OpenCores.Org

PLBv46 to Wishbone Bridge

Author: Mark Sasten

mark.sasten@xilinx.com

Rev. 0.1

January 20, 2009

This page has been intentionally left blank.

Revision History

Rev.	Date	Author	Description
0.1	7/30/08	Mark Sasten	First Draft
0.2	1/20/09	Mark Sasten	Added ports and diagrams.

Contents

1

Introduction

This document provides specifications for the PLBv46™ to Wishbone™ bridge that has been made to work inside Xilinx EDK projects. Many embedded projects require simple interfaces to different bus standards. One very popular bus interface is the Wishbone bus used by many OpenCores.org peripherals. This bridge allows Microblaze and PowerPC applications to utilize many of these open source cores.

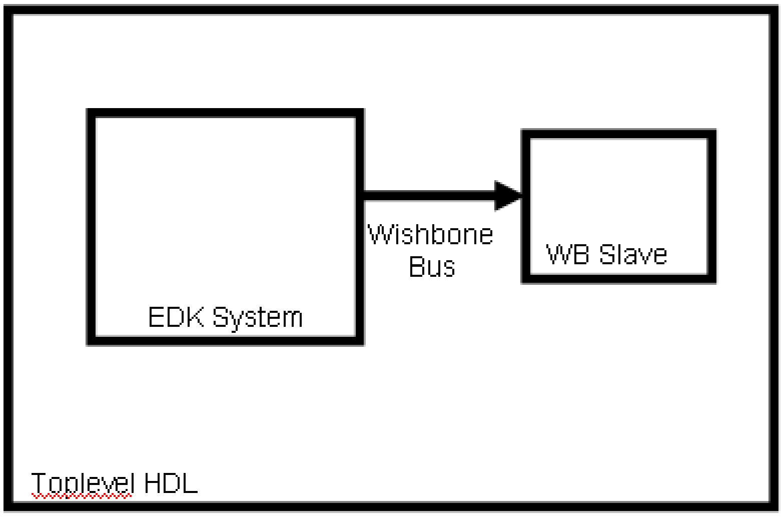
Features:

- PLBv46 Slave Attachment (non-bursting)
  - Native 32-bit slave interface to PLBv46 bus.
  - 32-bit master interface to Wishbone bus.
- Directly integrated into EDK tools as a custom pcore.
  - Comes as synthesizable VHDL
  - Microprocessor Peripheral Definition (MPD) file provided.
  - Compiles with PLBv46 subset of IBM CoreConnect™ Bus Standard.
- Supports
  - Handling of retries.
    - User can set the retry wait time.
    - User can set number of times to retry transaction.
    - Result of unsuccessful retry is a PLBv46 bus error ack.
  - Handling of Bus Errors
    - User can set how long to wait for a bus-timeout resulting from no-acknowledgement from a Wishbone transaction.
    - Results in an immediate PLBv46 Bus Error.
    - Will translate Wishbone Bus Errors to PLBv46 bus.

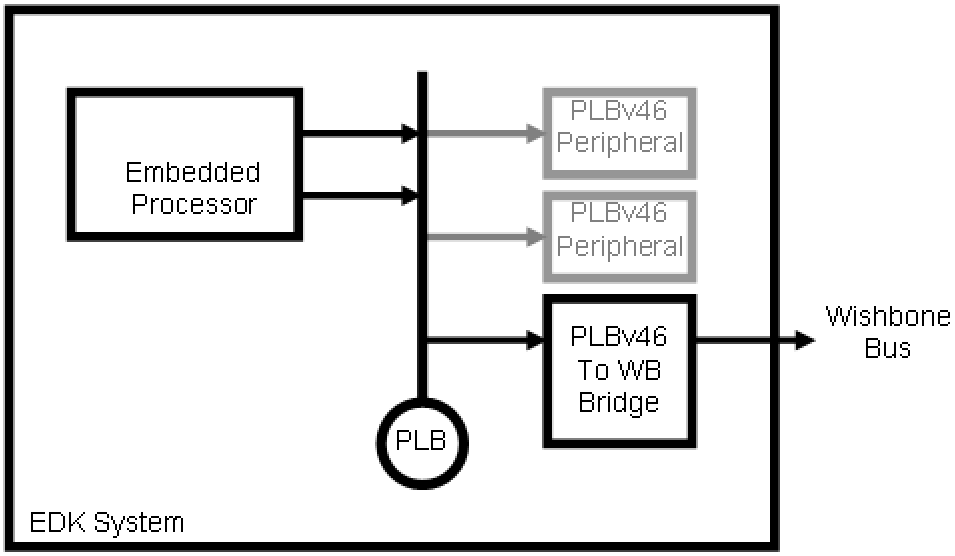
2

Architecture

The typical use of the PLBv46 to Wishbone Bridge is to allow the integration of Wishbone Slave peripherals into a Xilinx embedded design application. This could include the utilization of either the Microblaze or PowerPC processors. The standard method for embedding the PLBv46 to Wishbone bridge is to include it into the EDK design and to bring out the Wishbone Bus to an upper level HDL source to connect to the slave peripherals as shown below.



The EDK System is shown in the figure below.



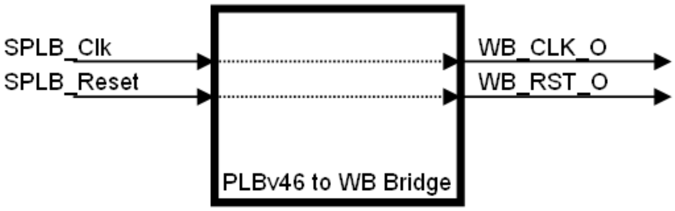
3

Operation

This section describes the operation of the core. Specific sequences, such as startup sequences, as well as the modes and states of the block should be described.

Clocking and Reset

Wishbone bridge clock and reset signals are derived from the PLBv46 clock and reset inputs as shown below. They are a direct mapping from the PLB clock source to the wishbone clock. No additional global buffering is required. This means that the WB_CLK_O domain will be at the same clock frequency as the SPLB_Clk domain. No additional clock synchronization is required by the bridge.

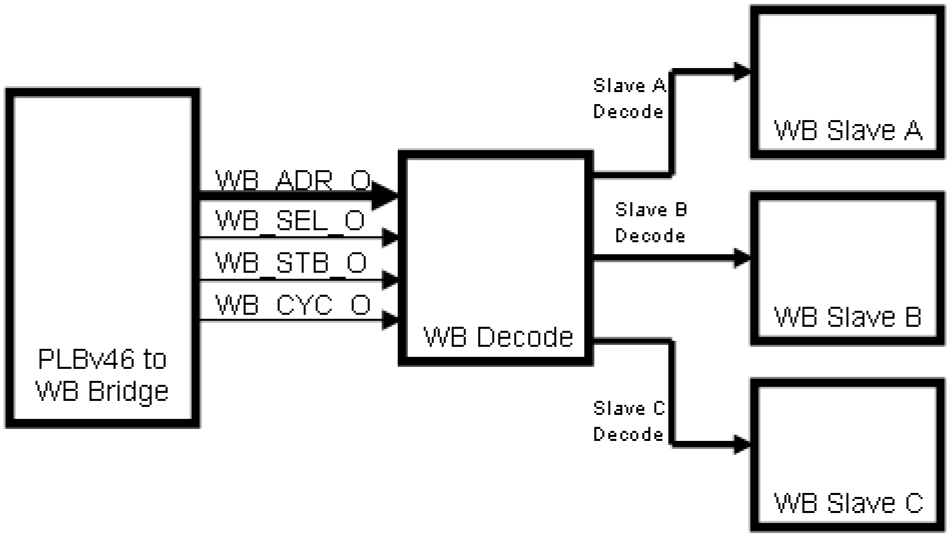


Address Decoding

There are two phases to the address decoding for this block. The first is handled by the bridge to map the bridge to the PLB address range. This is handled by the EDK tools when setting the C_MEM0_BASEADDR and C_MEM0_HIGHADDR in the MHS file.

The second address decoding is performed by user defined logic on the Wishbone side of the bridge. This requires the user to architect the Wishbone bus with the desired peripherals.

The following figure illustrates a typical decoder that will process the Wishbone signals coming out of the PLBv46 to Wishbone bridge and produce signals to decode the individual slave devices.



4

IO Ports

This section specifies the core IO ports. There are several ports defined by the PLBv46 specification that are not used for this bridge application. This is either due to the Xilinx PLBv46 simplification or because this core does not support burst transfers.

PLBv46 Interface Ports

Port	Width	Dir	Description
SPLB_Clk	1	I	
SPLB_Rst	1	I	
PLB_ABus	32	I	PLB Address Bus
PLB_PAValid	1	I	PLB Primary Address Valid
PLB_masterID	4*	I	PLB current master identifier
PLB_RNW	1	I	PLB read not write
PLB_BE	4	I	PLB byte enables
PLB_size	4	I	PLB size of requested transfer
PLB_type	3	I	PLB transfer type
PLB_wrDBus	32	I	PLB write data bus
Sl_addrAck	1	O	Slave address acknowledge
Sl_SSize	2	O	Slave data bus size
Sl_wait	1	O	Slave wait
Sl_rearbitrate	1	O	Slave bus rearbitrate
Sl_wrDAck	1	O	Slave bus data acknowledge
Sl_wrComp	1	O	Slave write transfer complete
Sl_rdDBus	32	O	Slave read data bus
Sl_rdDAck	1	O	Slave read data acknowledge
Sl_rdComp	1	O	Slave read data transfer complete
Sl_MBusy	*	O	Slave busy
Sl_MWrErr	*	O	Slave write error
Sl_MRdErr	*	O	Slave read error

Table: List of PLBv46 ports

Unused PLBv46 Interface Ports

Port	Width	Dir	Description
PLB_UABus	32	I	PLB upper address bus
PLB_SAValid	1	I	PLB secondary address valid
PLB_rdPrim	1	I	PLB secondary to primary read request indicator
PLB_wrPrim	1	I	PLB secondary to primary write request indicator
PLB_abort	1	I	PLB abort bus request
PLB_busLock	1	I	PLB bus lock
PLB_MSize	2	I	PLB data bus width indicator
PLB_TAttribute	16	I	PLB transfer attribute
PLB_lockErr	1	I	PLB lock error
PLB_wrBurst	1	I	PLB burst write transfer
PLB_rdBurst	1	I	PLB burst read transfer
PLB_wrPendReq	1	I	PLB pending bus write priority
PLB_rdPendReq	1	I	PLB pending bus read priority
PLB_wrPendPri	2	I	PLB pending bus write request
PLB_rdPendPri	2	I	PLB pending bus read request
PLB_reqPri	2	I	PLB pending write request priority
Sl_wrBTerm	1	O	Slave terminate write burst transfer
Sl_rdWdAddr	4	O	Slave read word address
Sl_rdBTerm	1	O	Slave read burst terminate
Sl_MIRQ	*	O	Master interrupt request

Table: List of Unused PLBv46 ports

Wishbone Interface Ports

Port	Width	Dir	Description
WB_CLK_O	1	O	System clock output (SYSCON)
WB_RST_O	1	O	System reset output (SYSCON)
WB_ADR_O	32	O	Address output array
WB_DAT_O	32	O	Data from bridge
WB_SEL_O	4	O	Byte select signals
WB_CYC_O	1	O	Valid bys cycle output
WB_LOCK_O	1	O	Locked transfer
WB_STB_O	1	O	Valid data transfer cycle
WB_WE_O	1	O	Write enable output
WB_DAT_I	32	I	Data to bridge
WB_ACK_I	1	I	Transfer acknowledgement
WB_ERR_I	1	I	Bus cycle error input
WB_RTY_I	1	I	Retry signaled to bridge.

5

Transaction Waveforms

TO BE ADDED