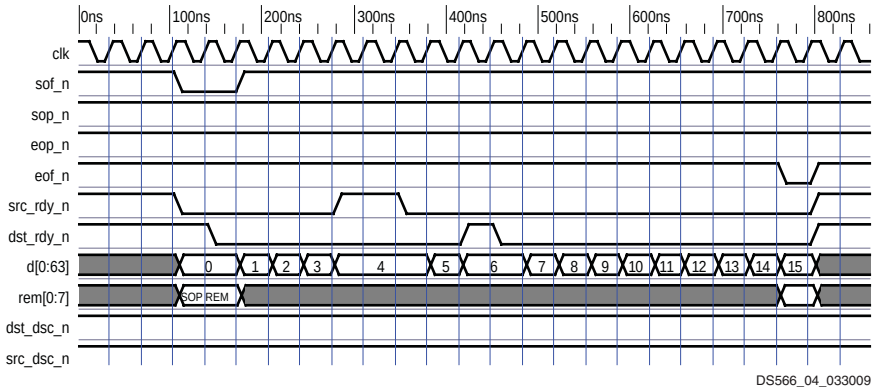

Figure 4: **Basic LocalLink Data Transfer**

I/O Signals

The PLBV46 Master signals are listed and described in Table 1.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
System Signals				
MPLB_clk	PLB System	Input		Synchronization clock for the PLB Interface
MPLB_Rst	PLB System	Input		Master Reset for the PLB Interface (Active high)
MD_Error	Sideband Signal	Output	'0'	Master Detected Error output. Asserted active high when Master encounters an error condition. Cleared by MPLB_Rst assertion.
PLB Master Request Signals				
M_request	PLB Bus	Output	'0'	See table note 1.
M_priority	PLB Bus	Output	all '0'	See table note 1.
M_buslock	PLB Bus	Output	'0'	See table note 1.
M_RNW	PLB Bus	Output	'0'	See table note 1.
M_BE(0 : **C_MPLB_DWIDTH**/8-1)	PLB Bus	Output	all '0'	See table note 1.
M_MSize(0 : 1)	PLB Bus	Output	all '0'	See table note 1.
M_size(0 : 3)	PLB Bus	Output	all '0'	See table note 1.
M_type(0 : 2)	PLB Bus	Output	all '0'	See table note 1.
M_TAttribute(0:15)	PLB Bus	Output	all '0'	See table note 1.
M_lockErr **(2)**	PLB Bus	Output	'0'	See table note 1.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
M_abort	PLB Bus	Output	'0'	See table note 1.
M_UABus(0 : 31)	PLB Bus	Output	all '0'	See table note 1.
M_ABus(0 : 31)	PLB Bus	Output	all '0'	See table note 1.
M_wrDBus(0 : **C_MPLB_DWIDTH**-1)	PLB Bus	Output	all '0'	See table note 1.
M_wrBurst	PLB Bus	Output	'0'	See table note 1.
M_rdBurst	PLB Bus	Output	'0'	See table note 1.
PLB Request Reply Signals				
PLB_MAddrAck	PLB Bus	Input		See table note 1
PLB_MSSize(0 : 1)	PLB Bus	Input		See table note 1
PLB_MRearbitrate	PLB Bus	Input		See table note 1
PLB_MTimeout	PLB Bus	Input		See table note 1
PLB_MBusy	PLB Bus	Input		See table note 1
PLB_MRdErr	PLB Bus	Input		See table note 1
PLB_MWrErr	PLB Bus	Input		See table note 1
PLB_MIRQ	PLB Bus	Input		See table note 1
PLB_MRdDBus(0 : **C_MPLB_DWIDTH**-1)	PLB Bus	Input		See table note 1
PLB_MRdWdAddr(0 : 3)	PLB Bus	Input		See table note 1
PLB_MRdDAck	PLB Bus	Input		See table note 1
PLB_MRdBTerm	PLB Bus	Input		See table note 1
PLB_MWrDAck	PLB Bus	Input		See table note 1
PLB_MWrBTerm	PLB Bus	Input		See table note 1
IP Client Read Command I/O Signals				
IP2Bus_MstRd_Req	IP Read Command Intfc.	Input		Active high signal initiating a Write transaction. This signal and associated qualifiers must be asserted and held until the Bus2IP_MstRd_CmdAck reply signal is sampled as asserted. At that time, the IP2Bus_MstRd_Req must be deasserted.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstRd_Addr(0 : **C_MPLB_AWIDTH**-1)	IP Read Command Intfc.	Input		Address bus from the User IP used to convey the desired starting address to be output on the PLB Bus during the requested IP Write operation. The qualifier must be stable and valid during the assertion of the IP2Bus_MstRd_Req signal.
IP2Bus_MstRd_Length(0 : **C_LENGTH_WIDTH**-1)	IP Read Command Intfc.	Input		This command qualifier bus specifies the not-to-exceed number of bytes to be transferred for the read command. The qualifier must be stable and valid during the assertion of the IP2Bus_MstRd_Req signal.
IP2Bus_MstRd_BE(0 : (**C_MPLB_NATIVE_DWIDTH**/8) -1)	IP Read Command Intfc.	Input		Byte Enable bus from the User IP used to convey the desired Byte Enables output on the PLB Bus during a Single Data Beat read operation. This qualifier is only valid during Single Data Beat requests. The qualifier must be stable and valid during the assertion of the IP2Bus_MstRd_Req signal.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstRd_Size(0 : 1)	IP Read Command Intfc.	Input		Read Command qualifier bus used to indicate the transfer width for a Fixed Length Burst transfer on the PLB. This qualifier is only valid during Burst requests. 00 = Words (32 bits) 01 = Double Words (64 bits) 10 = Quad words 11 = Reserved The qualifier must be stable and valid during the assertion of the IP2Bus_MstRd_Req signal.
IP2Bus_MstRd_Type	IP Read Command Intfc.	Input		Read Command qualifier bus used to indicate the desired transfer type to be requested by the Master. '0' = Single Data Beat '1' = Fixed Length Burst The qualifier must be stable and valid during the assertion of the IP2Bus_MstRd_Req signal.
IP2Bus_MstRd_Lock	IP Read Command Intfc.	Input		Reserved: Drive input to logic '0'. Read Bus Lock not currently supported.
IP2Bus_MstRd_Reset	IP Read Command Intfc.	Input	'0'	Active high signal requesting a reinitialization of the Write logic and Read FIFO.
Bus2IP_MstRd_CmdAck	IP Read Command Intfc.	Output	'0'	Active high signal asserted for one PLB Clock period to indicate that the PLB Bus has accepted the Read Command request and the data phase is being initiated. Request and qualifiers must be deasserted when this signal is sampled as asserted.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
Bus2IP_MstRd_Cmplt	IP Read Command Intfc.	Output	'0'	Active high signal asserted for one PLB Clock period to indicate that the PLB Bus has completed the data phase of the requested read transfer or has completed with an exception. This signal when asserted is also an indication to the User Logic that the associated status reply is valid and may be sampled
Bus2IP_MstRd_Error	IP Read Command Status Reply Intfc.	Output	'0'	Active high signal indicating that an error has occurred during the requested Read transaction.
Bus2IP_MstRd_Rearbitrate	IP Read Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the Master is attempting to initiate the requested read operation but the PLB Slave device is responding with a rearbitrate status.
Bus2IP_MstRd_Cmd_Aborted	IP Read Command Status Reply Intfc.	Output	'0'	This active high signal is only asserted if an invalid Read request is issued by the IP. An example is the IP2Bus_MstRd_Length value for a request is set to zeros.
Bus2IP_MstRd_Cmd_TImeout	IP Read Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the Master has received an address phase Timeout indication for the requested read operation.
Bus2IP_MstRd_Length_is_Zero	IP Read Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the data transferred during the Read Operation was limited by the IP2Bus_MstRd_Length qualifier.
Read LocalLink I/O Signals				

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstRd_clk	IP Read LocalLink Intfc.	Input		The clock used to synchronize the User IP and the Write LocalLink interface when an asynchronous Read LocalLink interface is specified via parameterization. This input is ignored with a synchronous Read LocalLink interface.
Bus2IP_MstRd_sof_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating the Start Of Frame for the requested read data packet.
Bus2IP_MstRd_eof_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating the End Of Frame for the requested read data packet.
Bus2IP_MstRd_sop_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating the Start Of Payload for the requested read data packet. This signal is not used.
Bus2IP_MstRd_eop_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating the End Of Payload for the requested read data packet. An EOP delimiter is required when a LocalLink footer is needed for the SG operations.This signal is not used.
Bus2IP_MstRd_rem(0 : **C_REM_WIDTH**-1)	IP Read LocalLink Intfc.	Output	See Table Note 2	One, Three, or eight bit bus indicating the valid Bytes in the associated LocalLink data bus (Bus2IP_MstRd_d). It is sampled when any of the LocalLink delimiter signals are asserted.
Bus2IP_MstRd_d(0 : **C_MPLB_NATIVE_DWIDTH**-1)	IP Read LocalLink Intfc.	Output	All '0'	Read LocalLink Data bus. A data packet is started with the assertion of the Bus2IP_MstRd_sof_n and ended with the assertion of the Bus2IP_MstRd_eof_n.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
Bus2IP_MstRd_src_rdy_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating the PLBV46 Master Read LocalLink interface has data available in the Read FIFO to transfer to the IP client. A data transfer occurs when both the IP2Bus_MstRd_dst_rdy_n and the Bus2IP_MstRd_src_rdy_n are asserted on the rising edge of the Read LocalLink synchronization clock.
Bus2IP_MstRd_src_dsc_n	IP Read LocalLink Intfc.	Output	'1'	Active low signal indicating that the Write Controller is discontinuing the read transfer.
IP2Bus_MstRd_dst_rdy_n	IP Read LocalLink Intfc.	Input		Active low signal indicating to the Write Controller that the IP client is ready to accept data. A data transfer occurs when both the IP2Bus_MstRd_dst_rdy_n and the Bus2IP_MstRd_src_rdy_n are asserted on the rising edge of the Read LocalLink synchronization clock.
IP2Bus_MstRd_dst_dsc_n	IP Read LocalLink Intfc.	Input		Active low signal indicating to the Write Controller that the client IP is discontinuing the LocalLink Read transfer.
Bus2IP_MstRd_RdCnt(0 : **C_RDFIFO_RDCNT_WIDTH**-1)	IP Read LocalLink Intfc.	Output	All '0'	This bus reflects the number of words that are loaded in the Write FIFO. It is synchronous to the Read LocalLink synchronization clock. This bus may be used by the User IP to monitor the Read FIFO occupancy.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstRd_Vacancy(0:**C_RDFIFO_VACANCY_WIDTH**-1)	IP Read LocalLink Intfc.	Input		This field is only used by the Read Controller when the Read FIFO has been omitted from the Read LocalLink Backend and the parameter **C_RDFIFO_VACANCY_WIDTH** is 2 or greater. This bus is driven by the User IP and indicates the available space in a User IP FIFO that the Read Controller will be writing data to via the Write LocalLink. The Read Controller will stop posting Read requests to the PLB if the value of this bus drops below a parameterized threshold. If the User IP does not implement a FIFO, then the bus should be driven with all bits asserted indicating maximum space available at all times or assign the **C_RDFIFO_VACANCY_WIDTH parameter** to a value of 1.
IP Client Write Command I/O Signals				
IP2Bus_MstWr_Req	IP Write Command Intfc.	Input		Active high signal initiating a Write transaction. This signal and associated qualifiers must be asserted and held until the Bus2IP_MstWr_CmdAck reply signal is sampled as asserted. At that time, the IP2Bus_MstWr_Req must be deasserted.
IP2Bus_MstWr_Addr(0 : **C_PLB_AWIDTH**-1)	IP Write Command Intfc.	Input		Address bus from the User IP used to convey the desired starting address to be output on the PLB Bus during the requested IP Write operation. The qualifier must be stable and valid during the assertion of the IP2Bus_MstWr_Req signal.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstWr_Length(0 : **C_LENGTH_WIDTH**-1)	IP Write Command Intfc.	Input		This command qualifier bus specifies the not-to-exceed number of bytes to be transferred for a write command. The qualifier must be stable and valid during the assertion of the IP2Bus_MstWr_Req signal.
IP2Bus_MstWr_BE(0 : (**C_MPLB_NATIVE_DWIDTH**/8) -1)	IP Write Command Intfc.	Input		Byte Enable bus from the User IP used to convey the desired Byte Enables to be output on the PLB Bus during IP Single Data Beat Write operations. The qualifier must be stable and valid during the assertion of the IP2Bus_MstWr_Req signal.
IP2Bus_MstWr_Size(0 : 1)	IP Write Command Intfc.	Input		Write Command qualifier bus used to indicate the transfer width for a Fixed Length Burst transfer on the PLB. This qualifier is only valid during Burst requests. 00 = Words (32 bits) 01 = Double Words (64 bits) 10 = Quad words 11 = Reserved The qualifier must be stable and valid during the assertion of the IP2Bus_MstWr_Req signal.
IP2Bus_MstWr_Type	IP Write Command Intfc.	Input		Write Command qualifier bus used to indicate the desired transfer type to be requested by the Master. '0' = Single Data Beat '1' = Fixed Length Burst The qualifier must be stable and valid during the assertion of the IP2Bus_MstWr_Req signal.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstWr_Lock	IP Write Command Intfc.	Input		Reserved: Drive input to logic '0'. Write Bus Lock is not currently supported.
IP2Bus_MstWr_Reset	IP Write Command Intfc.	Input		Active high signal requesting a reinitialization of the Write logic and Write FIFO.
Bus2IP_MstWr_CmdAck	IP Write Command Intfc.	Output	'0'	Active high signal asserted for one PLB Clock period to indicate that the PLB Bus has accepted the Write Command request and the data phase is being initiated.
Bus2IP_MstWr_Cmplt	IP Write Command Intfc.	Output	'0'	Active high signal asserted for one PLB Clock period to indicate that the PLB Bus has completed the data phase of the requested Write transfer or has completed with an exception. This signal when asserted is also an indication to the User Logic that the associated status reply is valid and may be sampled.
Bus2IP_MstWr_Error	IP Write Command Status Reply Intfc.	Output	'0'	Active high signal indicating that some type of error occurred during the requested Write transaction.
Bus2IP_MstWr_Rearbitrate	IP Write Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the Master is attempting to initiate the requested write operation but the PLB Slave device is responding with a rearbitrate status.
Bus2IP_MstWr_Cmd_Aborted	IP Write Command Status Reply Intfc.	Output	'0'	This active high signal is only asserted if an invalid Write request is issued by the IP. An example is the IP2Bus_MstWr_Length value for a request is set to zeros.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
Bus2IP_MstWr_Cmd_TImeout	IP Write Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the Master has received a PLB address phase Timeout indication during the course of executing the requested write operation.
Bus2IP_MstWr_Length_is_Zero	IP Write Command Status Reply Intfc.	Output	'0'	This active high signal indicates that the data transferred during the Write Operation was limited by the IP2Bus_MstWr_Length qualifier.
IP Client Write LocalLink I/O Signals				
IP2Bus_MstWr_clk	IP Write LocalLink Intfc.	Input		The clock used to synchronize the User IP and the Write interface when an **asynchronous** Write FIFO is being utilized. This signal is ignored when a synchronous or SRL Write FIFO has been specified.
IP2Bus_MstWr_sof_n	IP Write LocalLink Intfc.	Input		Active low signal indicating the Start Of Frame for the requested Write data packet.
IP2Bus_MstWr_eof_n	IP Write LocalLink Intfc.	Input		Active low signal indicating the End Of Frame for the requested Write data packet.
IP2Bus_MstWr_sop_n	IP Write LocalLink Intfc.	Input		Active low signal indicating the Start Of Payload for the requested read data packet. This signal is ignored by the Master.
IP2Bus_MstWr_eop_n	IP Write LocalLink Intfc.	Input		Active low signal indicating the End Of Payload for the requested read data packet. An EOP delimiter is required when a LocalLink footer is needed for the SG operations. This signal is ignored by the Master.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
IP2Bus_MstWr_rem(0 : **C_REM_WIDTH**-1)	IP Write LocalLink Intfc.	Input		One, Three, or eight bit bus indicating the valid Bytes in the associated LocalLink data bus (Bus2IP_MstRd_d). It is sampled when any of the LocalLink delimiter signals are asserted.
IP2Bus_MstWr_d(0 : **C_MPLB_NATIVE_DWIDTH**-1)	IP Write LocalLink Intfc.	Input		Write Data bus. A data packet is started with the assertion of the Bus2IP_MstWr_sof_n and ended with the assertion of the Bus2IP_MstWr_eof_n.
IP2Bus_MstWr_src_rdy_n	IP Write LocalLink Intfc.	Input		Active low signal indicating the Primary LocalLink Write interface is able to accept data from IP client. A data transfer occurs when both the Bus2IP_MstWr_dst_rdy_n and the IP2Bus_MstWr_src_rdy_n are asserted on the rising edge of the Write LocalLink synchronization clock.
IP2Bus_MstWr_src_dsc_n	IP Write LocalLink Intfc.	Input		Active low signal indicating to the Write interface that the IP Client is discontinuing the transfer.
Bus2IP_MstWr_dst_rdy_n	IP Write LocalLink Intfc.	Output	'1'	Active low signal indicating to the IP Client that the Write interface is ready to accept data. A data transfer occurs when both the Bus2IP_MstWr_dst_rdy_n and the IP2Bus_MstWr_src_rdy_n are asserted on the rising edge of the Write LocalLink synchronization clock.
Bus2IP_MstWr_dst_dsc_n	IP Write LocalLink Intfc.	Output	'1'	Active low signal indicating to the IP Client that the Write interface is discontinuing the transfer.

Table 1: **PLBV46 Master I/O Signal Description**

Signal Name	Interface	Signal Type	Init Status	Description
Bus2IP_MstWr_WrCnt(0 : **C_WRFIFO_WRCNT_WIDTH**-1)	IP Write LocalLink Intfc.	Output	All '0'	This bus reflects the number of words that are loaded in the Write FIFO. It is synchronous to the Write LocalLink synchronization clock. This bus may be used by the User IP to monitor the occupancy of the WrFIFO.
IP2Bus_MstWr_Occupancy(0:**C_WRFIFO_OCCUPANCY_WIDTH**-1)	IP Write LocalLink Intfc.	Input		This field is only used by the Write Controller when the Write FIFO has been omitted from the Write Controller and the parameter **C_WRFIFO_OCCUPANCY_WIDTH** is 2 or greater. This bus is driven by the User IP and indicates the available data in a User IP FIFO that the Write Controller will be receiving via the Write LocalLink. The Write Controller will stop posting Write requests to the PLB if the value of this bus drops below a parameterized threshold. If the User IP does not implement a FIFO, then the bus should be driven with all bits asserted indicating maximum data available at all times or assign the **C_WRFIFO_OCCUPANCY_WIDTH** parameter to a value of 1.

Notes:

1. This signal's function and timing is defined in the IBM® **128-Bit Processor Local Bus Architecture Specification Version 4.6.**
2. The LocalLink REM bus has parameterizable assertion polarity. The init state of the REM bus that is output by the Master is the opposite of the active assertion state.

Design Parameters

The PLBV46 Master has interface features and functional features that are parameterizable via VHDL Generic assignments. The names and description of thee parameters are shown in Table 2.

Table 2: **PLBV46 Master Design Parameters**

Feature/Description	Parameter Name	Allowable Values	Default Values	VHDL Type
IP Command Interface				
This parameter sets the needed bit width of the IP2Bus_MstRd_Length and IP2Bus_MstWr_Length input ports.	C_LENGTH_WIDTH	8 to 30	12	Integer
PLB Master Request Interface				
This parameter sets the initial priority for any request issued by this PLB Master.	C_START_PRIORITY	0 to 4 0,1,2, 3 = starting PLB request priority 4 = Priority fixed at 0	4	Integer
This parameter specifies a time-out interval (in number of PLB clocks) that, when expired, will cause this Master to bump (increase) the request priority value (during an active request). The actual time interval will be 2 greater than the value assigned to this parameter.	C_PRIORITY_BUMP_TIMEOUT	0 to 127 0 = Priority Bumping logic omitted 1-127 = number of PLB clocks set for bumping interval (plus 2 clocks for internal registering)	0	Integer
Write LocalLink FIFO Properties				
This Parameter specifies the Write LocalLink interface clocking relationship to the PLB Bus Clock	C_WR_LLINK_IS_ASYNC	0 = Write LocalLink Interface is synchronous to PLB Clock 1 = Write LocalLink Interface is asynchronous to PLB Clock	0	Integer
This Parameter specifies the use of SRL16 based memory core for the Write FIFO if the Write LocalLink backend is synchronous to the Bus Clock.	C_USE_SRL_WRFIFO	0 = Write FIFO will use BRAM memory core 1 = Write FIFO will use SRL16 based memory core	0	Integer
This parameter sets the depth of the Write FIFO	C_WRFIFO_DEPTH	16 to 16,384 (must be 2^N-1 for Async fifo; must be 2^N for Sync FIFO; must be 16, 32, or 64 for SRL fifo).	512	integer

Table 2: **PLBV46 Master Design Parameters**

Feature/Description	Parameter Name	Allowable Values	Default Values	VHDL Type
This parameter sets the width of the Bus2IP_MstWr_WrCnt bus.	C_WRFIFO_WRCNT_WIDTH	4 to 15 Set to log2(**C_WRFIFO_DEPTH**) +1 for Sync and SRL FIFO Set to log2(**C_WRFIFO_DEPTH**) for async FIFO	10	integer
Reserved: This parameter sets the width of the IP2Bus_MstWr_Occupancy input port.	C_WRFIFO_OCCUPANCY_WIDTH	1	1	integer
Read LocalLink FIFO Properties				
This Parameter specifies the Read LocalLink interface clocking relationship to the PLB Bus Clock	C_RD_LLINK_IS_ASYNC	0 = Read LocalLink Interface is synchronous to PLB Clock 1 = Read LocalLink Interface is asynchronous to PLB Clock	0	Integer
This Parameter specifies the use of SRL16 based memory core for the Read FIFO if the Read LocalLink backend is synchronous to the Bus Clock.	C_USE_SRL_RDFIFO	0 = Read FIFO will use BRAM memory core 1 = Read FIFO will use SRL16 based memory core	0	Integer
This parameter sets the depth of the Read FIFO	C_RDFIFO_DEPTH	16 to 16,384 (must be 2^N-1 for Async fifo; must be 2^N for Sync FIFO; must be 16,32, or 64 for SRL fifo).	512	integer
This parameter sets the width of the Bus2IP_MstRd_RdCnt bus.	C_RDFIFO_RDCNT_WIDTH	0 to 15 Set to log2(C_RDFIFO_DEPTH) +1 (See table note 2)	10	integer
Reserved: This parameter sets the width of the IP2Bus_MstRd_Vacancy input port.	C_RDFIFO_VACANCY_WIDTH	1	1	integer

Table 2: **PLBV46 Master Design Parameters**

Feature/Description	Parameter Name	Allowable Values	Default Values	VHDL Type
Trigger Threshold for the number of entries in the Read FIFO needed to cause a Local Link transfer to be initiated.	C_RDFIFO_LLTRANS_THRES	1 to 16384 **Cannot not exceed C_RDFIFO_DEPTH value**.	8	Integer
LocalLink REM Properties				
Set the bit width of the LocalLink REM bus in the Master LocalLink interfaces.	C_REM_WIDTH	1 to 16 For encoded REM use log2(C_MPLB_NATIVE_DWIDTH/8) For Mask REM, use (PLB_DWIDTH/8)	4	Integer
Specifies the type of representation of the values presented on the REM bus.	C_REM_CODING	1, 2 1 = Encoded REM 2= mask REM	2	Integer
Specifies the assertion polarity of the LocalLink REM transfer qualifiers.	C_REM_POLARITY	0 = Active Low 1 = Active High	0	Integer
PLBV46 Interface Properties				
Width of the PLB Address Bus	C_MPLB_AWIDTH	32	32 (1)	integer
Width of the PLB Data Bus	C_MPLB_DWIDTH	32, 64, 128 Must be greater than or equal to C_MPLB_NATIVE_DWIDTH.	128	integer
Specifies the Master's internal Data Bus Width (bits)	C_MPLB_NATIVE_DWIDTH	32.64,128	64	integer
Specifies the Smallest Slave on the PLB that the Master will encounter	C_SMALLEST_SLAVE	32.64,128	32	integer
Allows the User to override the automatic inclusion of Conversion Cycles and Burst Length Expansion functionality	C_INHIBIT_CC_BLE_INCLUSION	0,1 0 = Allow Automatic Inclusion 1 = Inhibit CC and BLE Inclusion	0	integer
Target Device Family Properties				
Xilinx FPGA Family	C_FAMILY	See C_FAMILY parameter values.		string

Note:

1. These parameters should not be changed from the default value.
2. The Master design incorporates additional logic on the Read side of the LocalLink Async FIFOs such that one extra data value is available. Thus the reported read count value can indicate a value that is one more than expected based on the parameterized depth of the FIFO. (i.e. An async FIFO of depth 511 can have a read count value of 512 when the FIFO is filled to capacity.

PLB Master Parameter Detailed Descriptions

C_LENGTH_WIDTH

This is an integer parameter has a range of 8 to 30 and specifies the width (in bits) of the IP2Bus_MstRd_Length and IP2Bus_MstWr_Length input port. The number of bits will limit the maximum read or write length (in bytes) that can be specified via the read and write command interfaces. For example, the default value of 12 will limit the requested read or write length to 4096 bytes.

C_START_PRIORITY

This integer parameter has a range of 0 to 4. It specifies to the Master what the initial assertion of the M_priority output bus will be for any request posted by this Master. The M_priority bus is used by the PLB Arbiter during request arbitration of the PLB. The higher priority request is granted access ahead of lower priority requests. A value of 0 is lowest priority and a value of 3 is highest priority. A value of 4 will fix the priority at 0 and priority bumping will be disabled. If enabled, the priority bumping logic which will increase request priority from this starting value at predefined time intervals (see **C_PRIORITY_BUMP_TIMEOUT**) until highest priority (3) is reached or the Master request is acknowledged.

C_PRIORITY_BUMP_TIMEOUT

This integer parameter has a range of 0 to 127 and specifies a priority bumping time-out period (in PLB clocks) that is allowed to elapse during a request assertion by this Master. If the time-out expires, the Master will increment the asserted request priority (via M_priority) to the next priority level and reinitialize the priority bumping timeout counter. This sequence continues until the request is acknowledged, the request is aborted, or highest priority is reached. If a value of 0 is assigned to this parameter, the priority bumping logic is removed. Request priority will be fixed to that specified by the **C_START_PRIORITY** parameter.

Note that there is an additional 2 clocks of timeout realized due to internal registering and counter management.

C_WR_LLINK_IS_ASYNC

This integer parameter specifies the needed clocking relationship of the Write LocalLink interface and the PLB Bus clock. When set to 1, the Write LocalLink Interface must be supplied with a User clock and an Asynchronous Core Generator system FIFO will be instantiated for the Write FIFO. If set to 0, the Write LocalLink Interface will use the PLB Bus clock and a Synchronous Core Generator system FIFO or SRL based FIFO will be instantiated for the Write FIFO. Note that asynchronous operation induces more latency in initiating and completing write transfers.

C_USE_SRL_WRFIFO

If the Write LocalLink interface is synchronous to the PLB Clock (see **C_WR_LLINK_IS_ASYNC**), then this parameter allows the User to select the use of an SRL16 based implementation of the Write FIFO. This will eliminate the use of BRAMs for the FIFO. If selected, the **C_WRFIFO_DEPTH** can be 16, 32, or 64.

C_WRFIFO_DEPTH

This integer parameter applies to the Write Controllers. This integer parameter specifies the storage depth of the WrFIFO. The width of the WrFIFO is automatically set to the value of **C_MPLB_NATIVE_DWIDTH** plus LocalLink packet delimiters (2) and LocalLink REM (0 to 16 bits

depending on rem parameterization). Synchronous and asynchronous FIFOs are limited to a maximum depth of 16K entries.

1. For asynchronous Write FIFOs, this value must be set to a value that is equal to 2^n-1 where n is an integer value.
2. For synchronous Write FIFOs, this value must be set to a value that is equal to 2^n where n is an integer value.
3. For SRL FIFOs, only values of 16, 32, and 64 are allowed. A depth of 32 or greater is recommended.

C_WRFIFO_WRCNT_WIDTH

This integer parameter applies to the Write Controller. This integer parameter specifies the size of the Bus2IP_MstWr_WrCnt bus which is internally sourced from the WrFIFO write count output bus and output to the IP Client as a sideband signal bus with Write LocalLink interface. The value has a range of 1 to log2(**C_WRFIFO_DEPTH**)+1. The Bus2IP_MstWr_WrCnt bus is populated from the most significant bit towards the least significant bit.

Notes:

1. **For asynchronous Write FIFOs, this value is set to $\log_2$(C_WRFIFO_DEPTH).**
2. For synchronous Write FIFOs (Core Generatpr tool and SRL), this value is set to $\log_2$(C_WRFIFO_DEPTH)+1.

C_WRFIFO_OCCUPANCY_WIDTH

This parameter is reserved and should be set to 1.

C_RD_LLINK_IS_ASYNC

This integer parameter applies to the Write Controllers. This integer parameter specifies the needed clocking relationship of the Read LocalLink interface and the PLB Bus clock. When set to 1, the Read LocalLink Interface must be supplied with a User clock and an Asynchronous Core Generator system FIFO will be instantiated for the Read FIFO. If set to 0, the Read LocalLink Interface will use the PLB Bus clock and a Synchronous Core Generator system FIFO will be instantiated for the Read FIFO. Note that asynchronous operation induces more latency in initiating and completing read transfers.

C_USE_SRL_RDFIFO

If the LocalLink interface is synchronous to the PLB Clock as specified by the **C_RD_LLINK_IS_ASYNC**, then this parameter allows the User to select the use of an SRL16 based implementation of the Write FIFO. This will eliminate the use of BRAMs for the FIFO. If selected, the **C_RDFIFO_DEPTH** can be 16, 32, or 64.

C_RDFIFO_DEPTH

This parameter specifies the storage depth of the RdFIFO in the Read Controller. The width of the RdFIFO is automatically set to the value of **C_MPLB_NATIVE_DWIDTH** plus LocalLink packet delimiters (2) and LocalLink rem (0 to 16 bits depending on rem parameterization). Synchronous and asynchronous FIFOs are limited to a maximum depth of 16K entries.

Notes:

1. For asynchronous Read FIFOs, this value must be set to a value that is equal to 2^n-1 where n is an integer value.
2. For synchronous Read FIFOs, this value must be set to a value that is equal to 2^n where n is an integer value.
3. For SRL FIFOs, only values of 16, 32, and 64 are allowed. It is recommended that a depth of 32 and greater is used.

C_RDFIFO_RDCNT_WIDTH

This integer parameter applies to the Read LocalLink Backend. It specifies the size of the Bus2IP_MstRd_RdCnt bus which is internally sourced from the RdFIFO read count bus. The value has

a range of 1 to log2(**C_RDFIFO_DEPTH**)+1. The Bus2IP_MstRd_RdCnt bus is populated from the most significant bit towards the least significant bit.

Notes:

1. **For asynchronous Read FIFOs, this value is set to log_2(C_RDFIFO_DEPTH+1).** The Master design incorporates additional logic on the Read side of the LocalLink Async FIFOs such that one extra data value is available. Thus the reported read count value can indicate a value that is one more than expected based on the parameterized depth of the FIFO. (I.E. An async FIFO of depth 511 can have a read count value of 512 when the FIFO is filled to capacity and kept full by the writing logic.
2. For synchronous Read FIFOs (Core Generator tool and SRL), this value is set to log_2(C_RDFIFO_DEPTH)+1.

C_RDFIFO_VACANCY_WIDTH

This parameter is reserved and should be set to 1.

C_RDFIFO_LLTRANS_THRES

This integer parameter specifies the minimum number of stored entries (occupancy) that must be present in the RdFIFO before the Read LocalLink will initiate a data transfer (asserting the Bus2IP_MstRd_src_rdy_n) on the Read LocalLink interface. The value entered for this parameter must be established by the User to minimize excessive throttling (due to the RdFIFO going Empty) by this Master when the Read Controller is loading data into the RdFIFO at a slower rate than the User IP is accepting it. This can occur if the User IP IP2Bus_MstRd_clk frequency is higher than the PLB clock or the PLB Slave providing the read data is throttling the PLB transfer. This threshold is overridden when the Read Controller loads the RdFIFO with an EOP flag (EOP indicates the last data word of a requested data transfer).

Note:

1. This parameter value must not exceed the value of C_RDFIFO_DEPTH-16 when C_RDFIFO_DEPTH is 16 or greater.
2. The use of Fixed Length Burst transfers that may start on an unaligned address boundary require this value to be set to no less than C_RDFIFO_DEPTH-17 to account for the starting Single data beat (to align to a Master Native Dwidth boundary) and an initial Fixed Length Burst of up to 16 data beats. This condition will require the RdFIFO depth to be set to at least 32 storage locations.

C_REM_WIDTH

This integer parameter specifies the size of the Bus2IP_MstRd_rem and Bus2IP_MstWr_rem bus bit widths. If the rem is not going to be used, then the parameter is assigned a value of 1. If rem is going to be used (the normal case), then the value assigned to this parameter must be derived from the value assigned to **C_MPLB_NATIVE_DWIDTH**. For an encoded representation, the assigned value must be log_2(**C_MPLB_NATIVE_DWIDTH**/8). For a Mask representation, the value assigned must be **C_MPLB_NATIVE_DWIDTH**/8.

Note: The LocalLink rem value is stored in the Read and Write FIFOs with the data values so FIFO memory resources may fluctuate depending on this parameter's assigned value.

C_REM_CODING

This integer parameter specifies the coding representation of the values placed on the Bus2IP_MstRd_rem and Bus2IP_MstWr_rem buses during data transmission. Either encoded representation (=1) or mask representation (=2) may be selected. The use of Encoded format has use restrictions in that single data beat transfers cannot always be accurately represented. In these cases, the User needs to drive the REM value to "all bytes valid" and use a Single Data Beat command with the desired Byte Enables set. Mask representation is the recommended use format.

C_REM_POLARITY

This integer parameter specifies the assertion polarity of the values placed on the Bus2IP_MstRd_rem and the interpretation of the IP2Bus_MstWr_rem buses during data transmission. Assignment of 0 selects active low assertion and an assignment of 1 selects active high assertion level.

C_MPLB_AWIDTH

This integer parameter is used to specify the number of address bits that are used in the PLB system. PLB V4.6 has 64 address bits defined. The only allowed value to be assigned is currently restricted to 32.

C_MPLB_DWIDTH

This integer parameter is used to indicate the width of the PLB data bus to which the Master will be attaching. This value may be set to 32, 64, and 128 but cannot be less than then the value assigned to C_MPLB_NATIVE_DWIDTH.

C_MPLB_NATIVE_DWIDTH

This integer parameter is used to size the internal data bus related components within the Master as well as the data width of the LocalLink data interface. The assigned value may be 32, 64, and 128 but cannot exceed the value assigned to C_MPLB_DWIDTH.

C_MPLB_SMALLEST_SLAVE

This integer parameter indicates the Native Bit Width of the smallest PLB Slave that the Master will encounter during read and write operations. This parameter is used to optimize write data bus steering logic and the automatic inclusion of Conversion Cycle and Burst Length Expansion functionality. Automatic CC and BLE functionality is included if the value for this parameter is less than the value assigned to C_MPLB_NATIVE_DWIDTH. The assigned value may be 32, 64, and 128 but cannot exceed the value assigned to C_MPLB_DWIDTH.

C_INHIBIT_CC_BLE_INCLUSION

This parameter allows the User to override the automatic inclusion of Conversion Cycle and Burst Length Expansion logic. an assigned value of 1 inhibits the CC and BLE inclusion, a 0 is default and allows for automatic inclusion if needed.

C_FAMILY

This parameter is defined as a string. It identifies the target FPGA technology for implementation of the PLB46 Master. This parameter is required by the LocalLink FIFO designs which utilize Xilinx BRAM primitives. The size and configuration of these primitives can vary from one FPGA technology family to another.

Table 3: **Xilinx Predefined Identifiers for FPGA Families**

PLB Master Category	Xilinx Identifier	Defined VHDL type	Assigned Value	Family Description
PLB Master Supported Families	See EDK Supported Device Families.	String	"spartan®3e"	See EDK Supported Device Families.
		String	"virte4"	
		String	"virtex5"	

Allowable Parameter Combinations

See detailed parameter descriptions in the section PLB Master Parameter Detailed Descriptions.

Parameter - Port Dependencies

Table 4: **PLBV46 Master Parameter-Port Dependencies**

Parameter Name	Affects Port	Depends	Relationship Description
C_MPLB_AWIDTH	IP2Bus_MstRd_Addr IP2Bus_MstWr_Addr		Directly sets the bit width of the command address bus
C_MPLB_DWIDTH	M_BE		The bit width of the Master output BE bus is set by dividing the parameter value by 8
C_MPLB_DWIDTH	M_wrDBus PLB_MRdDBus		Directly sets the bit width of the PLB Data bus attachments
C_MPLB_NATIVE_DWIDTH	IP2Bus_MstRd_BE IP2Bus_MstWr_BE		The bit width of the command BE bus is set by dividing the parameter value by 8
C_MPLB_NATIVE_DWIDTH	Bus2IP_MstRd_d IP2Bus_MstWr_d		Directly sets the bit width of the LocalLink Data bus attachments
C_REM_WIDTH	Bus2IP_MstRd_rem IP2Bus_MstWr_rem		Directly sets the bit width of the LocalLink REM bus attachments
C_LENGTH_WIDTH	IP2Bus_MstRd_Length		Directly sets the bit width of the Command interface Read Length bus attachments
C_LENGTH_WIDTH	IP2Bus_MstWr_Length		Directly sets the bit width of the Command interface Write Length bus attachments
C_RDFIFO_VACANCY_WIDTH	IP2Bus_MstRd_Vacancy		Directly sets the bit width of the input Read LocalLink interface Read FIFO vacancy. Set to constant 1.
C_RDFIFO_RDCNT_WIDTH	Bus2IP_MstRd_RdCnt		Directly sets the bit width of the output Read LocalLink interface Read FIFO read count bus
C_WRFIFO_OCCUPANCY_WIDTH	IP2Bus_MstWr_Occupancy		Directly sets the bit width of the input Write LocalLink interface Write FIFO occupancy. Set to constant 1.
C_WRFIFO_WRCNT_WIDTH	Bus2IP_MstWr_WrCnt		Directly sets the bit width of the output Write LocalLink interface Write FIFO write count bus

Register Descriptions

This Master does not incorporate any programmable or User accessible registers.

User Application Topics

Native Data Width and PLB Data Width

The PLBV46 Master has parameters (C_MPLB_DWIDTH) that allow the core to be configured for connection to three different PLB data bus widths (32, 64, and 128 bits). In addition, another parameter (C_MPLB_NATIVE_DWIDTH) allows the Master to be configured to one of three different internal data widths (32, 64, and 128 bits). The internal data width is also defined as the native data width. The native data width selection cannot exceed the PLB data bus width or alternatively, the PLB data bus width cannot be less than the assigned Native data width value of the Master. The native data width is also the width of the LocalLink interfaces with the IP Client.

In the case that the PLB data bus width is wider than the native data width, the Master incorporates mirroring logic to mirror the write data bus per the PLB requirements. This eliminates the need for this logic in the PLB Bus structure. Also per the PLB requirements, the Master will only use the appropriate slice of the Read Data bus that matches it's native data width. The remaining portion of the PLB Read Data bus is ignored.

Unaligned Read Transfers

The PLBV46 Master supports unaligned burst read transfer requests. An unaligned request is defined as having a starting address that is not aligned to a multiple of the Master's native data bus width and/or the specified Length qualifier causes a burst end condition where the last data beat is not a full native data bus width. Under these conditions, the Read Controller will detect unaligned start and end conditions of a burst and adapt automatically. An unaligned start condition will cause the Master to calculate and submit a single data beat PLB read request for the unaligned starting data bytes. The Read Controller then calculates and requests burst transfers using the full data bus width until completed. If an unaligned end condition is detected, the Read Controller will initiate PLB Burst transfers of the full data bus width until the last data beat. For the last data beat, the Read Controller will calculate and submit a single data beat PLB read request for the remaining unaligned starting data bytes. Thus, a general unaligned burst transfer request may be required to be mechanized on the PLB by the Read Controller as a starting single beat, a series of bus-width bursts, and then ending with a single data beat

Short burst requests are also automatically converted. For example, a read request for Fixed Length burst type may be requested via the Read Command interface and the associated Length and starting address values may be such that the transfer cannot be performed with a burst transfer request on the PLB. The Read Controller will automatically convert to single data beat requests on the PLB to complete the transfer.

Unaligned Write Transfers

The PLBV46 Master also supports unaligned burst write transfer requests. As with the Read side, unaligned write request is defined as having a resolved starting address that is not aligned to a multiple of the Master's native data bus width and/or the specified Length qualifier causes a burst end condition where the last data beat is not a full data bus width. The Write Controller side also has one additional unaligned transfer condition that is determined by the actual LocalLink REM values received from the User IP via the Write LocalLink interface. Under these conditions, the Write Controller Service will detect unaligned start and end conditions of a burst and adapt automatically.

An unaligned start condition will cause the Master to calculate and submit a single data beat PLB write request for the unaligned starting data bytes. The Write Controller then calculates and requests write burst transfers using the full data bus width until completed. If an unaligned end condition is detected, the Write Controller will initiate PLB Burst transfers of the full data bus width until the last data beat. For the last data beat, the Write Controller will calculate and submit a single data beat PLB write request for the remaining unaligned starting data bytes. Thus, a general unaligned burst transfer request may be required to be mechanized on the PLB by the Write Controller as a starting single beat, a series of bus-width bursts, and then ending with a single data beat

Short write burst requests are also automatically converted. For example, a write request for Fixed Length burst type may be requested via the Write Command interface and the associated Length and starting address values may be such that the transfer cannot be completed with burst transfer protocol on the PLB due to insufficient length. The Write Controller will automatically convert to single data beat requests on the PLB to complete the transfer.

The following rules must be used for the relationship between the IP2Bus_MstWr_Addr and the IP2Bus_MstWr_rem when an unaligned starting condition must be specified for a Fixed Length Burst. The User may use any one of the three relationships.

1. The starting address and the first valid byte indicated by the REM on the first data beat of the write must point to the same byte position,
2. or the starting REM can indicate the first valid byte of the transfer and the IP2Bus_MstWr_Addr indicates the aligned address of the data beat containing the first valid byte,
3. or the REM value indicates all bytes of the data beat are valid and the User supplied address on the IP2Bus_MstWr_Addr indicates where the first byte will be written.

Do not mix an unaligned starting address with an unaligned REM where the two do not point to the same byte position. The unaligned REM will override the specified starting address and can lead to data being written to an address that is before the starting address specified in the IP2Bus_MstWr_Addr input.

IP Master Bus Locking

The PLB Bus Locking feature is not supported by this Master.

PLB Master Interface Transaction Timing

This section of the document provides typical timing representation of various PLB transfers initiated by the PLBV46 Master. The User must realize that actual timing relationships will vary due to arbitration timing, dynamic PLB Slave response characteristics, and values assigned by the User for the Master's parameters.

The provided timing diagrams are indicative of a Master that is attached to a 64-bit PLB and has a matching native data width of 64 bits. The LocalLink Backends have been configured to be asynchronous to the PLB Clock and therefore the Client IP must provide the Read and Write LocalLink synchronization clocks. Operating in this mode also contributes to additional latency in the transaction timing of Client IP requests.

The threshold parameter C_RDFIFO_LLTRANS_THRES also can change transaction behavior for read operations. The User needs to review the parameter's definition in Table 2, "PLBV46 Master Design Parameters," on page 22 and in the detailed parameter description section see "PLB Master Parameter Detailed Descriptions" on page 25.

Single Data Beat Read Operation

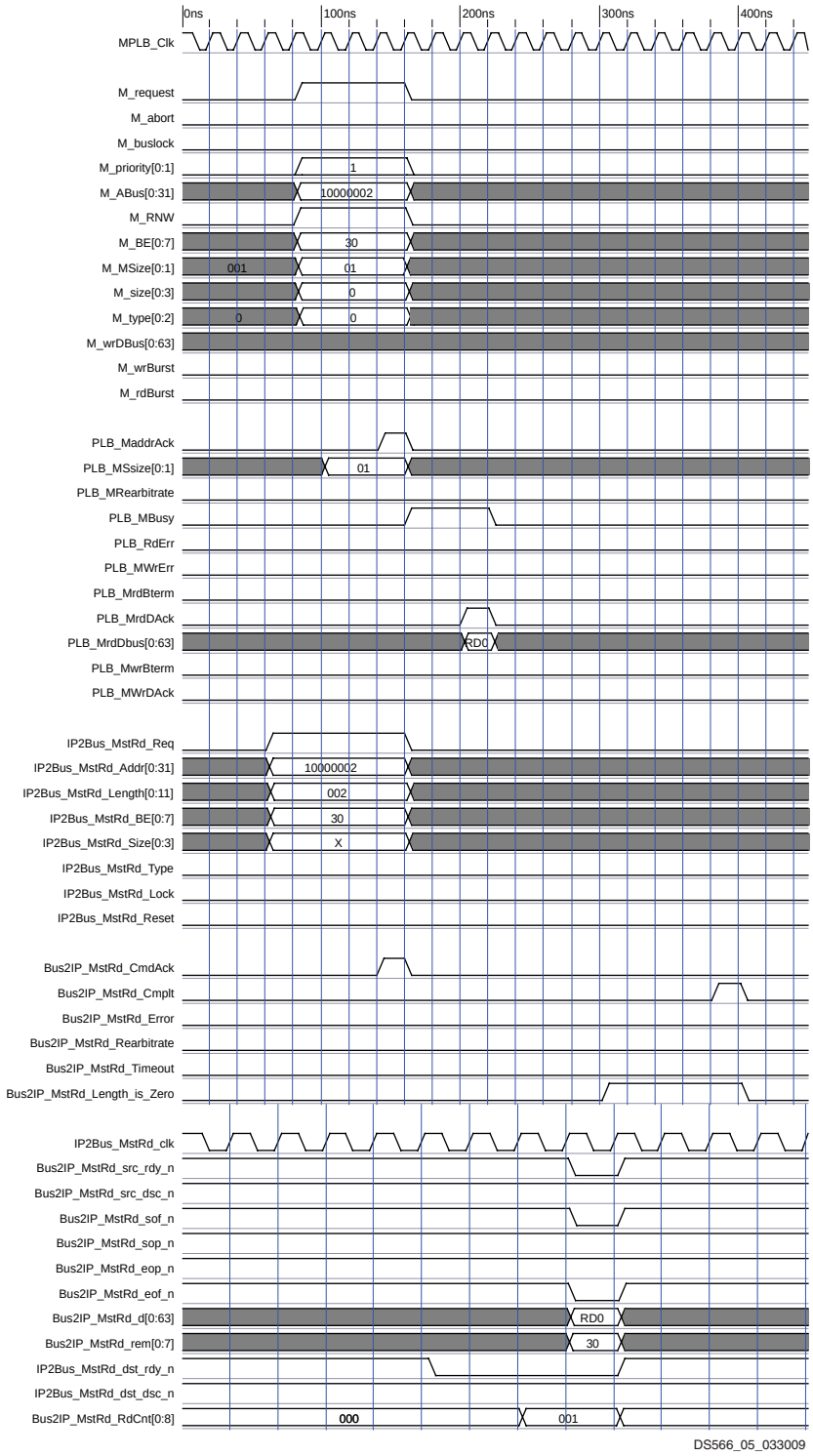

Figure 5: **PLBV46 Master Single Data Beat Read Timing**

Single Data Beat Write Operation

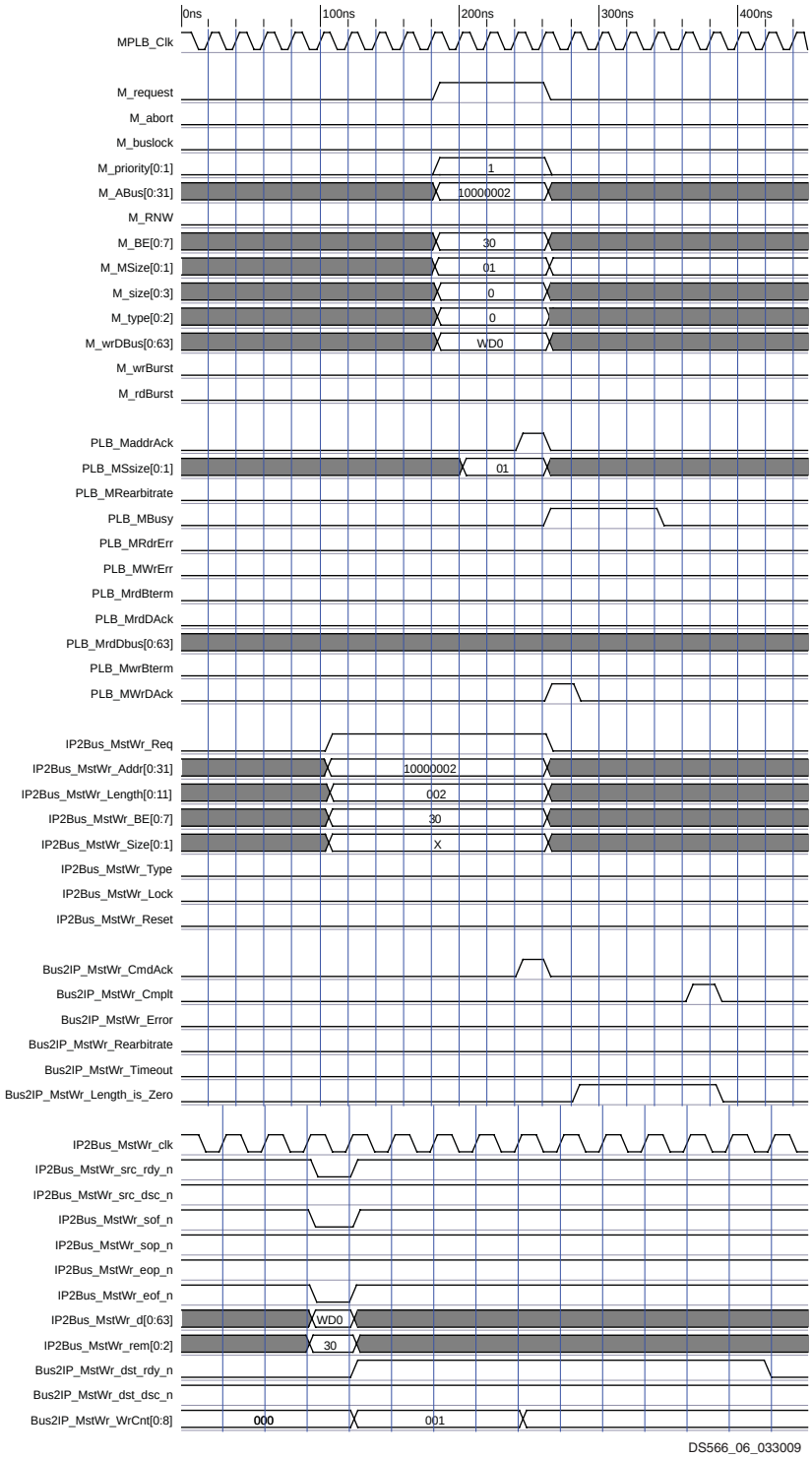

Figure 6: **PLBV46 Master Single Data Beat Write Timing**

Fixed Length Burst Read Operation

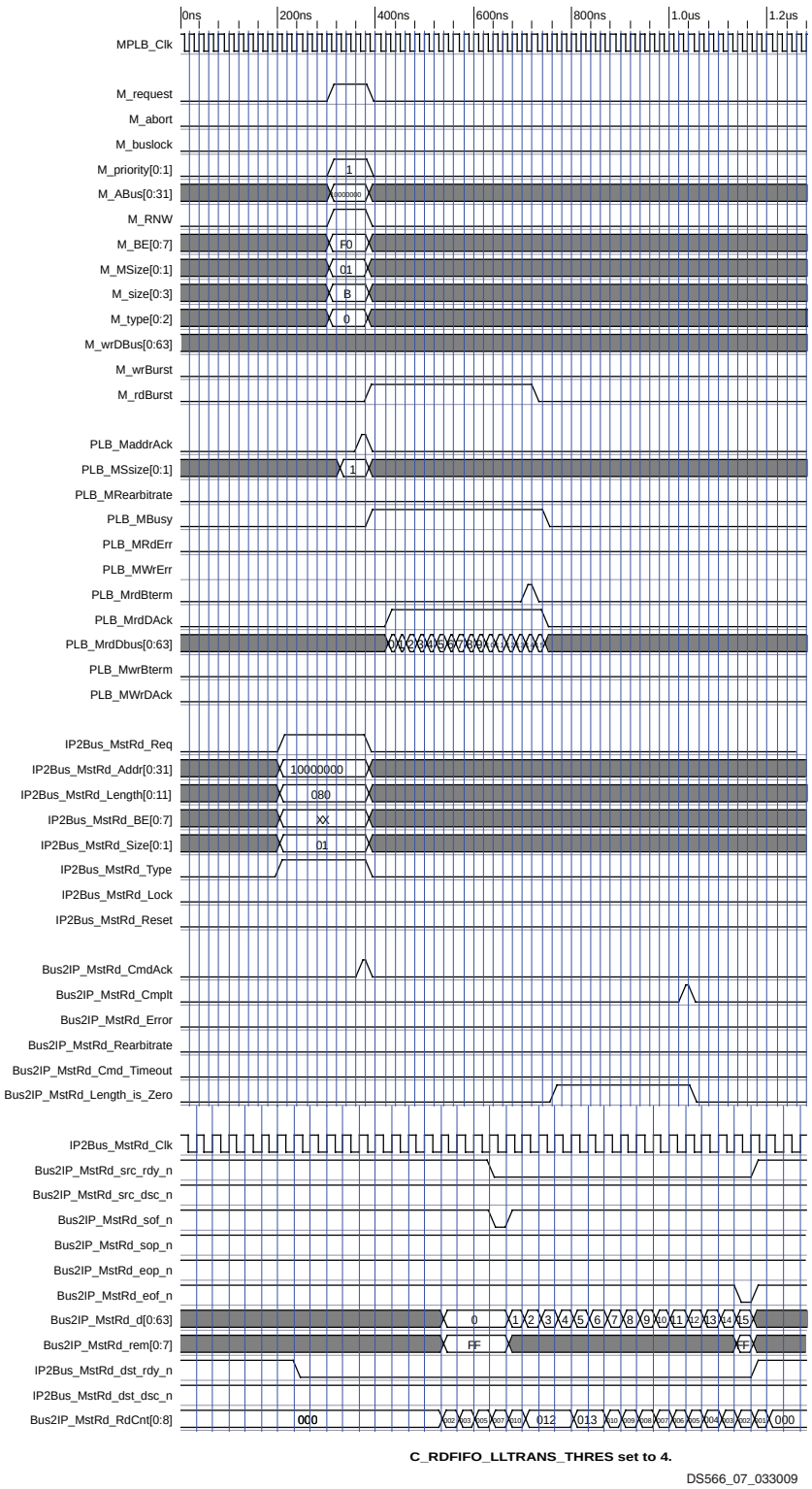

Figure 7: **PLBV46 Master Fixed Length Burst Read Timing**

Fixed Length Burst Write Operation

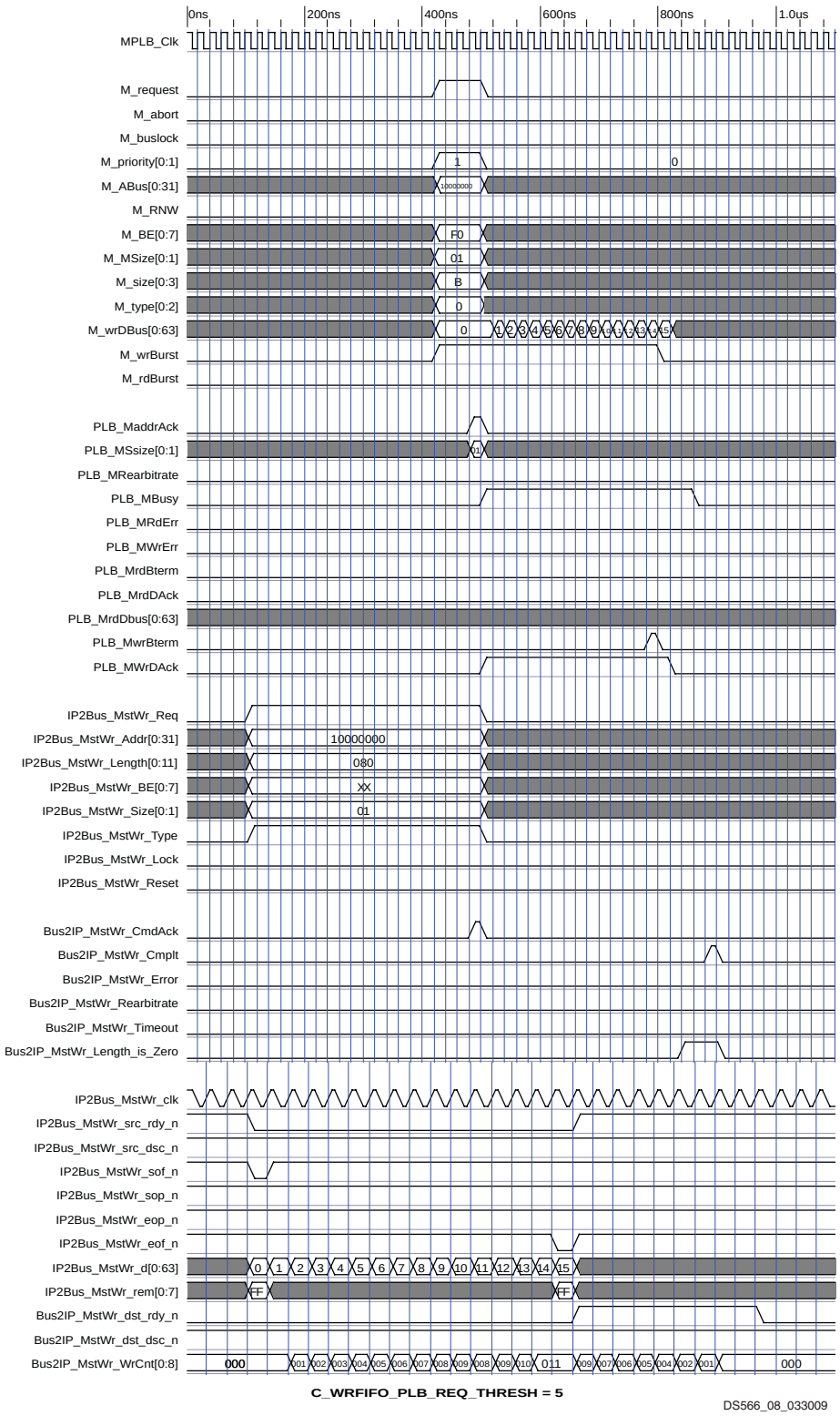

Figure 8: **PLBV46 Master Fixed Length Burst Write Timing**

Design Implementation

Target Technology

The target technology is an FPGA listed in EDK Supported Device Families.

Device Utilization and Performance Benchmarks

Since the PLB Master is a module that will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are just estimates. As the PLB Master is combined with other pieces of the FPGA design, the utilization of FPGA resources and timing will vary from the results reported here.

Three different groups of parameter settings have been chosen to represent a Full-up 128-bit, a Typical 64-bit, and a 32-bit (using SRL FIFOs) implementation of the PLBV46 Master. These settings are indicated in Table 6.

In order to analyze the PLB Master timing within an FPGA, a design wrapper was created that instantiated the PLB Master with registers on all of the module's inputs and outputs. This allowed a constraint to be placed on the clock nets for the design to yield more realistic timing results. The Xilinx ISE Synthesis tool (XST) was run on the wrapper and the synthesis results used for the information presented here. Some selected configurations of the PLBV46 Master were implemented and the resulting FPGA performance and resource utilization benchmarks are shown in Table 5..

Table 5: **Performance and Resource Utilization Benchmarks for the Virtex®-5 FPGA**

Parameter Settings Group	Device Resources				F_{MAX} (MHz)
	Slices	Slice Flip-Flops	4-input LUTs	BRAMs	F_{MAX}
Full-up 128-bit with Async BRAM based FIFOs 16K deep	2166	1940	3438	130 (36k) or 260 (18K)	125
Typical 64-bit with Sync BRAM based FIFOs 512 deep	855	849	1265	3 (36K) or 6 (18K)	130
32-bit with SRL FIFOs, 32 deep	512	502	986	0	140

Table 6: **Parameter Settings for Resource Utilization Estimates**

Parameter Name	Full-up 128-bit	Typical 64-bit	32-bit SRL FIFO
C_LENGTH_WIDTH	30	16	12
C_START_PRIORITY	1	1	4
C_PRIORITY_BUMP_TIMEOUT	64	16	0

Table 6: **Parameter Settings for Resource Utilization Estimates *(Contd)***

Parameter Name	Full-up 128-bit	Typical 64-bit	32-bit SRL FIFO
C_WR_LLINK_IS_ASYNC	1	0	0
C_USE_SRL_WRFIFO	0	0	1
C_WRFIFO_DEPTH	16383	512	32
C_WRFIFO_WRCNT_WIDTH	14	10	6
C_WRFIFO_OCCUPANCY_WIDTH	1	1	1
C_RD_LLINK_IS_ASYNC	1	0	0
C_USE_SRL_RDFIFO	0	0	1
C_RDFIFO_DEPTH	16383	512	32
C_RDFIFO_RDCNT_WIDTH	14	10	6
C_RDFIFO_VACANCY_WIDTH	1	1	1
C_RDFIFO_LLTRANS_THRES	16	8	4
C_REM_WIDTH	16	8	4
C_REM_CODING	2	2	2
C_REM_POLARITY	1	32	32
C_MPLB_AWIDTH	32	32	32
C_MPLB_DWIDTH	128	64	32
C_MPLB_NATIVE_DWIDTH	128	64	32
C_MPLB_SMALLEST_SLAVE	32	32	32
C_INHIBIT_CC_BLE_INCLUSION	0	0	0
C_FAMILY	"virtex5"	"virtex5"	"virtex5"

Specification Exceptions

The following PLB Master features are not supported by the PLBV46 Master design.

- Parity
- Indeterminate Length Bursts
- Fixed Length Bursts of length 17 to 256 data beats
- Cacheline transfers
- Bus Locking

The following LocalLink features are not supported:

- Channelization
- Start of payload and End of Payload assertions
- Parity
- Multiple REM bus per LocalLink

Reference Documents

1. IBM CoreConnect **128-Bit Processor Local Bus, Architectural Specification** (v4.6).
2. Xilinx **SP026 PLBV46 Interface Simplifications**
3. Xilinx **SP006 LocalLink Interface Specification**.

Revision History

Date	Version	Revision
07/17/07	1.0	Initial Xilinx release.
7/28/08	1.2	Added QPro Virtex-4 Hi-Rel and QPro Virtex-4 Rad Tolerant FPGA support.
4/24/09	1.3	Replaced references to supported device families and tool name(s) with hyperlink to PDF file.

Notice of Disclaimer