Reference System: MCH OPB DDR SDRAM with OPB Central DMA

XILINX®

XAPP912 (v1.3) June 1, 2007

Author: Casey Cain

Abstract

This application note describes a reference system that demonstrates the use of the Multi-CHannel (MCH) On-chip Peripheral Bus (OPB) Double Data Rate (DDR) Synchronous DRAM (SDRAM) controller in a MicroBlaze™ processor system. Two MCH ports of the MCH OPB DDR SDRAM controller are connected to the cache ports of the MicroBlaze processor. The ports are configured with the Xilinx Cachelink (XCL) protocol which allows for efficient cacheline accesses by the processor. The OPB Central DMA controller is included in this system to generate OPB sequential address (burst) transactions. The software application provided with this reference system tests the capability of the MCH OPB DDR SDRAM core to handle cacheline accesses from the MicroBlaze processor and OPB burst transactions from the OPB Central DMA controller. The MCH OPB DDR SDRAM controller can be implemented on Virtex™-II Pro, Spartan™-3, and Virtex-4 FPGAs with boards that support DDR SDRAM.

This application note describes how to set up MicroBlaze parameters for caching, the clocking structure for the MCH OPB DDR SDRAM, and parameters for OPB burst transactions from the OPB Central DMA controller. This reference system is targeted for the Xilinx SP305 Spartan-3 development board.

Included Systems

Included with this application note is the reference system for Xilinx SP305 Spartan-3 Development board.The reference system is available for download at:

www.xilinx.com/bvdocs/appnotes/xapp912.zip

Introduction

Many software applications using FPGAs are executed from external memory. Executing code from external memory is faster when caching is used. Instead of caching over the OPB, the MicroBlaze processor can be configured to cache instructions and data using the XCL protocol. Using the MicroBlaze XCL caches allows instruction fetches and data cache accesses to execute independently of OPB transfers, thereby improving performance and reducing the traffic over the OPB. Specialized memory controllers, like the MCH OPB DDR SDRAM, provide channel interfaces that can be connected to the cache ports of the MicroBlaze processor and configured to use the XCL protocol. The reference system described in this application note is built for the SP305 board with the MicroBlaze processor configured for caching using the XCL protocol and the MCH OPB DDR SDRAM controller configured to use a DDR SDRAM clock that is two times the frequency of the OPB clock.

Hardware and Software Requirements

The hardware and software requirements are:

- Xilinx SP305 Spartan-3 Development board
- Xilinx Platform USB cable or Parallel IV programming cable
- RS232 serial cable and serial communication utility (HyperTerminal)
- Xilinx Platform Studio 9.1.01i
- Xilinx Integrated Software Environment (ISE™) 9.1.03i

Reference System Specifics

This reference system is built for the SP305 Rev B development board. The system uses the MicroBlaze processor with 8 KB for both the instruction cache and the data cache located on main memory. The reference system also includes MCH OPB DDR SDRAM, OPB UART Lite with interrupts, OPB GPIO to control four LEDs, and OPB Central DMA. The DDR clock is set to run at 100 MHz and the OPB clock is set to run at 50 MHz. The reference system block diagram is shown in Figure 1 and the address map of the system is listed in Table 1.
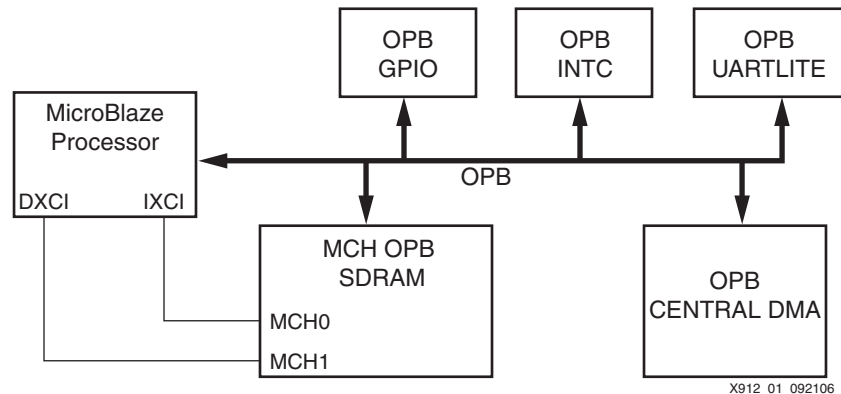
Block Diagram

Figure 1: **Reference System Block Diagram**

Address Map

Table 1: **Reference System Address Map**

Peripheral	Instance	Base Address	High Address
opb_mdm	debug_module	0x84000000	0x8400FFFF
lmb_bram_if_cntlr	dlmb_cntlr	0x00000000	0x00003FFF
lmb_bram_if_cntlr	ilmb_cntlr	0x00000000	0x00003FFF
opb_uartlite	RS232	0x84010000	0x8401FFFF
opb_gpio	LEDs_4Bit	0x40000000	0x4000FFFF
mch_opb_ddr	DDR_SDRAM_32Mx32	0x30000000	0x31FFFFFF
opb_intc	opb_intc_0	0x41200000	0x4120FFFF
opb_central_dma	opb_central_dma_0	0xA0000000	0xA000FFFF

System Configuration

This system requires that the MicroBlaze processor be configured to support Xilinx CacheLink (XCL) caching. The DDR SDRAM clock is also set to run at a different frequency than the OPB bus clock.The MCH OPB DDR SDRAM is also configured to support the MicroBlaze processor MCH connections and OPB bursting. This illustrates the capacity of the MCH OPB DDR SDRAM to handle OPB bursts and cacheline transactions concurrently. The OPB Central DMA controller is configured to allow access to its registers.

The following sections explain the configuration of the MicroBlaze processor, the MCH OPB DDR SDRAM, and the OPB Central DMA controller.

Setting the MicroBlaze Processor Parameters

Under the Parameters Tab of the MicroBlaze processor, change the parameters as shown in Figure 2. These parameters set the cacheable block of main memory between `0x30000000` and `0x3000FFFF`. Both the instruction cache and the data cache are enabled. The size of the instruction cache and data cache are set to 8K respectively. The XCL interface is enabled for both the instruction cache and the data cache. This allows the MicroBlaze processor to use the dedicated XCL interface instead of the shared OPB bus for cacheline transfers.

Note: The I-cache writes have been enabled. This enables the WIC (Write Instruction Cache) instruction to write into the I-cache.

See the *MicroBlaze Processor Reference Guide* under the sections Instruction Cache and Data Cache for more details on the MicroBlaze processor caches.
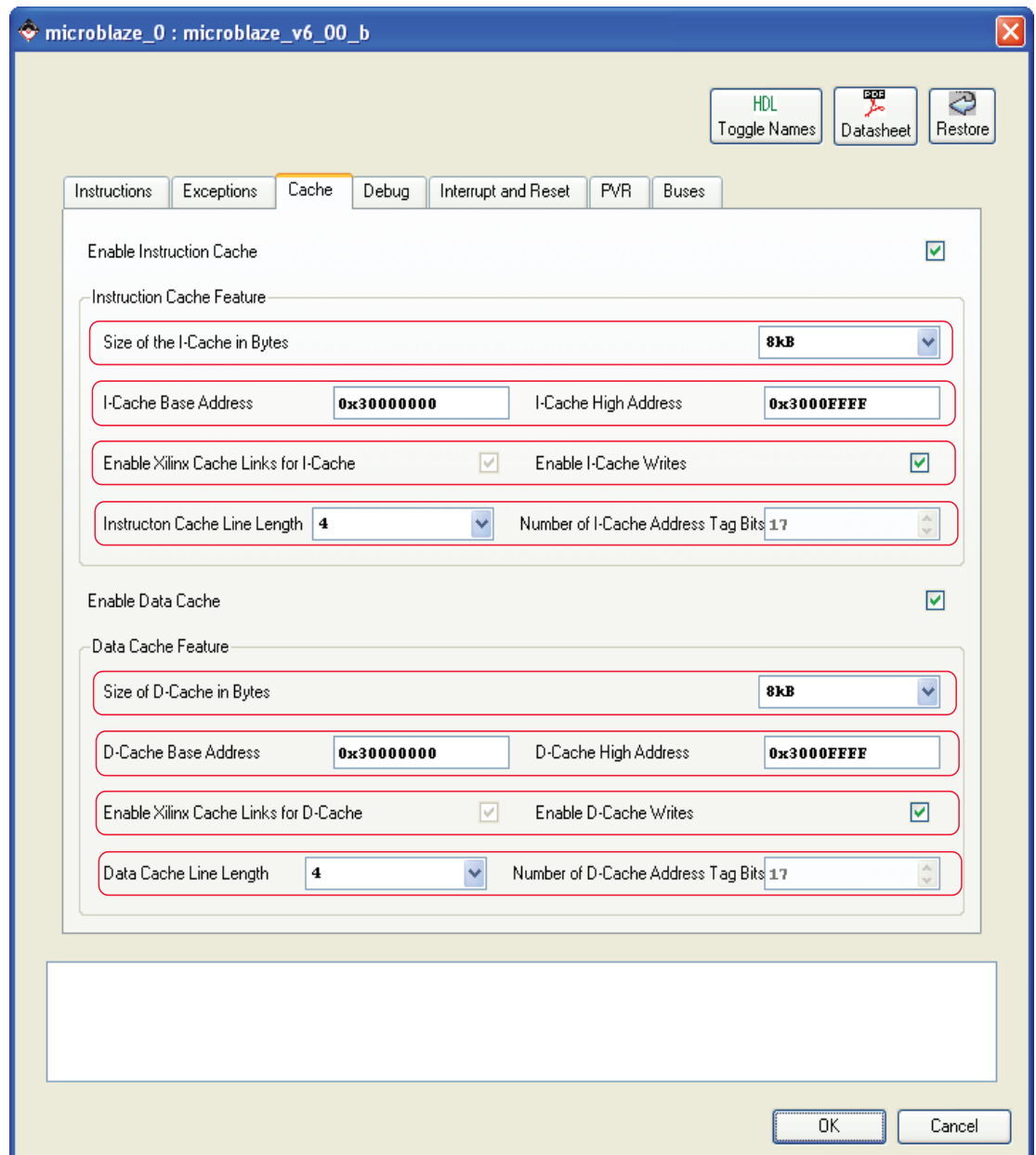

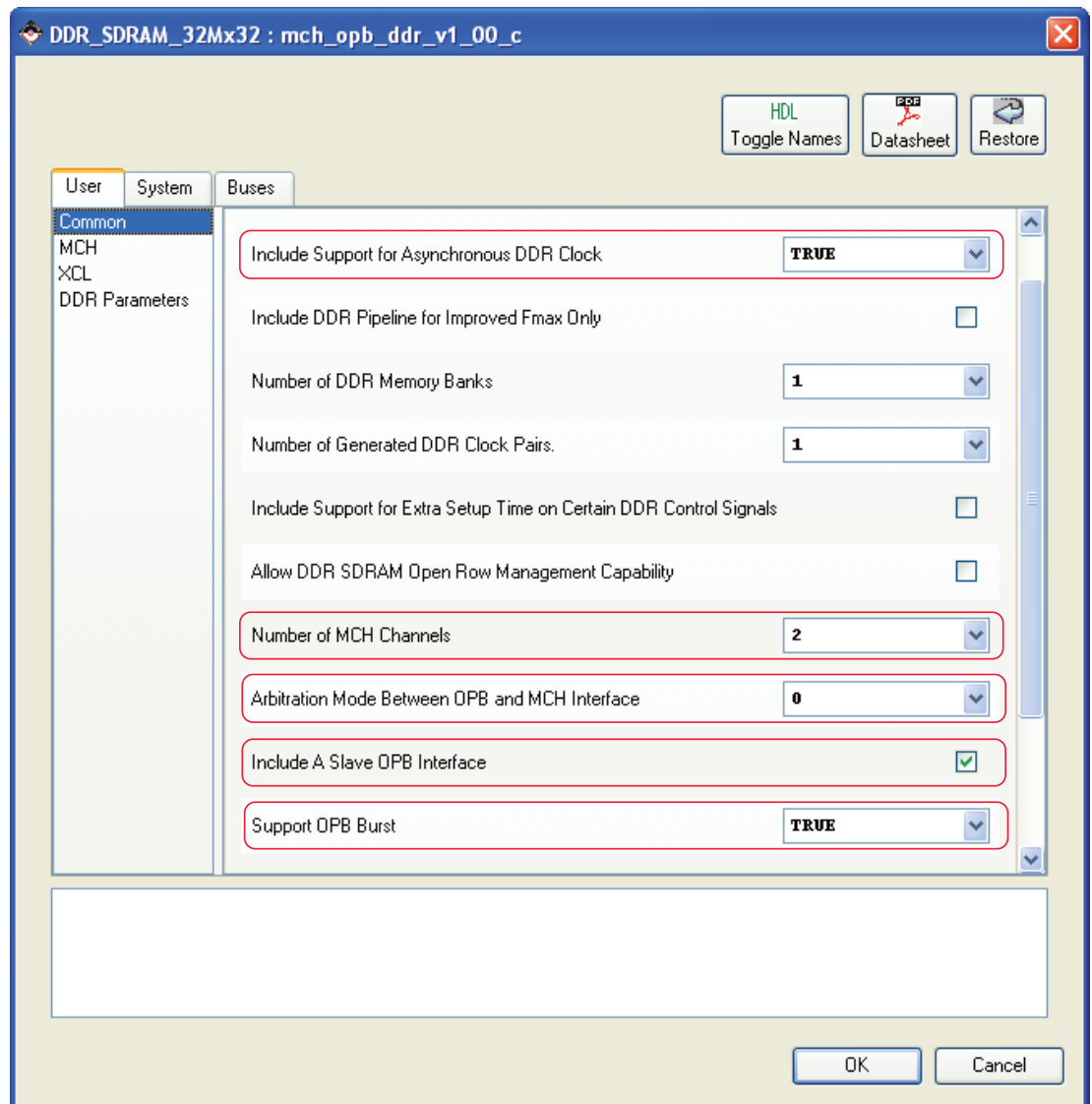*Figure 2:* **Setting MicroBlaze Cache Parameters**

Setting the MCH OPB DDR SDRAM Parameters

The MCH OPB DDR SDRAM needs to be configured to support two MCH channels which connect to the MicroBlaze processor instruction and data caches. The XCL properties of these channels also need to be set appropriately. In addition, the MCH OPB DDR SDRAM needs to be configured to support OPB bursting to handle the transactions from the OPB Central DMA controller effectively.

The MCH OPB DDR SDRAM core supports the capability to seperate the DDR clock from the OPB bus clock. This capability is enabled by setting the parameter *Include Support for Asynchronous DDR Clock* to TRUE.

The MCH OPB DDR SDRAM is configured for two channels by configuring the parameter *Number of MCH Channels* to 2. The MCH OPB DDR SDRAM is configured to support burst transactions by setting the parameters *Include A Slave OPB Interface* to 1 and *Support OPB Burst* to TRUE in the User tab under the MCH OPB DDR SDRAM core as shown in Figure 3.

XAPP912_03_051807

Figure 3: **Setting MCH OPB DDR SDRAM Parameters**

www.BDTIC.com/XILINX

Setting MCH Properties for the MCH OPB DDR SDRAM

The MCH interface properties need to be set for the MCH OPB DDR SDRAM. The MCH OPB DDR SDRAM currently supports only the XCL protocol, therefore the parameter *Interface Protocol* for the channels is set to 0 which indicates the XCL protocol. The *Depth of Access Buffer* parameter is set to the default value 16 for all the channels. The parameter *Depth of Readdata Buffer* for the channels that connect the I-cache and D-cache is set to 0, because the MicroBlaze processor can consume the data as soon as its available. Setting this parameter eliminates the read data buffer and the latency that normally exists while reading the data from this buffer.

These parameters are set in the MCH properties section in the User tab of the MCH OPB DDR SDRAM core as shown in Figure 4.
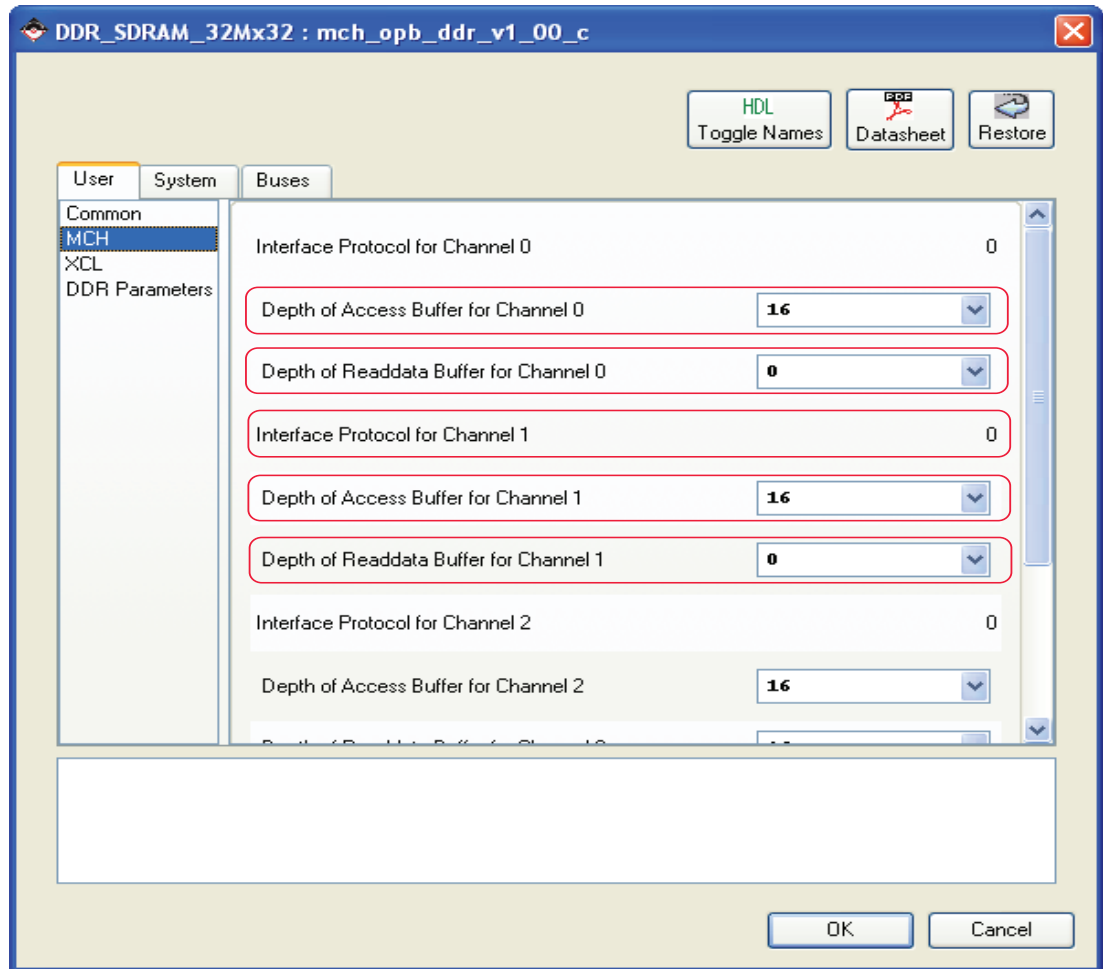

XAPP912_04_051807

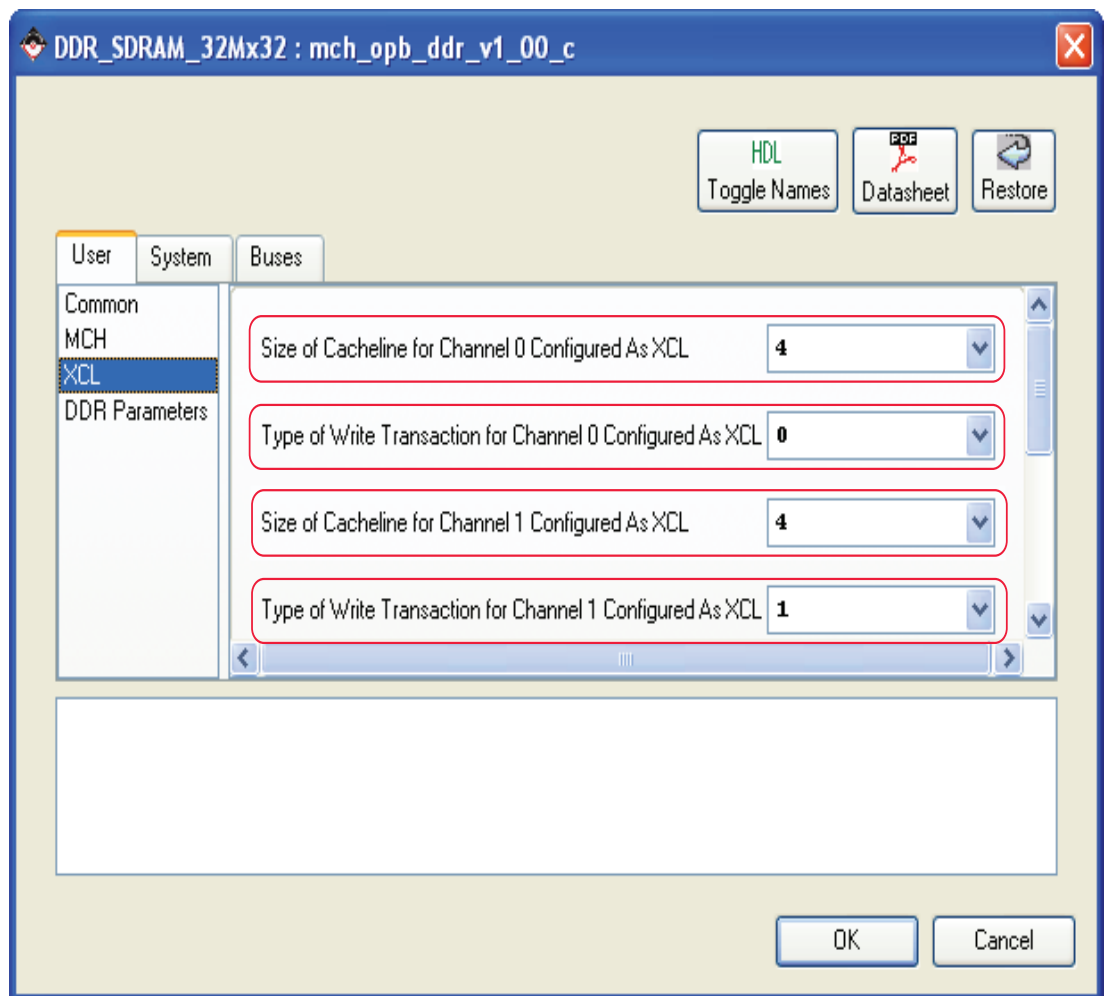*Figure 4:* **Setting MCH Properties for MCH OPB DDR SDRAM**

Setting XCL Properties for the MCH OPB DDR SDRAM

The size of the cacheline, in number of 32-bit words, is set for all the channels that are configured as XCL channels. The MCH Channel 0 is connected to the I-cache of the MicroBlaze processor. The MCH Channel 1 is connected to the D-cache of the MicroBlaze processor. Since the cacheline size of the I-cache and D-cache for MicroBlaze processor is four words, the cachelines sizes for channels 0 and 1 are set to four words.

The I-cache of MicroBlaze processor will only do read accesses to memory, therefore the parameter *Type of Write Transaction for Channel 0 Configured as XCL* is set to 0. This indicates that this channel will not perform any memory write transfers and reduces the logic implemented for this channel. In Figure 2, I-cache writes are enabled which enables the WIC instruction to write into the I-cache. Enabling the WIC instruction does not affect the setting of the *Type of Write Transaction for Channel 0 Configured as XCL* parameter of MCH channel 0.

The D-cache of the MicroBlaze processor will perform only single beat writes to memory, therefore the *Type of Write Transaction for Channel 0 Configured as XCL* is set to 1. This indicates that only single-beat writes are performed on this channel.

These parameters are set in the XCL properties section under the User tab of the MCH OPB DDR SDRAM core as shown in Figure 5.

XAPP912_05_051807

Figure 5: **Setting XCL Properties for MCH OPB DDR SDRAM**

Connecting the MCH OPB DDR SDRAM to the MicroBlaze Processor

To allow the MicroBlaze processor to cache over XCL, connections must be made between the MicroBlaze processor and the MCH OPB DDR SDRAM. These connections can be viewed in the `system.mhs` file.

The MicroBlaze processor uses the following interface connections:

- ♦ BUS_INTERFACE IXCL = ixcl
- ♦ BUS_INTERFACE DXCL = dxcl

A portion of the MHS file showing the bus interface connections of the MicroBlaze processor is shown in Figure 6.

```
BEGIN microblaze
  PARAMETER INSTANCE = microblaze_0
  PARAMETER HW_VER = 6.00.b
  PARAMETER C_DEBUG_ENABLED = 1
  PARAMETER C_NUMBER_OF_PC_BRK = 1
  PARAMETER C_NUMBER_OF_RD_ADDR_BRK = 0
  PARAMETER C_NUMBER_OF_WR_ADDR_BRK = 0
  PARAMETER C_AREA_OPTIMIZED = 1
  PARAMETER C_USE_ICACHE = 1
  PARAMETER C_USE_DCACHE = 1
  PARAMETER C_ICACHE_USE_FSL = 1
  PARAMETER C_DCACHE_USE_FSL = 1
  PARAMETER C_CACHE_BYTE_SIZE = 8192
  PARAMETER C_DCACHE_BYTE_SIZE = 8192
  PARAMETER C_ICACHE_BASEADDR = 0x30000000
  PARAMETER C_ICACHE_HIGHADDR = 0x3000FFFF
  PARAMETER C_DCACHE_BASEADDR = 0x30000000
  PARAMETER C_DCACHE_HIGHADDR = 0x3000FFFF
  BUS_INTERFACE DLMB = dlmb
  BUS_INTERFACE ILMB = ilmb
  BUS_INTERFACE DOPB = mb_opb
  BUS_INTERFACE IOPB = mb_opb
# MicroBlaze caches connect to MCH interfaces on the MCH OPB DDR
  BUS_INTERFACE DXCL = dxcl
  BUS_INTERFACE IXCL = ixcl
  PORT CLK = sys_clk_s
  PORT RESET = microblaze_rst
  PORT DBG_CAPTURE = DBG_CAPTURE_s
  PORT DBG_CLK = DBG_CLK_s
  PORT DBG_REG_EN = DBG_REG_EN_s
  PORT DBG_TDI = DBG_TDI_s
  PORT DBG_TDO = DBG_TDO_s
  PORT DBG_UPDATE = DBG_UPDATE_s
  PORT Interrupt = Interrupt
END
```

XAPP912_06_051807

Figure 6: **MicroBlaze XCL Connections in the MHS File**

The MCH OPB DDR SDRAM uses the following interface connections.

- ♦ BUS_INTERFACE MCH0 = ixcl
- ♦ BUS_INTERFACE MCH1 = dxcl

This connects the MicroBlaze I-cache to MCH channel 0 and the MicroBlaze D-cache to channel 1 of the MCH OPB DDR SDRAM.

A portion of the MHS file with the bus interface connections of the MCH OPB DDR SDRAM core is shown in Figure 7.

```
BEGIN mch_opb_ddr
 PARAMETER INSTANCE = DDR_SDRAM_32Mx32
 PARAMETER HW_VER = 1.00.c
 PARAMETER C_NUM_CLK_PAIRS = 1
 PARAMETER C_DDR_ASYNC_SUPPORT = 1
 PARAMETER C_INCLUDE_DDR_PIPE = 0
 PARAMETER C_NUM_CHANNELS = 2
 PARAMETER C_INCLUDE_OPB_IPIF = 1
 PARAMETER C_INCLUDE_OPB_BURST_SUPPORT = 1
 PARAMETER C_MCH_OPB_CLK_PERIOD_PS = 20000
 PARAMETER C_DDR_TMRD = 20000
 PARAMETER C_DDR_TWR = 20000
 PARAMETER C_DDR_TRAS = 60000
 PARAMETER C_DDR_TRC = 90000
 PARAMETER C_DDR_TRFC = 80000
 PARAMETER C_DDR_TRCD = 30000
 PARAMETER C_DDR_TRRD = 15000
 PARAMETER C_DDR_TRP = 30000
 PARAMETER C_DDR_TREFI = 7800000
 PARAMETER C_DDR_DWIDTH = 32
 PARAMETER C_DDR_AWIDTH = 13
 PARAMETER C_DDR_COL_AWIDTH = 9
 PARAMETER C_DDR_BANK_AWIDTH = 2
 PARAMETER C_MCH0_ACCESSBUF_DEPTH = 16
 PARAMETER C_MCH0_RDDATABUF_DEPTH = 0
 PARAMETER C_XCL0_LINESIZE = 4
 PARAMETER C_XCL0_WRITEXFER = 0
 PARAMETER C_MCH1_ACCESSBUF_DEPTH = 16
 PARAMETER C_MCH1_RDDATABUF_DEPTH = 4
 PARAMETER C_XCL1_LINESIZE = 4
 PARAMETER C_XCL1_WRITEXFER = 1
 PARAMETER C_MEM0_BASEADDR = 0x30000000
 PARAMETER C_MEM0_HIGHADDR = 0x31ffffff
 # MCH interfaces connect to MicroBlaze caches
 BUS_INTERFACE MCH0 = ixcl
 BUS_INTERFACE MCH1 = dxcl
 BUS_INTERFACE SOPB = mb_opb
 PORT MCH_OPB_Clk = sys_clk_s
 PORT MCH_OPB_Rst = periph_rst_7
 PORT Device_Clk = dcm_0_FB
 PORT Device_Clk_n = dcm_0_FB_n
 PORT Device_Clk90_in = clk_90_s
 PORT Device_Clk90_in_n = clk_90_n_s
 PORT DDR_Clk90_in = ddr_clk_90_s
```

XAPP912_07_051807

Figure 7: **MCH OPB DDR SDRAM Interface Connections in the MHS File**

Setting up the MCH OPB DDR SDRAM Clocking Structure

This reference system is set up so that the DDR clock runs at 100 MHz and the OPB clock runs at 50 MHz. The SP305 board provides a 100 MHz clock. The 100 MHz clock is fed into a DCM, which creates the 50 MHz clock. The 50 MHz clock, which is the system clock, is fed into the processor and the other cores in the system. The 50 MHz clock is then fed into the MCH_OPB_Clk port of the MCH OPB DDR SDRAM and the 100 MHz clock is fed into the Device_Clk port of the MCH OPB DDR SDRAM. A second DCM is used to generate the phase shifted DDR feedback clock. The DDR clocking structure is shown in Figure 8.
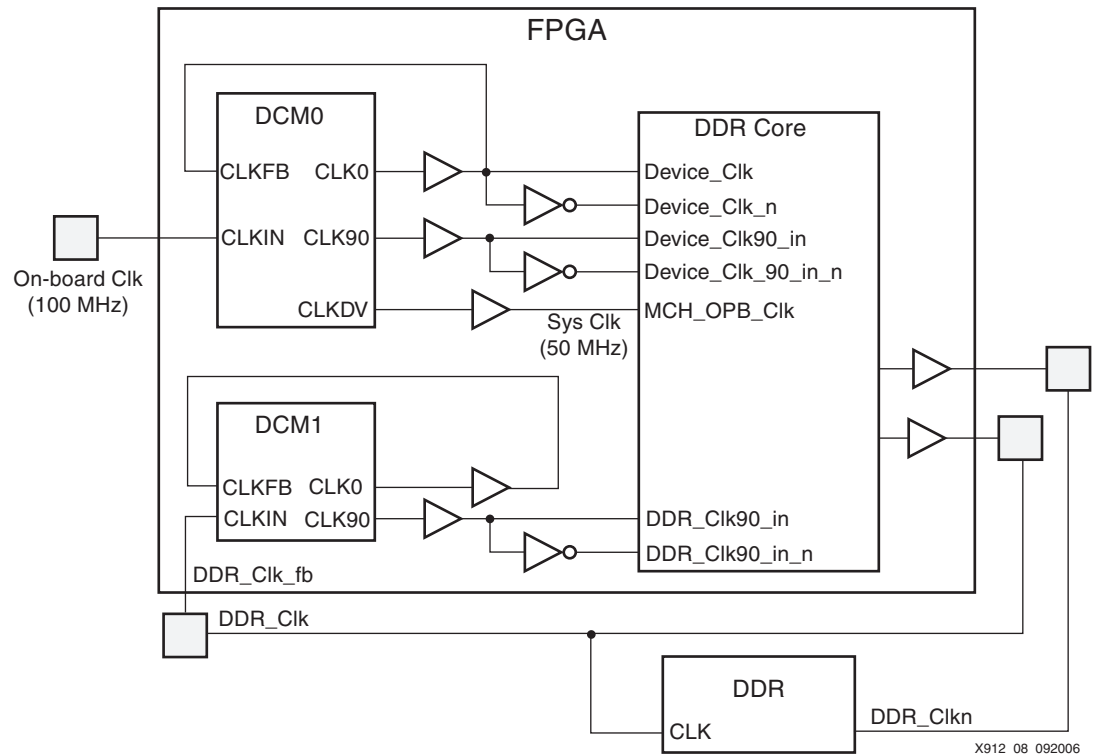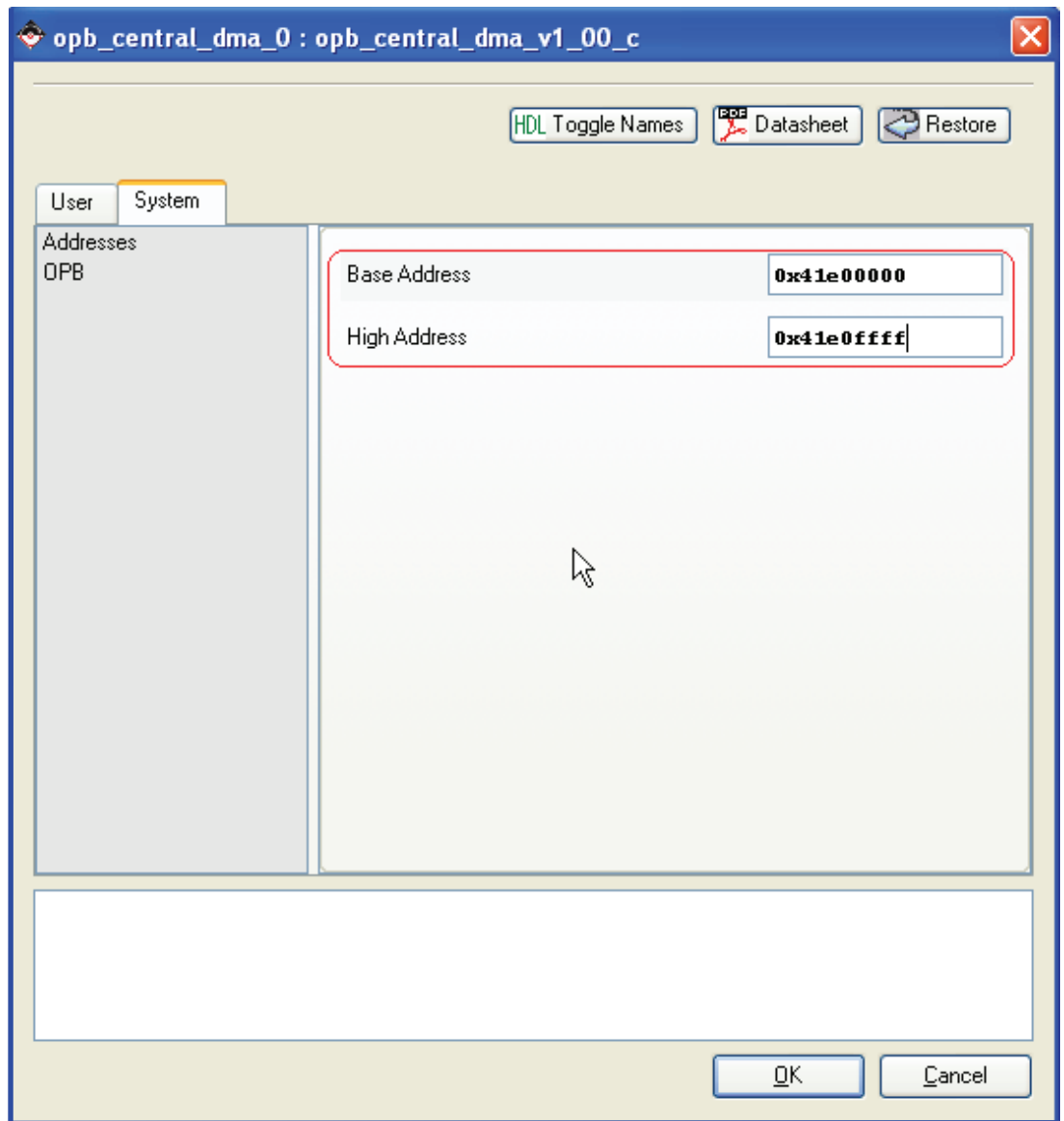

Figure 8: **DDR Clocking Structure**

www.BDTIC.com/XILINX

Setting System Parameters for the OPB Central DMA

The addresses of the OPB Central DMA controller must be set to allow the MicroBlaze processor to access its registers. In the System tab of the OPB Central DMA core, the Base Address is set as $0x41E00000$ and the High Address is set to $0x41E0FFFF$ as shown in Figure 9.
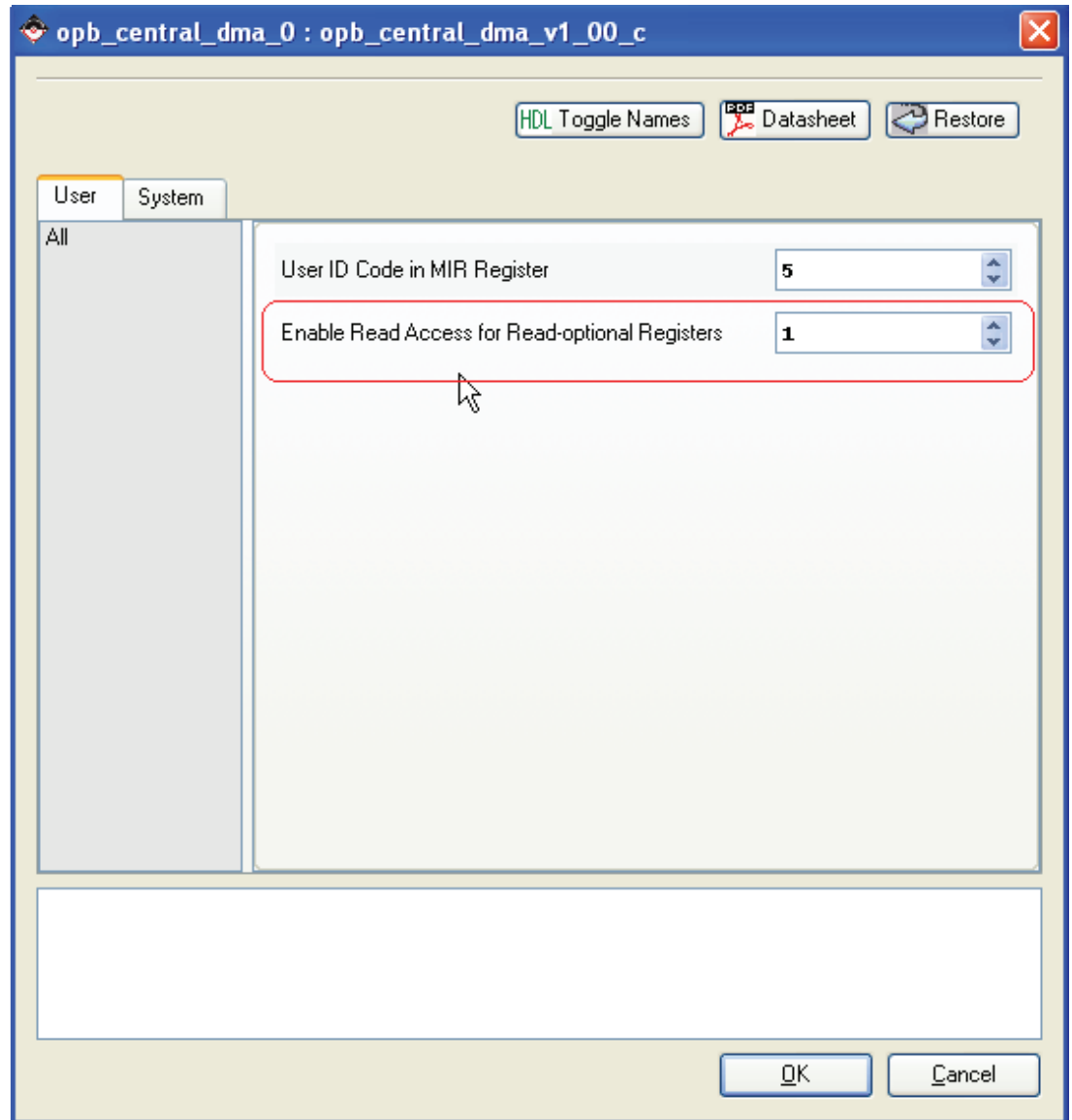

Figure 9: **Setting System Parameters for OPB Central DMA**

Setting User Parameters for the OPB Central DMA

In the User tab of the OPB Central DMA core, set the parameter *Enable Read Access for Read-optional Registers* to 1 as shown in Figure 10. This enables the software application to read the OPB Central DMA registers to check if the DMA interrupt is properly cleared after a DMA operation has completed.

X912_10_092006

Figure 10: **Setting User Parameters for OPB Central DMA**

Connecting the OPB Central DMA Controller to the System

Connect the **opb_central_dma_0 sopb** as slave and the **opb_central_dma_0 mopb** as master to the OPB Bus connection by marking the connection to the OPB Bus in the Bus Interface filter of the reference system.

The Bus Interface connections of the OPB Central DMA to the OPB Bus are shown in Figure 11.
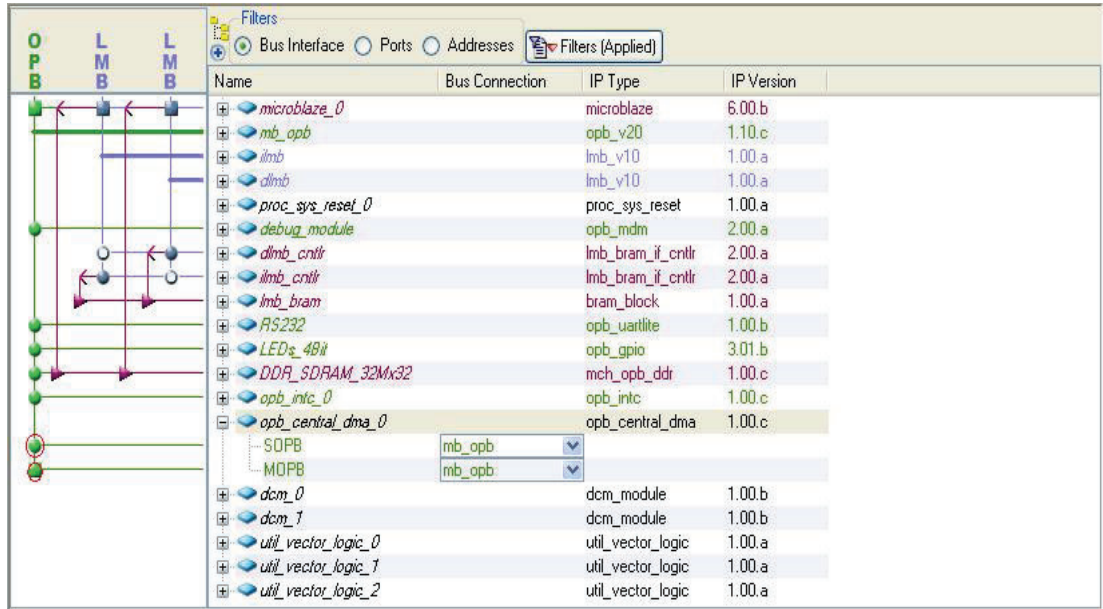

Figure 11: **Bus Interface Filter, Setting Master and Slave for OPB Central DMA**

The DMA interrupt signal is connected to the OPB Interrupt Controller (OPB INTC) in the reference system. Choose the **Ports Filter** and expand the opb_central_dma_0 tree node. Connect the output port **DMA_Interrupt** to **DMA_Irpt**. Obtain the list of interrupts by expanding the opb_intc_0 tree node and clicking on the last port under the **Net** for **Intr**. This will bring up the **Interrupt Connection Dialog** box. Add the **DMA_Irpt** output to Connected Interrupts as shown in Figure 12.
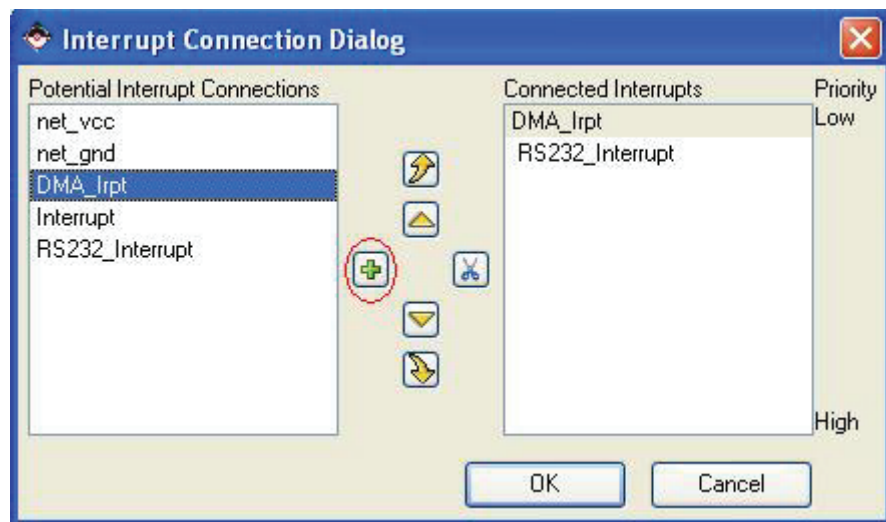

Figure 12: **Interrupt Connect Dialog Box, Adding OPB Central DMA Interrupt**

www.BDTIC.com/XILINX

The Software Application

The software application, which is executed from the cacheable block of main memory, tests DMA operations out of the non-cacheable block of main memory.

At the start of the application, the memory block at the DMA source address and the DMA destination address are cleared. The data is written to the memory block at the source address. OPB Central DMA is initialized and set up to use interrupts. DMA operations start when the source base address and destination base address are written to the appropriate OPB Central DMA register. An interrupt occurs when the DMA transfer is complete.

When DMA operations are complete, the data at the source address are compared with the data at the destination address to ensure the correct data transfer. Also, the software application clears the interrupt generated by the DMA transfer.
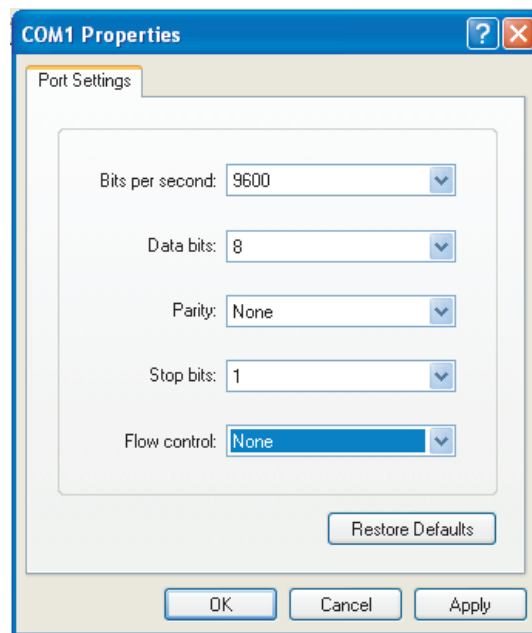
The software application described above is run out of the on-board DDR memory.

In the software application, set all the linker script options to main memory and do not initialize the block RAMs. Add the source for the software application to the project and copy the source to the new reference system directory. The software application is found under the project root directory `Test_App/TestApp_MemoryCaching.c`.

Executing the Reference System

To execute the reference system, the bitstream needs to be generated and the software application needs to be compiled. The bitstream and the compiled software application for this system are available in `ready_for_download/` under the project root directory.

A HyperTerminal or similar program needs to be connected to the COM port and the board's UART needs to be connected to the COM port. Set the HyperTerminal to the Baud Rate of **9600**, Data Bits to **8**, Parity to **None** and Flow Control to **None**. See Figure 13 for the settings.

x912_13_092006

Figure 13: **HyperTerminal Settings**

Executing the Reference System using the Pre-Built Bitstream and the Compiled Software Applications

To execute the system using files inside the `ready_for_download/` in the project root directory, follow these steps:

1. Change directories to the `ready_for_download` directory.
2. Use iMPACT to download the bitstream by using the following:

    `impact -batch xapp912.cmd`

3. Invoke XMD and connect to the MicroBlaze processor by the following command:

    `xmd -opt xapp912.opt`

4. Download the executables by the following command:

    `dow executable.elf`

Executing the Reference System from EDK

To execute the system using EDK, follow these steps:

1. Open `system.xmp` inside EDK.
2. Use **Hardware→Generate Bitstream** to generate a bitstream for the system.
3. Use **Software→Build All User Applications** to build the software applications.
4. Download the bitstream to the board with **Device Configuration→Download Bitstream**.
5. Launch XMD with **Debug→Launch XMD...**
6. Download the executables by the following command:

    `dow executable.elf`

Running the Software Applications

To run the either of software applications, use the *`run`* command inside XMD. The status of the software application is displayed in the HyperTerminal data screen.

Running the OPB Central DMA Software Application

After downloading the bitstream, download the software application `executable.elf` to main memory using XMD. After downloading the software application, the program must be executed. The status of the software application is displayed to the HyperTerminal. Once the DMA operations are complete and verified, the LEDs blink several times and the output reads as follows:

```
-- Entering main() --

Starting Writing and Clearing Source and Destination Address.
Finished Writing and Clearing Source and Destination Address.
Starting DMA Transfer
Waiting:.
DMA Transfer Complete, Verifying Destination Data
Destination Data is Correct
DMA Interrupt Cleared
Congratulations! DMA Operations Completed Successfully!

-- Exiting main() --
```

Conclusion

This application note describes how to take advantage of the caching features in the MCH OPB DDR SDRAM memory controller. The reference system (built for the Xilinx SP305 Rev B board)

includes a stand-alone software application that runs inside the cacheable block of main memory. This application tests DMA operation inside the non-cacheable block.

References

UG081, *MicroBlaze Processor Reference Design*

DS496, *MCH OPB Synchronous Double Data Rate(DDR) SDRAM Controller (v1.00a)* Product Specification

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
10/20/05	1.0	Initial Xilinx release.
10/24/06	1.1	Updated for EDK 8.2.01i.
3/2/07	1.2	Updated for EDK 8.2.02i; corrected .../xapp912.zip Hyperlink on pg1.
6/1/07	1.3	Updated for EDK 9.1.01i.