LogiCORE IP XPS Multi-channel External Memory Controller (XPS MCH EMC) (3.01a)

DS575 June 22, 2011 **Product Specification**

Introduction

The Xilinx LogiCORE™ Multi-channel External Memory Controller (XPS MCH EMC) provides the control interface for external synchronous, asynchronous SRAM, and Flash memory devices through the MCH or PLB interfaces. It is assumed that the reader is familiar with the PLB and MCH protocol.

Features

The XPS MCH EMC is a soft IP core designed for Xilinx FPGAs and contains the following features:

- Connects as a 32-bit or 64-bit slave on PLB v4.6 bus of 32, 64, or 128 bits
- Parameterizable number (0 to 4) of channel interfaces that can be configured with a Xilinx CacheLink (XCL) protocol (see "Reference Documents")
- Can be used with PLB interface only or MCH interface only or in combination of both PLB and MCH interfaces
- Supports multiple (up to 4) external memory banks
- Supports single-beat and burst transactions
- Supports both linear and target word cache line transaction of 1, 4, 8 and 16. See Table 4 for more details
- Supports low latency PLB Point-to-Point topology
- Supports Synchronous / Asynchronous SRAMs and Nor Flash memory devices
- Supports page mode Nor flash
- Supports target-word first PLB Cacheline read and line-word first PLB Cacheline write transactions of 4, 8 and 16 words
- Supports memory data widths of 64-bit, 32-bit, 16-bit, and 8-bit
- Supports data width matching
- Supports configurable cycle time for read and write operations

LogiCORE Facts	
Core Specifics	
Supported Device Family (1)	Virtex-6, Virtex-5/5TX/5FX, Virtex-4/4Q/4QR, Spartan-6, Spartan-3, Spartan-3A, Spartan-3E, Spartan-3A DSP, Automotive Spartan-3/3E/3A/3A DSP
Supported User Interfaces	MCH and PLB
Resources Used	
See Table 16, Table 17, Table 18, Table 19 and Table 20	
Provided with Core	
Documentation	Product Specification
Design File Formats	VHDL
Constraints File	N/A
Verification	N/A
Instantiation Template	N/A
Reference Designs & Application Notes	N/A
Design Tool Requirements (2)	
Xilinx Implementation Tools	XPS 13.2
Verification	N/A
Simulation	ModelSim PE/SE
Synthesis	ISE 13.2
Support	
Provided by Xilinx, Inc.	

Notes:
1. For a complete listing of supported devices, see the release notes for this core.
2. For a listing of the supported tool versions, see the ISE Design Suite 13: Release Note Guide.

Functional Description

The top-level block diagram of the XPC MCH EMC core is shown in Figure 1. The core consists of the MCH PLB Interface Module, Select Parameters module, Mem State Machine module, Mem Steer module, Address Counter Mux module, Counters module, and the I/O Registers module.

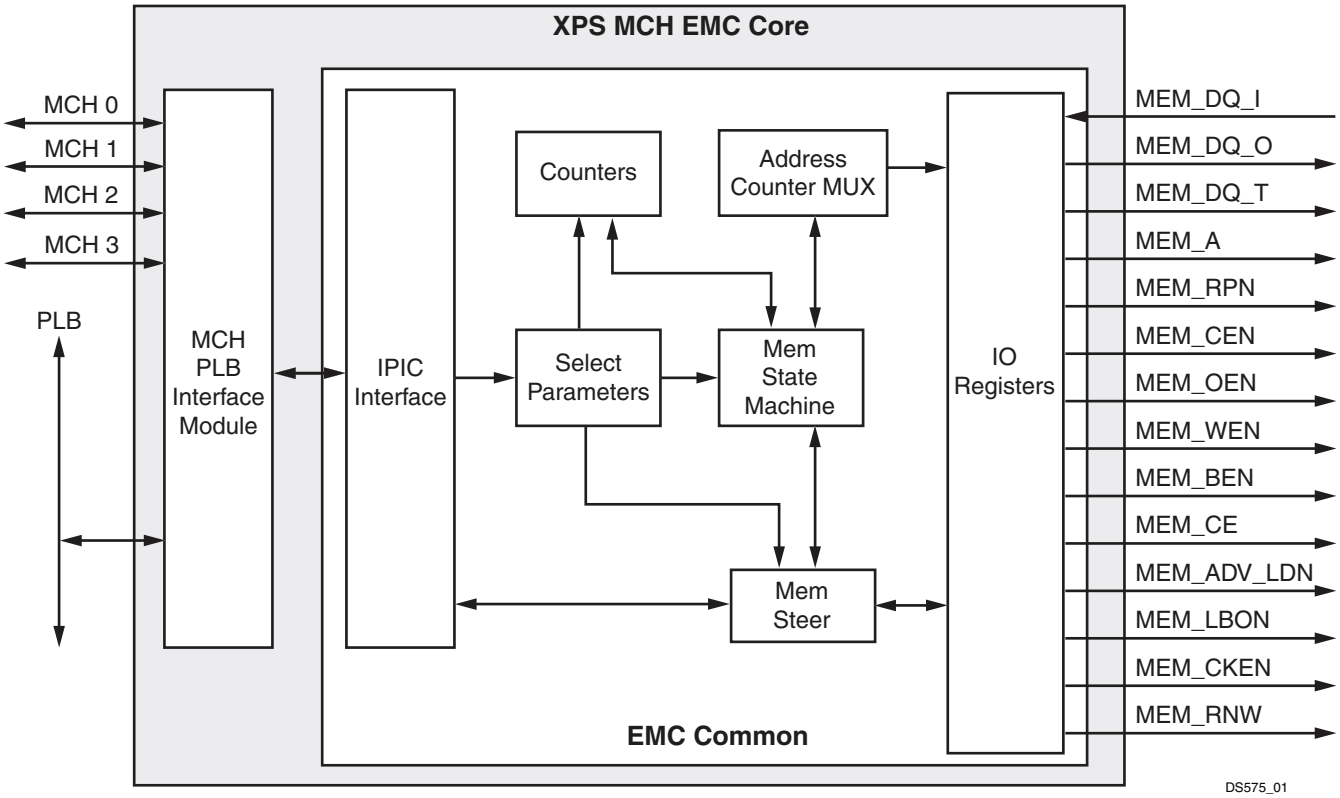

Figure 1: **Top Level Block Diagram of the XPS MCH EMC Core**

MCH PLB Interface Module

The MCH PLB Interface Module provides an optional interface to the MCH interface and PLB. The MCH PLB Interface Module consists of an optional PLB slave interface, a parameterizable number of channel interfaces and arbitration logic to select between the MCH and the PLB interfaces. The appropriate multiplexers/de-multiplexers connect the selected channel or PLB transaction to the EMC core. This module does the necessary protocol and timing translation between the MCH and IPIC interfaces or between the PLB and IPIC interfaces.

Select Parameters Module

The Select Parameters module provides necessary pipeline delays or timing delays based on the type of memory. It also indicates the type of memory connected to the core.

Mem State Machine Module

The state diagram of Mem State Machine is shown in Figure 2.

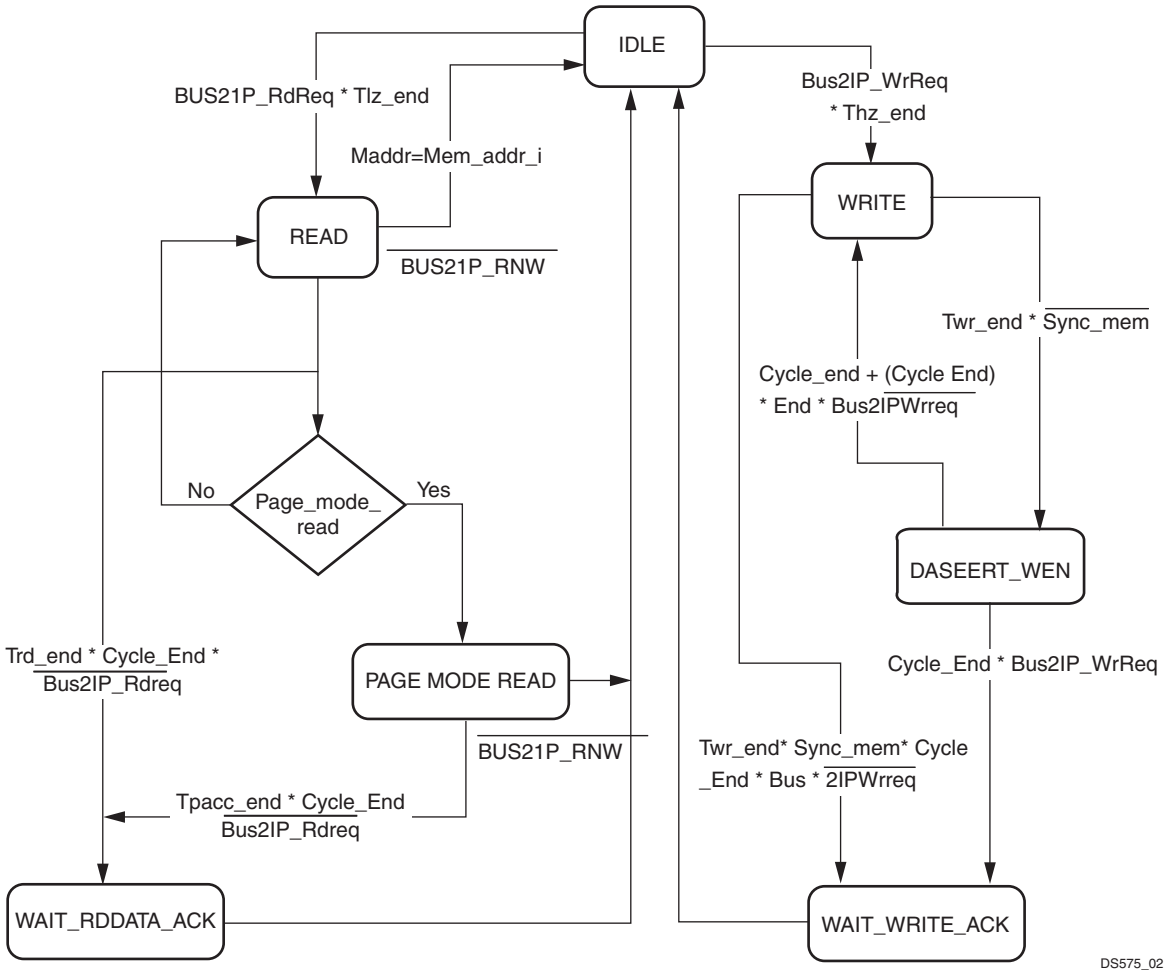

Figure 2: **XPS MCH EMC Mem State Machine**

The Mem State Machine controls read and write transactions for the memory. It handles all single, burst and page mode read (Nor flash) conditions and provides necessary control signals to the Counters, Mem Steer and Address Counter Mux modules.

Mem Steer Module

The Mem Steer module contains the logic to provide the steering of read data, write data and memory control signals. It generates the acknowledge signals for the MCH PLB Interface Module. This module contains data width matching logic.

Address Counter Mux Module

The Address Counter Mux module provides the address count to the Mem Steer module and provides the address suffix to generate the memory address. It also handles the cycle end logic which is directed to the Mem State Machine.

Counters Module

This module contains the counters for counting the read cycle time and write cycle time.

Mem State Machine Module

The state diagram of Mem State Machine is shown in Figure 2

The Mem State Machine controls read and write transactions for the memory., handles all single, burst and page mode read (flash) conditions, and provides necessary the control signals to the Counters, Mem Steer, and Address Counter Mux modules.

I/O Registers Module

Registers are used on all signals to and from the memory bank to provide consistent timing on the memory interface. The I/O Registers module present in the design depends upon the setting of the C_INCLUDE_NEGEDGE_IOREGS. All signals output to the memory bank are registered on the rising edge of the system clock. If C_INCLUDE_NEGEDGE_IOREGS = 1, the signals are registered again on the falling edge of the system clock, as shown in Figure 3, and can be used at lower clock frequency to provide adequate setup and hold times to synchronous memories.

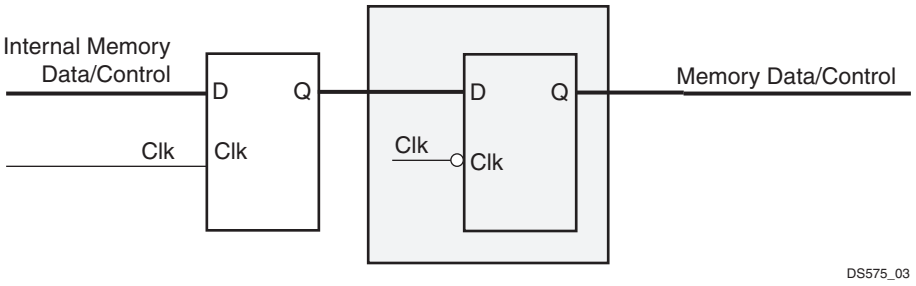

Figure 3: **XPS MCH EMC Output Registers When C_INCLUDE_NEGEDGE_IOREGS = 1**

Design Parameters

To allow the user to create a XPS MCH EMC that is uniquely tailored for the user's system, certain features are parameterizable in the XPS MCH EMC design. This allows the user to have a design that utilizes only the resources required by the system and runs at the best possible performance. The features that are parameterizable in the XPS MCH EMC a re as shown in Table 1.

Table 1: **Design Parameters**

Generic	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type
XPS MCH EMC Parameters					
G1	Target FPGA family	C_FAMILY	virtex, virtex5, virtex4, spartan6, spartan3, spartan3A, spartan3e, spartan3adsp, aspartan3, aspartan3e, aspartan3a, aspartan3adsp	virtex5	string
G2	Number of memory banks	C_NUM_BANKS_MEM	1 - 4	1	integer
G3	Arbitration mode between MCH and PLB	C_PRIORITY_MODE	0 = Fixed priority mode	0	integer
G4	Include PLB slave interface	C_INCLUDE_PLB_IPIF(1)	0 = Don't include PLB interface 1 = Include PLB interface	1	integer
G5	Includes support for write buffer	C_INCLUDE_WRBUF	0 = Don't include the write buffer 1 = Include the write buffer	1	integer
G6	PLB master ID bus width	C_SPLB_MID_WIDTH	$\log_2$(C_SPLB_NUM_MASTERS) with a minimum value of 1	1	integer
G7	Number of PLB masters	C_SPLB_NUM_MASTERS	1 - 16	1	integer
G8	Selects point-to-point or shared bus topology	C_SPLB_P2P	0 = Shared bus topology 1 = Point-to-Point bus topology	0	integer
G9	Data bus width for MCH and PLB	C_SPLB_DWIDTH(2)	32, 64, 128	32	integer
G10	Address bus width for MCH and PLB	C_MCH_SPLB_AWIDTH	32	32	integer
G11	Data bus width of the smallest master	C_SPLB_SMALLEST_MASTER	32, 64, 128	32	integer
G12	Slave data bus width	C_MCH_NATIVE_DWIDTH(2)(3)	32, 64	32	integer
G13	Input/output data and control signals using the falling edge of the clock	C_INCLUDE_NEGEDGE_IOREGS(4)(7)	0 = Don't include negative edge IO registers (data and control signals are input/output on the rising edge of the clock) 1 = Include negative edge IO registers (data and control signals are input/output on the falling edge of the clock)	0	integer

Table 1: **Design Parameters *(Cont'd)***

Generic	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type
G14	Flash type of memory bank x	C_PAGEMODE_FLASH_x(5)(12)	0= Flash type is not page mode 1= Flash type is page mode	0	integer
G15	Width of memory bank x data bus	C_MEMx_WIDTH(5)	8, 16, 32, 64	32	integer
G16	Execute multiple memory access cycles to match memory bank x data width to MCH and PLB data width	C_INCLUDE_DATAWIDTH_MATCHING_x(5)(9)	0 = Don't include data width matching 1 = Include data width matching	0	integer
G17	Memory type of memory bank x	C_SYNCH_MEM_x(5)	0 = Memory type is asynchronous 1 = Memory type is synchronous	0	integer
G18	Pipeline delay (in clock cycles) of memory bank x	C_SYNCH_PIPEDELAY_x(5)(8)	1 = Flow-Through model 2 = Pipeline Model	2	integer
Memory Bank Timing Parameters					
G19	Read cycle chip enable low to data valid duration of memory bank x	C_TCEDV_PS_MEM_x(5)(10)(11)	Integer number of picoseconds	15000(12)	integer
G20	Read cycle address valid to data valid duration of memory bank x	C_TAVDV_PS_MEM_x(5)(10)(13)	Integer number of picoseconds	15000(12)	integer
G21	Read cycle chip enable high to data bus high impedance duration of memory bank x	C_THZCE_PS_MEM_x(5)(14)(15)	Integer number of picoseconds	7000(12)	integer
G22	Read cycle output enable high to data bus high impedance duration of memory bank x	C_THZOE_PS_MEM_x(5)(14)(16)	Integer number of picoseconds	7000(12)	integer
G23	Page access time of memory bank x in page mode flash mode	C_TPACC_PS_FLASH_x(5)(17)	Integer number of picoseconds	25000(12)(17)	integer
G24	Write cycle time of memory bank x	C_TWC_PS_MEM_x(5)(18)(19)	Integer number of picoseconds	15000(12)	integer
G25	Write enable minimum pulse width duration of memory bank x	C_TWP_PS_MEM_x(5)(18)(20)	Integer number of picoseconds	12000(12)	integer
G26	Write cycle write enable high to data bus low impedance duration of memory bank x	C_TLZWE_PS_MEM_x(5)(21)(22)	Integer number of picoseconds	0(12)	integer
Auto Calculated Parameter(23)					
G27	Maximum data width of the memory devices in all banks	C_MAX_MEM_WIDTH(5)	8, 16, 32, 64	32	integer
Address Space Parameters					
G28	Base address of memory bank x	C_MEMx_BASEADDR(5)	Valid address range(24)	None(24)	std_logic_vector

Table 1: **Design Parameters *(Cont'd)***

Generic	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type
G29	High address of memory bank x	C_MEMx_HIGHADDR(5)	Valid address range(24)	None(24)	std_logic_vector
Clock Period Parameter					
G30	MCH/PLB clock period	C_MCH_SPLB_CLK_PERIOD_PS(25)	Integer number of picoseconds	10000	integer
MCH Interface Parameters					
G31	Number of MCH channels	C_NUM_CHANNELS(26)	0 - 4	2	integer
G32	Interface protocol for channel x (x = 0 to 3)	C_MCHx_PROTOCOL(27)	0 = XCL protocol	0	integer
G33	Depth of the access buffer for channel x (x = 0 to 3)	C_MCHx_ACCESSBU_DEPTH(28)	4, 8, 16	16	integer
G34	Depth of the read data buffer for channel x (x = 0 to 3)	C_MCHx_RDDATABU_DEPTH(29)	0, 4, 8, 16	16	integer
XCL Channel Parameters					
G35	Size of the cacheline in number of 32-bit words for each channel x configured as an XCL channel	C_XCLx_LINESIZE(28)	1, 4, 8, 16	4	integer
G36	Type of write transactions for each channel x configured as an XCL channel	C_XCLx_WRITEXFER(30)	0 = No write transfers 1 = Single transfers only 2 = Cacheline transfers only	1	integer

Table 1: **Design Parameters *(Cont'd)***

Generic	Feature / Description	Parameter Name	Allowable Values	Default Value	VHDL Type

Notes:

1. When C_INCLUDE_PLB_IPIF = 0, the XPS MCH EMC will have only MCH interface, PLB transactions are not supported for this setting.
2. C_MCH_NATIVE_DWIDTH should always less than or equal to C_SPLB_DWIDTH
3. The parameter C_MCH_NATIVE_DWIDTH = 64 is valid only when C_NUM_CHANNELS = 0 & C_INCLUDE_PLB_IPIF = 1
4. This parameter should only be set to 1 under the following conditions:
   – Tfpga_output_buffer_delay + Tmemory_setup + Tboard_route_delay < Clock_period/2
   – Tmemory_output_bufferdelay + Tfpga_setup + Tboard_route_delay < Clock_period/2
   – Tfpga_output_buffer_delay is the delay between the flop output and the output pin
   – Tmemory_setup is the memory input setup time requirement
   – Tboard_route_delay is time delay between FPGA output pin to Memory input pinand vice versa
   – Tfpga_setup time is the input setup time requirement of input pins of FPGA.
5. x = values for memory banks 0 to 3.
6. C_PAGEMODE_FLASH_x can be set to 1 only when C_SYNCH_MEM_x = 0
7. C_INCLUDE_NEGEDGE_IOREGS unused when C_SYNCH_MEM_x = 0
8. C_SYNCH_PIPEDELAY_x is unused when C_SYNCH_MEM_x = 0
9. Always set this parameter to 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
10. Read cycle time is the maximum of C_TCEDV_PS_MEM_x and C_TAVDV_PS_MEM_x, and C_TCEDV_PS_MEM_x or C_TAVDV_PS_MEM_x should be >= tRC timing parameter of the given asynchronous memory.
11. Chip enable low to data valid, C_TCEDV_PS_MEM_x, is equivalent to t_{ACE} for asynchronous SRAM and t_{ELQV} for Flash in the respective memory device data sheets.
12. A value must be set for this parameter if the memory type in this bank is asynchronous. Refer to the memory device data sheet for the correct value
13. Address valid to data valid, C_TAVDV_PS_MEM_x, is equivalent to t_{AA} for asynchronous SRAM and t_{AVQV} for Flash in the respective memory device data sheets.
14. Read cycle recovery to write is the maximum of C_THZCE_PS_MEM_x and C_THZOE_PS_MEM_x.
15. Chip enable high to data bus high impedance, C_THZCE_PS_MEM_x, is equivalent to t_{HZCE} for asynchronous SRAM and t_{EHQZ} for Flash in the respective memory device data sheets.
16. Output enable high to data bus high impedance, C_THZOE_PS_MEM_x, is equivalent to t_{HZOE} for asynchronous SRAM and t_{GHQZ} for Flash in the respective memory device data sheets.
17. Page access time, C_TPACC_PS_FLASH_x is equivalent to t_{PACC} of page mode flash device data sheet and must be assigned only if C_PAGEMODE_FLASH_x is set to 1
18. Write enable low time is the maximum of C_TWC_PS_MEM_x and C_TWP_PS_MEM_x.
19. Write cycle time, C_TWC_PS_MEM_x, is equivalent to t_{WC} for asynchronous SRAM and t_{CW} for Flash in the respective memory device data sheets.
20. Write cycle minimum pulse width, C_TWP_PS_MEM_x is equivalent to t_{WP} for asynchronous SRAM and t_{PWE} for Flash in the respective memory device data sheets.
21. Write enable high to data bus low impedance, C_TLZWE_PS_MEM_x, is equivalent to t_{LZWE} for asynchronous SRAM and t_{WHGL} for Flash in the respective memory device data sheets.
22. C_TLZWE_PS_MEM_x is the parameter set to meet write recovery to read time requirements.
23. This parameter is automatically calculated when using EDK, otherwise the user must set this parameter to the maximum value of the C_MEMx_WIDTH generics.
24. No default value will be specified for C_MEMx_BASEADDR and C_MEMx_HIGHADDR to insure that the actual value is set, i.e. if the value is not set, a compiler error will be generated. The range specified by C_MEMx_BASEADDR and C_MEMx_HIGHADDR must be a power of 2.
25. Proper value must be assigned to this parameter when C_INCLUDE_PLB_IPIF=0.
26. When C_NUM_CHANNELS = 0, the XPS MCH EMC will have only PLB interface, MCH transactions are not supported for this setting.
27. Each channel can be configured to support a transfer protocol. Only the Xilinx CacheLink (XCL) protocol is supported at this time.
28. Optimal performance can be achieved with C_MCHx_ACCESSBUF_DEPTH ≥ C_XCLx_LINESIZE.
29. If the channel is connected to a master which can consume data as soon as it is available (i.e., instruction side interfaces), set the depth of the read data buffer to zero for that channel to save resources and latency.
30. If an XCL channel is connected to a master that will only perform read transfers, then the entry in C_XCLx_WRITEXFER should be set to 0 indicating that no write transfers will be performed

I/O Signals

Table 2: **I/O Signals**

Port	Signal Name	Interface	I/O	Default Value	Description
System Signals					
P1	MCH_SPLB_Clk(1)	System	I	-	MCH/PLB clock
P2	RdClk(2)	System	I	-	Read clock to capture the data from Memory
P3	MCH_SPLB_Rst(1)	System	I	-	MCH/PLB reset
PLB Interface Signals(2)					
P4	PLB_ABus[0 : 31]	PLB	I	-	PLB address bus
P5	PLB_PAValid	PLB	I	-	PLB primary address valid indicator
P6	PLB_masterID[0:C_SPLB_MID_WIDTH-1]	PLB	I	-	PLB current master identifier
P7	PLB_RNW	PLB	I	-	PLB read not write
P8	PLB_BE[0:(C_SPLB_WIDTH/8) - 1]	PLB	I	-	PLB byte enables
P9	PLB_wrBurst	PLB	I	-	PLB burst write transfer indicator
P10	PLB_rdBurst	PLB	I	-	PLB burst read transfer indicator
P11	PLB_size[0 : 3]	PLB	I	-	PLB transfer size
P12	PLB_type[0 : 2]	PLB	I	-	PLB transfer type
P13	PLB_wrDBus[0:C_SPLB_DWIDTH - 1]	PLB	I	-	PLB write data bus
P14	PLB_MSize[0 : 1]	PLB	I	-	PLB master data bus size
Unused PLB Interface Signals					
P15	PLB_UABus[0 : 31]	PLB	I	-	PLB upper address bus
P16	PLB_SAValid	PLB	I	-	PLB secondary address valid indicator
P17	PLB_rdPrim	PLB	I	-	PLB secondary to primary read request indicator
P18	PLB_wrPrim	PLB	I	-	PLB secondary to primary write request indicator
P19	PLB_abort	PLB	I	-	PLB abort bus indicator
P20	PLB_busLock	PLB	I	-	PLB bus lock
P21	PLB_TAttribute[0 : 15]	PLB	I	-	PLB transfer attribute bus
P22	PLB_lockerr	PLB	I	-	PLB lock error indicator
P23	PLB_wrPendReq	PLB	I	-	PLB pending write bus request indicator
P24	PLB_rdPendReq	PLB	I	-	PLB pending read bus request indicator
P25	PLB_rdPendPri[0 : 1]	PLB	I	-	PLB pending write request priority
P26	PLB_wrPendPri[0 : 1]	PLB	I	-	PLB pending write request priority
P27	PLB_reqPri[0 : 1]	PLB	I	-	PLB current request priority
PLB Slave Interface Signals(3)					

Table 2: **I/O Signals *(Cont'd)***

Port	Signal Name	Interface	I/O	Default Value	Description
P28	SI_addrAck	PLB	O	0	Slave address acknowledge
P29	SI_SSize[0 : 1]	PLB	O	0	Slave data bus size
P30	SI_wait	PLB	O	0	Slave wait indicator
P31	SI_rearbitrate	PLB	O	0	Slave rearbitrate bus indicator
P32	SI_wrDack	PLB	O	0	Slave write data acknowledge
P33	SI_wrComp	PLB	O	0	Slave write transfer complete indicator
P34	SI_wrBTerm	PLB	O	0	Slave terminate write burst transfer
P35	SI_rdBus[0:C_SPLB_ DWIDTH - 1]	PLB	O	0	Slave read data bus
P36	SI_rdDAck	PLB	O	0	Slave read data acknowledge
P37	SI_rdComp	PLB	O	0	Slave read transfer complete indicator
P38	SI_rdBTerm	PLB	O	0	Slave terminate read burst transfer
P39	SI_rdWdAddr[0 : 3]	PLB	O	0	Slave read word address
P40	SI_MBusy[0:C_SPLB_NUM_ MASTERS - 1]	PLB	O	0	Slave busy indicator
P41	SI_MWrErr[0:C_SPLB_NUM_ MASTERS - 1]	PLB	O	0	Slave write error indicator
P42	SI_MRdErr[0:C_SPLB_NUM_ MASTERS - 1]	PLB	O	0	Slave read error indicator
MCH Interface Signals[4]					
P43	MCHx_Access_Control[5]	MCH	I	-	Control signal to the access buffer of the MCH interface. This signal indicates the type of access to be performed (read or write) and the size of the access (byte, halfword or word)
P44	MCHx_Access_Data[0:C_MCH_NATIVE _DWIDTH - 1][5]	MCH	I	-	Write data to the access buffer of the MCH interface
P45	MCHx_Access_Write[5]	MCH	I	-	Write signal to access buffer of the MCH interface
P46	MCHx_Access_Full[5]	MCH	O	0	Indicates that the access buffer of the MCH interface is full
P47	MCHx_ReadData_Control[5]	MCH	O	0	Control signal for the read data buffer of the MCH interface. These signals indicates if the data from the read data buffer is valid
P48	MCHx_ReadData_Data[0:C_ MCH_NATIVE_DWIDTH - 1][5]	MCH	O	0	Read data from the read data buffer of the MCH interface
P49	MCHx_ReadData_Read[5]	MCH	I	-	Read signal to the read data buffer of the MCH interface
P50	MCHx_ReadData_Exists[5]	MCH	O	0	Indicates that the read data buffer of the MCH interface is not empty
External Memory Signals					

Table 2: **I/O Signals *(Cont'd)***

Port	Signal Name	Interface	I/O	Default Value	Description
P51	MEM_DQ_I[0:C_MAX_MEM_ WIDTH - 1]	External memory	I	-	Memory input data bus
P52	MEM_DQ_O[0:C_MAX_MEM_ WIDTH - 1]	External memory	O	0	Memory output data bus
P53	MEM_DQ_T[0:C_MAX_MEM_ WIDTH - 1]	External memory	O	0	Memory output 3-state signal
P54	MEM_A[0:C_MCH_SPLB_ AWIDTH - 1]	External memory	O	0	Memory address bus
P55	MEM_RPN	External memory	O	1	Memory reset/power down
P56	MEM_CEN[0:C_NUM_BANKS_MEM - 1]	External memory	O	1	Memory chip enables(6) (active low)
P57	MEM_OEN[0:C_NUM_BANKS_MEM - 1]	External memory	O	1	Memory output enable
P58	MEM_WEN	External memory	O	1	Memory write enable
P59	MEM_QWEN[0:(C_MAX_MEM_WIDTH/ 8) - 1]	External memory	O	1	Memory qualified write enables
P60	MEM_BEN[0:(C_MAX_MEM_ WIDTH/8) - 1]	External memory	O	0	Memory byte enables
P61	MEM_CE[0:C_NUM_BANKS_ MEM - 1]	External memory	O	0	Memory chip enables(6) (active high)
P62	MEM_ADV_LDN	External memory	O	1	Memory advance burst address/load new address
P63	MEM_LBON	External memory	O	1	Memory linear/interleaved burst order
P64	MEM_CKEN	External memory	O	0	Memory clock enable
P65	MEM_RNW	External memory	O	1	Memory read not write

Notes:

1. User must connect this port manually when C_INCLUDE_PLB_IPIF=0
2. This clock is used to capture the data from memory. Connection to this port is must otherwise design will fail. Generally this should be connected to system/bus clock. User can connected this to other clock nets e.g. phase shift clock or feedback clock.
3. The signals in this section are unused if C_INCLUDE_PLB_IPIF = 0. The inputs are ignored and the outputs are driven to zero.
4. MCH signals are unused when C_NUM_CHANNELS = 0.
5. x = values for XCL channels 0 to 3.
6. Most asynchronous memory devices will only use MEM_CEN. Most synchronous memory devices will use both MEM_CEN and MEM_CE. Refer to the device data sheet for correct connection of these signals.

Allowable Parameter Combinations

Note that it is assumed all the external memory devices are accessible through the MCH interface and the PLB interface, if the design has been parameterized to include the PLB interface.

The PLB interface is included in the design only if C_INCLUDE_PLB_IPIF is set to 1. When C_INCLUDE_PLB_IPIF = 0, then the C_INCLUDE_WRBUF is unused.

If C_SYNCH_MEM_x = 1, then C_SYNCH_PIPEDELAY_x specifies the pipeline delay of that synchronous memory type. All other timing parameters for that memory bank can remain at the default value of 0. If C_SYNCH_MEM_x = 0, C_SYNCH_PIPEDELAY_x is unused. All other timing parameters for that memory bank must be set to the value specified in the memory device data sheet.

C_INCLUDE_NEGEDGE_IOREGS provides no benefit when interfacing to asynchronous memories. Therefore, if there are no synchronous memories in the system, set this parameter to 0.

Optimal MCH Parameter Settings

The only channel transfer protocol supported at this time is the Xilinx Cachelink (XCL) interface.

If an XCL channel is connected to a master that perform read transactions only, set C_XCLx_WRITEXFER to 0. This setting reduces the channel logic to only contain logic for read transactions, therefore no write transfers will be performed.

If an XCL channel is connected to a master that can consume data as soon as it is available, set C_MCHx_RDDATABUF_DEPTH for that channel to 0 which will eliminate the read data buffer and the latency that would normally exist while reading data from this buffer. If the master cannot consume data as soon as it is available, set C_MCHx_RDDATABUF_DEPTH for that channel to accommodate any latency that the master has while reading data from the read data buffer.

Optimal performance will be achieved when the buffer depth of the access buffer is set greater than or equal to the line size of the channel (C_MCHx_ACCESSBUF_DEPTH ≥ C_XCLx_LINESIZE).

Parameter Port Dependencies

The dependencies between the design parameters and the I/O signals are described in Table 3. When certain features are parameterized out of the design, the related logic will no longer be part of the design. The unused input signals and related output signals are set to a specified value

Table 3: **Parameter Port Dependencies**

Generic or Port	Parameter	Affects	Depends	Description
Design Parameters				
G2	C_NUM_BANKS_MEM	P56, P57, P61	-	Specifies the number of external memory (SRAMs, Flash) banks
G4	C_INCLUDE_PLB_IPIF	P4 - P42	-	PLB input signals are unused and output signals are tied to 0 when C_INCLUDE_PLB_IPIF = 0
G5	C_INCLUDE_WRBUF	-	G4	Unused when C_INCLUDE_PLB_IPIF = 0
G6	C_SPLB_MID_WIDTH	P6	G4, G7	Specifies width of master ID bus Unused when C_INCLUDE_PLB_IPIF = 0 Width is equal to $\log_2$(C_SPLB_NUM_MASTERS)
G7	C_SPLB_NUM_MASTERS	P40 - P42	G4	Defines width of slave response signals Unused when C_INCLUDE_PLB_IPIF = 0
G8	C_SPLB_P2P	-	G4	Unused when C_INCLUDE_PLB_IPIF = 0
G9	C_SPLB_DWIDTH	P8, P13, P35	-	Data bus width of MCH/PLB interface
G10	C_MCH_SPLB_AWIDTH	P7	-	Address bus width of MCH/PLB interface

Table 3: **Parameter Port Dependencies**

Generic or Port	Parameter	Affects	Depends	Description
G11	C_SPLB_SMALLEST_MASTER	-	G4	Unused when C_INCLUDE_PLB_IPIF = 0
G12	C_MCH_NATIVE_DWIDTH	P44, P48	G4,G9, G31	Defines data bus width of MCH interface signals and C_SPLB_DWIDTH. C_MCH_NATIVE_DWIDTH should be always less than or equal to C_SPLB_DWIDTH
G23	C_TPACC_PS_FLASH_x		G14	Unused when C_PAGEMODE_FLASH_x =0
G27	C_MAX_MEM_WIDTH	P51, P52, P53, P59, P60	-	Maximum data bus width of the external memory (SRAMs, Flash) devices in all banks
G31	C_NUM_CHANNELS	P43 - P50	-	For all the channels numbered greater than the value of the parameter C_NUM_CHANNELS, the MCH interface input signals are unused and the output signals are tied to the default value
G32	C_MCHx_PROTOCOL	-	G31	Unused when x is greater than C_NUM_CHANNELS - 1
G33	C_MCHx_ACCESSBUF_DEPTH	-	G31	Unused when x is greater than C_NUM_CHANNELS - 1
G34	C_MCHx_RDDATABUF_DEPTH	-	G31	Unused when x is greater than C_NUM_CHANNELS - 1
G35	C_XCLx_LINESIZE	-	G31	Unused when x is greater than C_NUM_CHANNELS - 1
G36	C_XCLx_WRITEXFER	-	G31	Unused when x is greater than C_NUM_CHANNELS - 1
I/O Signals				
P4	PLB_ABus[0 : 31]	-	G4	PLB address bus
P5	PLB_PAValid		G4	PLB primary address valid indicator
P6	PLB_masterID[0:C_SPLB_MID_WIDTH - 1]	-	G4,G6	PLB current master identifier
P7	PLB_RNW		G4,G10	PLB read not write
P8	PLB_BE[0:(C_SPLB_WIDTH/8) - 1]	-	G4,G9	PLB byte enables
P9	PLB_wrBurst		G4	PLB burst write transfer indicator
P10	PLB_rdBurst	-	G4	PLB burst read transfer indicator
P11	PLB_size[0 : 3]		G4	PLB transfer size
P12	PLB_type[0 : 2]	-	G4	PLB transfer type
P13	PLB_wrDBus[0:C_SPLB_DWIDTH - 1]		G4,G9	PLB write data bus
P14	PLB_MSize[0 : 1]	-	G4	PLB master data bus size
P15	PLB_UABus[0 : 31]		G4	PLB upper address bus
P16	PLB_SAValid		G4	PLB secondary address valid indicator
P17	PLB_rdPrim	-	G4	PLB secondary to primary read request indicator

Table 3: **Parameter Port Dependencies**

Generic or Port	Parameter	Affects	Depends	Description
P18	PLB_wrPrim		G4	PLB secondary to primary write request indicator
P19	PLB_abort		G4	PLB abort bus indicator
P20	PLB_busLock		G4	PLB bus lock
P21	PLB_TAttribute[0 : 15]	-	G4	PLB transfer attribute bus
P22	PLB_lockerr		G4	PLB lock error indicator
P23	PLB_wrPendReq		G4	PLB pending write bus request indicator
P24	PLB_rdPendReq		G4	PLB pending read bus request indicator
P25	PLB_rdPendPri[0 : 1]	-	G4	PLB pending write request priority
P26	PLB_wrPendPri[0 : 1]		G4	PLB pending write request priority
P27	PLB_reqPri[0 : 1]		G4	PLB current request priority
P28	Sl_addrAck		G4	Slave address acknowledge
P29	Sl_SSize[0 : 1]		G4	Slave data bus size
P30	Sl_wait		G4	Slave wait indicator
P31	Sl_rearbitrate		G4	Slave rearbitrate bus indicator
P32	Sl_wrDack		G4	Slave write data acknowledge
P33	Sl_wrComp		G4	Slave write transfer complete indicator
P34	Sl_wrBTerm		G4	Slave terminate write burst transfer
P35	Sl_rdBus[0:C_SPLB_ DWIDTH - 1]		G4,G9	Slave read data bus
P36	Sl_rdDAck		G4	Slave read data acknowledge
P37	Sl_rdComp		G4	Slave read transfer complete indicator
P38	Sl_rdBTerm		G4	Slave terminate read burst transfer
P39	Sl_rdWdAddr[0 : 3]		G4	Slave read word address
P40	Sl_MBusy[0:C_SPLB_NUM_ MASTERS - 1]		G4	Slave busy indicator
P41	Sl_MWrErr[0:C_SPLB_NUM_ MASTERS - 1]		G4	Slave write error indicator
P42	Sl_MRdErr[0:C_SPLB_NUM_ MASTERS - 1]		G4	Slave read error indicator
P43	MCHx_Access_Contro		G31	Control signal to the access buffer of the MCH interface. This signal indicates the type of access to be performed (read or write) and the size of the access (byte, halfword or word)
P44	MCHx_Access_Data[0:C_MCH_NATIV E_DWIDTH - 1]		G12,G31	Write data to the access buffer of the MCH interface
P45	MCHx_Access_Write		G31	Write signal to access buffer of the MCH interface
P46	MCHx_Access_Full		G31	Indicates that the access buffer of the MCH interface is full

Table 3: **Parameter Port Dependencies**

Generic or Port	Parameter	Affects	Depends	Description
P47	MCHx_ReadData_Control		G31	Control signal for the read data buffer of the MCH interface. These signals indicates if the data from the read data buffer is valid
P48	MCHx_ReadData_Data[0:C_MCH_NATIVE_DWIDTH - 1]		G12,G31	Read data from the read data buffer of the MCH interface
P49	MCHx_ReadData_Read		G31	Read signal to the read data buffer of the MCH interface
P50	MCHx_ReadData_Exists		G31	Indicates that the read data buffer of the MCH interface is not empty
P51	MEM_DQ_I[0:C_MAX_MEM_WIDTH - 1]	-	G27	External memory (SRAMs, Flash) input data bus
P52	MEM_DQ_O[0:C_MAX_MEM_WIDTH - 1]	-	G27	External memory (SRAMs, Flash) output data bus
P53	MEM_DQ_T[0:C_MAX_MEM_WIDTH - 1]	-	G27	External memory (SRAMs, Flash) output 3-state signal
P54	MEM_A[0:C_MCH_SPLB_AWIDTH - 1]	-	G10	External memory (SRAMs, Flash) address bus
P56	MEM_CEN[0:C_NUM_BANKS_MEM - 1]	-	G2	External memory (SRAMs, Flash) active low chip enables
P57	MEM_OEN[0:C_NUM_BANKS_MEM - 1]	-	G2	External memory (SRAMs, Flash) active low output enables
P59	MEM_QWEN[0:C_MAX_MEM_WIDTH/8 - 1]	-	G27	External memory (SRAMs, Flash) qualified write enables
P60	MEM_BEN[0:C_MAX_MEM_WIDTH/8 - 1]	-	G27	External memory (SRAMs, Flash) byte enables
P61	MEM_CE[0:C_NUM_BANKS_MEM - 1]	-	G2	External memory (SRAMs, Flash) active high chip enables

XCL Channel Interface

The XPS MCH EMC core supports both linear Address cacheline transaction and Target word first of 1, 4, 8, and 16 words XCL cache line transactions

However, the Microblaze™ processor, based on its version, supports various versions of XCL as described in Table 4.

Selecting an appropriate assignment to the Microblaze parameter, C_XCL_x_SUBTYPE, may improve the performance of the logic utilization, frequency, and latency of the system.

Table 4: **XCL**

No	C_XCL_WRITEXFER	IXCL	IXCL2	DXCL	DXCL2		XCL	
		-	-	-	-	0	1	2
1	Write single word (32-Bit)			X	X		X	
2	Write single half word (16-Bit)			X	X		X	
3	Write single byte (8-Bit)			X	X		X	
4	Read Micro Blaze Cacheline (4-,8- wordline size), first failing word	X		X		X	X	
5	Read Micro Blaze Cacheline (4-,8- wordline size), linear address		X		X			
6	Write Micro Blaze Cacheline (4-,8- wordline size), linear address				X			
7	Read Cacheline (1-, 4-, 8-, 16- wordline size), first failing word					X	X	X
8	Write Cacheline (1-, 4-, 8-, 16- wordline size), first failing word							X

Address Map Description

As shown in the Table 5, the this core supports up to four banks of external memory. The number of available banks used is determined by the value of the C_NUM_BANKS_MEM parameter. This parameter may take values between 1 and 4, inclusive. The banks used, if any, are banks 0 to C_NUM_BANKS_MEM - 1. Each bank of memory has it's own independent base address and high address range. The address range of a bank of memory is restricted a byte size that is in terms of power of 2 and address-aligned to the same power of 2. This means that for an address-range of size is 2^n, the n least significant bits of the base address will be 0 and n least significant bits of the high address will be 1. For example, a memory bank with an addressable range of 16M (2^{24}) bytes could have a base address of 0xFF000000 and a high address of 0xFFFFFFFF. A memory bank with an addressable range of 64K (2^{16}) bytes could have a base address of 0xABCD0000 and a high address of 0xABCDFFFF. The core transactions must fall between the Bank x base address C_MEMx_BASEADDR and high address C_MEMx_HIGHADDR.

The addresses for memory bank is shown in Table 5.

Table 5: **XPS MCH EMC Memory Bank**

Memory	Base Address	High Address	Access
Bank 0	C_MEM0_BASEADDR	C_MEM0_HIGHADDR	Read/Write
Bank 1	C_MEM1_BASEADDR	C_MEM1_HIGHADDR	Read/Write
Bank 2	C_MEM2_BASEADDR	C_MEM2_HIGHADDR	Read/Write
Bank 3	C_MEM3_BASEADDR	C_MEM3_HIGHADDR	Read/Write

Memory Data Types and Organization

Memory can be accessed through the XPS MCH EMC core as one of four types:

- byte (8-bit)
- halfword (16-bit)
- word (32-bit)
- doubleword (64-bit)

Data to and from the MCH and PLB is organized and supported as big-endian. The bit and byte labeling for the big-endian data types is shown in Figure 4.

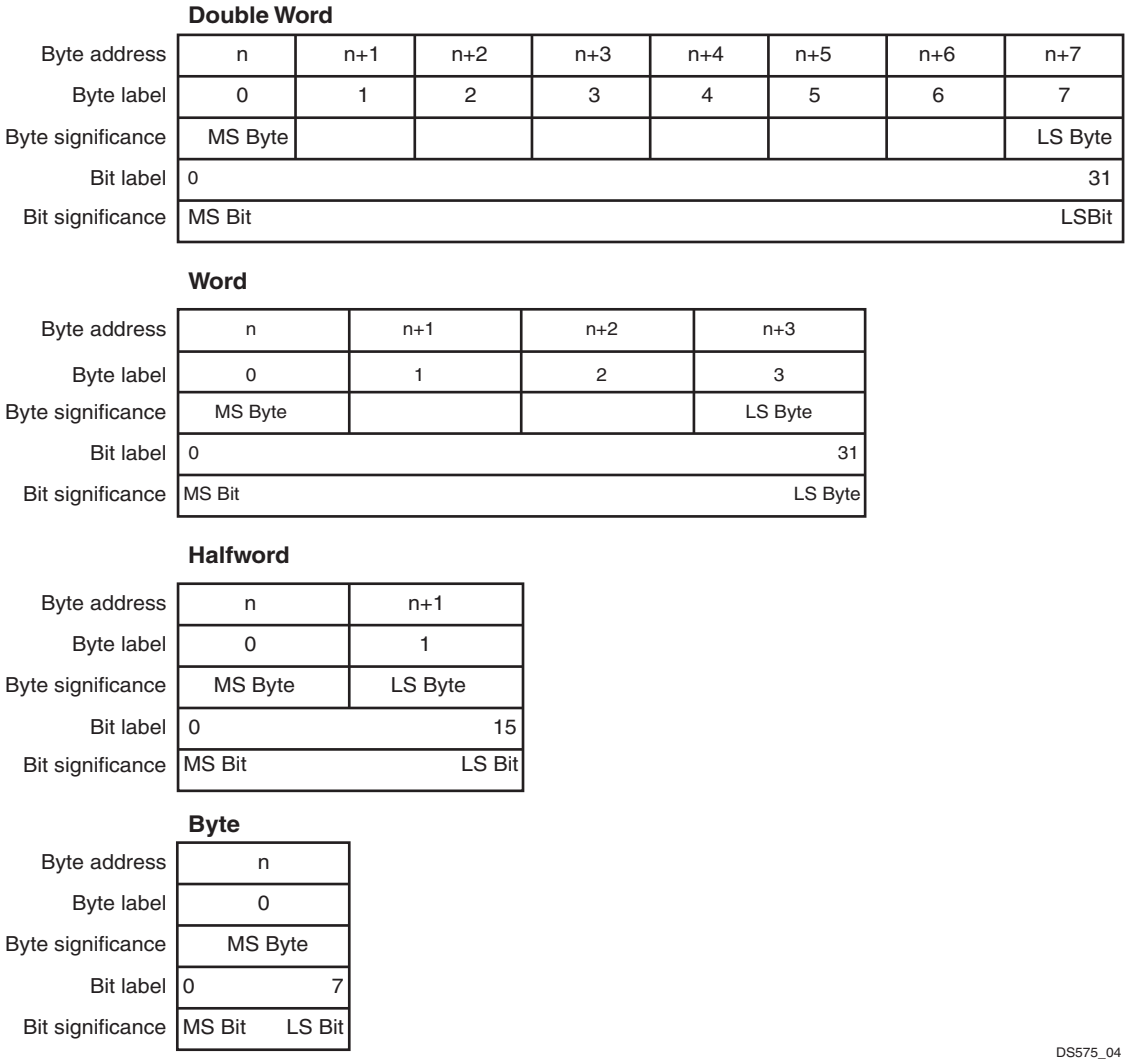

Figure 4: **Memory Data Types**

Connecting to Memory

Clocking Synchronous Memory

The XPS MCH EMC core does not provide a clock output to any synchronous memory. Route the MCH/PLB Clock through an output buffer to provide the clock to the synchronous memory.

To synchronize the synchronous memory clock to the internal FPGA clock, the FPGA system design should include a DCM external to the XPS MCH EMC core that uses the synchronous memory clock input as the feedback clock as shown in Figure 5. This means that the synchronous clock output from the FPGA must be routed back to the FPGA on a clock pin with a connection to a DCM clock feedback input.

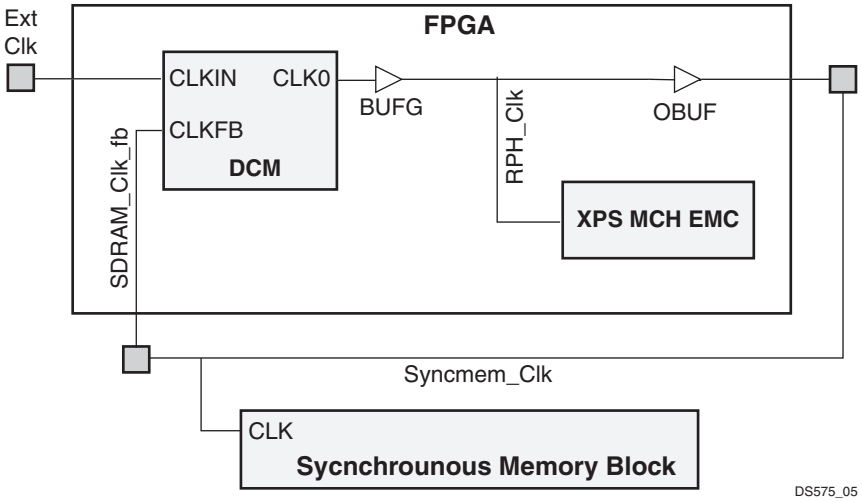

Figure 5: **Synchronous Memory Bank Clocked by FPGA Output With Feedback**

If the synchronous memory is clocked by the same external clock as the FPGA, or if the clock feedback is not available, include the DCM in the FPGA, external to the XPS MCH EMC core, as shown in Figure 6.

Note: If DLLs are used, the designer must reference the application note, XAPP132 v2.4 Using the Virtex Delay-Locked Loop, for the correct DLL implementation.

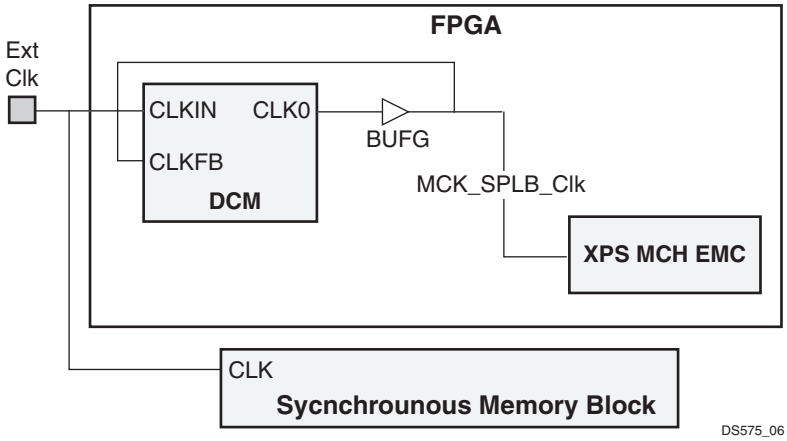

Figure 6: **Synchronous Memory Bank Clocked by External Clock**

Address Bus, Data Bus and Control Signal Connections

The three primary considerations for connecting the controller to memory devices are the width of the MCH/PLB data bus, the width of the memory subsystem, and the number of memory devices used.

The data and address signals at the memory controller are labeled with big-endian bit labeling (for example: D(0 : 31), D(0) is the MSB), whereas most memory devices are either endian agnostic (they can be connected either way) or little-endian D(31 : 0) with D(31) as the MSB.

Take care when connecting the chip enable signals. Most asynchronous memory devices will use MEM_CEN only, while both MEM_CEN and MEM_CE is used by most synchronous memory devices. MEM_CEN is a function of the address decode while MEM_CE is a function of the state machine logic.

Take care with the connections to the external memory devices to avoid incorrect data and address connections. The following tables show the correct mapping of memory controller pins to memory device pins.

Table 6 shows variables used in defining memory subsystem and Table 7 shows interconnection of the core signals to the memory interface signals.

Table 6: **Variables Used In Defining Memory Subsystem**

Variable	Allowed Range	Definition
BN	0 to 3	Memory bank number
DN	0 to 63	Memory device number within a bank. The memory device attached to the most significant bit in the memory subsystem is 0; device numbers increase toward the least significant bit.
MW	8 to 64	Width in bits of memory subsystem
DW	1 to 63	Width in bits of data bus for memory device
MAW	1 to 32	Width in bits of address bus for memory device
AU	1 to 63	Width in bits of smallest addressable data word on the memory device
AS	X(1)	Address shift for address bus = $\log_2((MW*AU/DW)/8)$
HAW	1 to 32	Width in bits of MCH/PLB address bus

Notes:
1. The value of X depends on variables MW, AU and DW.

Connecting to SRAM

Table 7: **XPS MCH EMC To Memory Interconnect**

Description	XPS MCH EMC Signals (MSB:LSB)	Memory Device Signals (MSB:LSB)
Data bus	MEM_DQ(DN*DW:((DN+1)*DW)-1)	D(DW-1 : 0)
Address bus	MEM_A(HAW-MAW-AS:HAW-AS-1)	A(MAW-1 : 0)
Chip enable (active low)	MEM_CEN(BN)	CEN
Output enable (active low)	MEM_OEN	OEN
Write enable (active low)	MEM_WEN	WEN (for devices that have byte enables)
Qualified write enable (active low)	MEM_QWEN(DN*DW/8)	WEN (for devices that do not have byte enables)
Byte enable (active low)	MEM_BEN((DN*DW/8) : (((DN+1)*DW/8)-1))	BEN(DW/8-1 : 0)

Example 1: Connection to 32-bit memory using two IDT71V416S SRAM parts.

Table 8 shows variables for simple SRAM example.

Table 8: **Variables For Simple SRAM Example**

Variable	Value	Definition
BN	0	Memory bank number
DN	0 to 1	Memory device number within a bank. The memory device attached to the most significant bit in the memory subsystem is 0; device numbers increase toward the least significant bit.
MW	32	Width in bits of memory subsystem
DW	16	Width in bits of data bus for memory device
MAW	18	Width in bits of address bus for memory device
AU	16	Width in bits of smallest addressable data word on the memory device
AS	2	Address shift for address bus = $\log_2((MW*AU/DW)/8)$
HAW	32	Width in bits of host address bus (e.g. MCH/PLB)

Table 9 shows connection to 32-bit memory using 2 IDT71V416S (256K X 16-Bit) parts.

Table 9: **Connection To 32-bit Memory Using two IDT71V416S Parts**

DN	Description	XPS MCH EMC Signals (MSB:LSB)	Memory Device Signals (MSB:LSB)
0	Data bus	MEM_DQ(0 : 15)	I/O(15 : 0)
	Address bus	MEM_A(12 : 29)	A(17 : 0)
	Chip enable (active low)	MEM_CEN(0)	CS
	Output enable (active low)	MEM_OEN	OE
	Write enable (active low)	MEM_WEN	WE
	Byte enable (active low)	MEM_BEN(0 : 1)	$\overline{BLE}$:$\overline{BHE}$
1	Data bus	MEM_DQ(16 : 31)	I/O(15 : 0)
	Address bus	MEM_A(12 : 29)	A(17 : 0)
	Chip enable (active low)	MEM_CEN(0)	CS
	Output enable (active low)	MEM_OEN	OE
	Write enable (active low)	MEM_WEN	WE
	Byte enable (active low)	MEM_BEN(2 : 3)	$\overline{BLE}$:$\overline{BHE}$

Connecting to Intel StrataFlash

Because StrataFlash parts contain an identifier register, a status register, and a command interface, the bit label ordering for these parts is critical to function properly. Table 10 shows and example of how to connect the big-endian XPS MCH EMC bus to the little-endian StrataFlash parts.

The proper connection ordering is also indicated in a more general form in Table 7. StrataFlash parts have an x8 mode and an x16 mode which is selectable with the BYTE# input pin. To calculate the proper address shift, the minimum addressable word is 8 bits for both the x8 and x16 modes, because A0 always selects a byte.

Example 2: Connection to 32-bit memory using two StrataFlash parts in x16 mode.

Supports byte read, but not byte write; the smallest data type that can be written is 16-bit data.

Table 10 shows variables for StrataFlash (x16 mode) example.

Table 10: **Variables For StrataFlash (x16 mode) Example**

Variable	Value	Definition
BN	0	Memory bank number
DN	0 to 1	Memory device number within a bank. The memory device attached to the most significant bit in the memory subsystem is 0; device numbers increase toward the least significant bit.
MW	32	Width in bits of memory subsystem
DW	16	Width in bits of data bus for memory device
MAW	24	Width in bits of address bus for memory device
AU	8	Width in bits of smallest addressable data word on the memory device
AS	1	Address shift for address bus = $\log_2((MW*AU/DW)/8)$
HAW	32	Width in bits of host address bus (e.g. MCH/PLB)

Table 11 shows connection to 32-bit memory using two StrataFlash parts.

Table 11: **Connection To 32-bit Memory Using two StrataFlash Parts**

DN	Description	XPS MCH EMC Signals (MSB:LSB)	StrataFlash Signals (MSB:LSB)
0	Data bus	MEM_DQ(0 : 15)	DQ(15 : 0)
	Address bus	MEM_A(6 : 29)	A(23 : 0)
	Chip enable (active low)	GND,GND,MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(0)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}
1	Data bus	MEM_DQ(16 : 31)	DQ(15 : 0)
	Address bus	MEM_A(6 : 29)	A(23 : 0)
	Chip enable (active low)	GND, GND, MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(0)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}

Example 3: Connection to 32-bit memory using four StrataFlash parts in x8 mode.

Supports byte reads and writes.

Table 12 shows variables for StrataFlash (x8 mode) example.

Table 12: **Variables For StrataFlash (x8 mode) Example**

Variable	Value	Definition
BN	0	Memory bank number
DN	0 to 3	Memory device number within a bank. The memory device attached to the most significant bit in the memory subsystem is 0; device numbers increase toward the least significant bit.
MW	32	Width in bits of memory subsystem
DW	8	Width in bits of data bus for memory device
MAW	24	Width in bits of address bus for memory device
AU	8	Width in bits of smallest addressable data word on the memory device
AS	2	Address shift for address bus = $\log_2((MW*AU/DW)/8)$
HAW	32	Width in bits of host address bus (e.g. MCH/PLB)

Table 13 shows connection to 32-bit memory using four StrataFlash parts.

Table 13: **Connection To 32-bit Memory Using four StrataFlash Parts**

DN	Description	XPS MCH EMC Signals (MSB:LSB)	StrataFlash Signals (MSB:LSB)
0	Data bus	MEM_DQ(0 : 7)	DQ(7 : 0)(1)
	Address bus	MEM_A(8 : 29)	A(21 : 0)
	Chip enable (active low)	GND, GND, MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(0)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}
1	Data bus	MEM_DQ(8 : 15)	DQ(7 : 0)(1)
	Address bus	MEM_A(8 : 29)	A(21 : 0)
	Chip enable (active low)	GND,GND, MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(1)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}

Table 13: **Connection To 32-bit Memory Using four StrataFlash Parts *(Cont'd)***

DN	Description	XPS MCH EMC Signals (MSB:LSB)	StrataFlash Signals (MSB:LSB)
2	Data bus	MEM_DQ(16 : 23)	DQ(7 : 0)(1)
	Address bus	MEM_A(8 : 29)	A(21 : 0)
	Chip enable (active low)	GND, GND, MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(2)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}
3	Data bus	MEM_DQ(24 : 31)	DQ(7 : 0)(1)
	Address bus	MEM_A(8 : 29)	A(21 : 0)
	Chip enable (active low)	GND, GND, MEM_CEN(0)	CE(2 : 0)
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(3)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to GND	BYTE#
	Program enable (active high)	N/A - tie to VCC	V_{PEN}

Notes:

1. In x8 configuration, DQ(15:8) are not used and should be treated according to manufacturer's data sheet.

Connecting to Spansion PageModeFlash S29GL032N

Table 14 shows variables for Spansion Page Mode Flash(x16 mode) example.

Table 14: **Variables For Page Mode Flash (x16 mode) Example**

Variable	Value	Definition
BN	0	Memory bank number
DN	0	Memory device number within a bank. The memory device attached to the most significant bit in the memory subsystem is 0.
MW	32	Width in bits of memory subsystem
DW	16	Width in bits of data bus for memory device
MAW	21	Width in bits of address bus for memory device
AU	16	Width in bits of smallest addressable data word on the memory device
AS	2	Address shift for address bus = $\log_2((MW*AU/DW)/2)$
HAW	32	Width in bits of host address bus (e.g. MCH/PLB)

Table 15: **Connection To 16-bit Memory Using PageModeFlash Parts**

DN	Description	XPS MCH EMC Signals (MSB:LSB)	PagemModeFlash Signals (MSB:LSB)
0	Data bus	MEM_DQ(0:15)	DQ(15 : 0)
	Address bus	MEM_A(9:30)	A(21 : 0)
	Chip enable (active low)	MEM_CEN(0)	CE
	Output enable (active low)	MEM_OEN	OE#
	Write enable (active low)	MEM_QWEN(0)	WE#
	Reset/Power down (active low)	MEM_RPN	RP#
	Byte mode select (active low)	N/A - tie to VCC	BYTE#
	Program enable (active high)	N/A - tie to VCC	WP#

Timing Diagrams

XCL Protocol Cacheline Write Timing Diagrams

Figure 7 shows an XCL channel performing a cacheline write on a 32-bit Asynchronous SRAM.

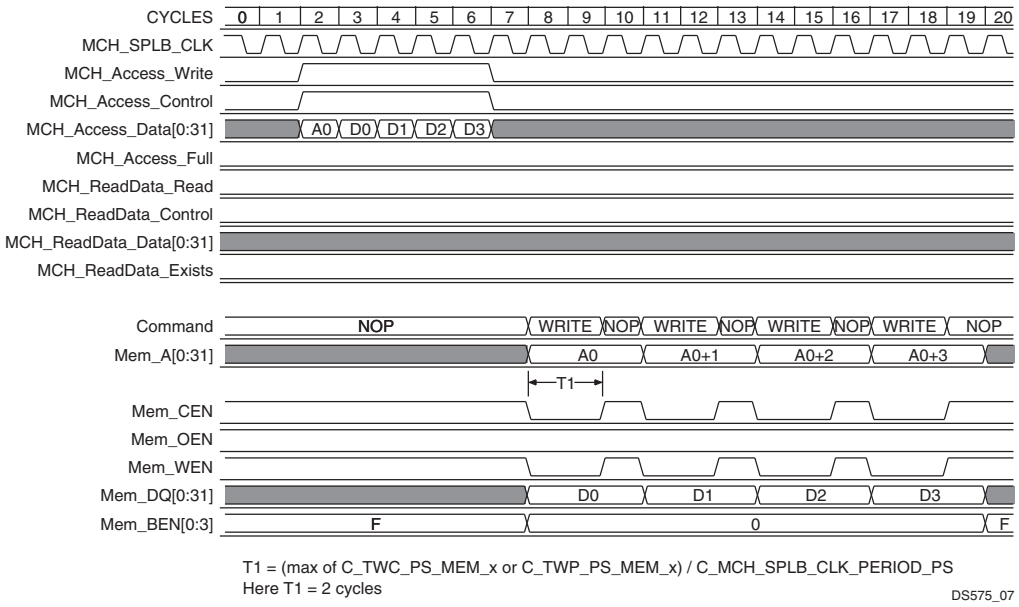

Figure 7: **XCL Cacheline Write Operation With 32-Bit Asynchronous SRAM**

Figure 8 shows an XCL channel performing a cacheline write on a 32-bit Synchronous SRAM.

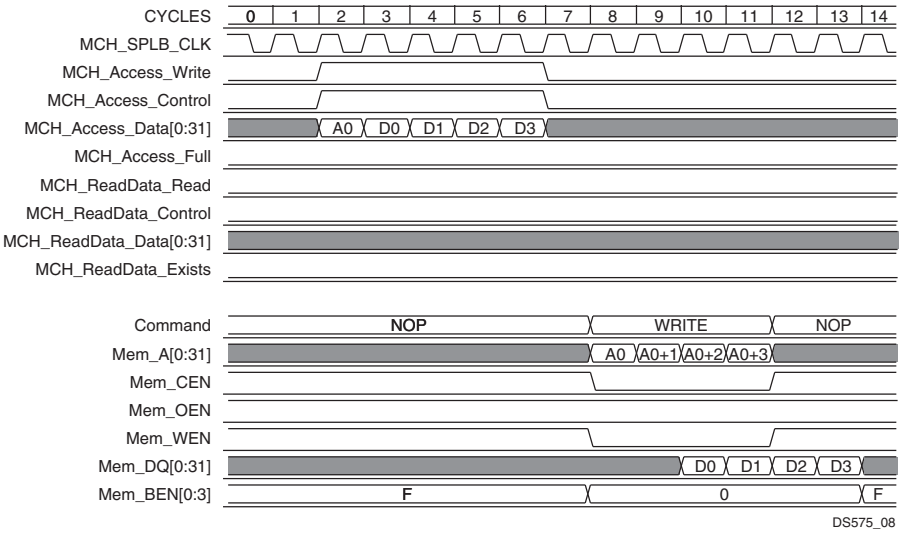

Figure 8: **XCL Cacheline Write Operation With 32-Bit Synchronous SRAM**

XCL Protocol Cacheline Read Timing Diagrams

Figure 9 shows an XCL channel performing a cacheline read on a 32-bit Asynchronous SRAM.

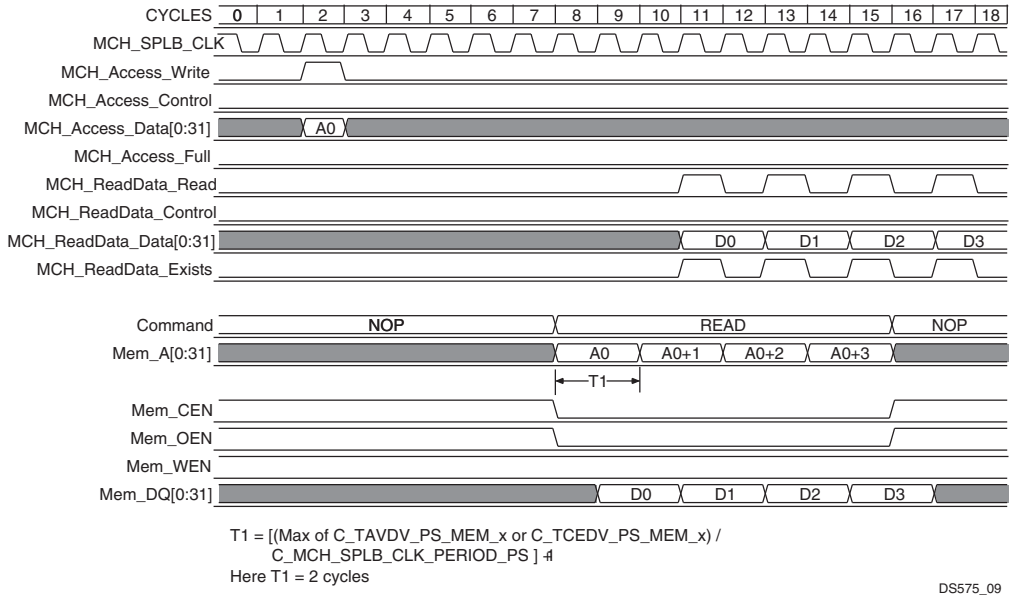

Figure 9: **XCL Cacheline Read Operation With 32-Bit Asynchronous SRAM**

Figure 10 shows an XCL channel performing a cacheline read on a 32-bit Synchronous SRAM.

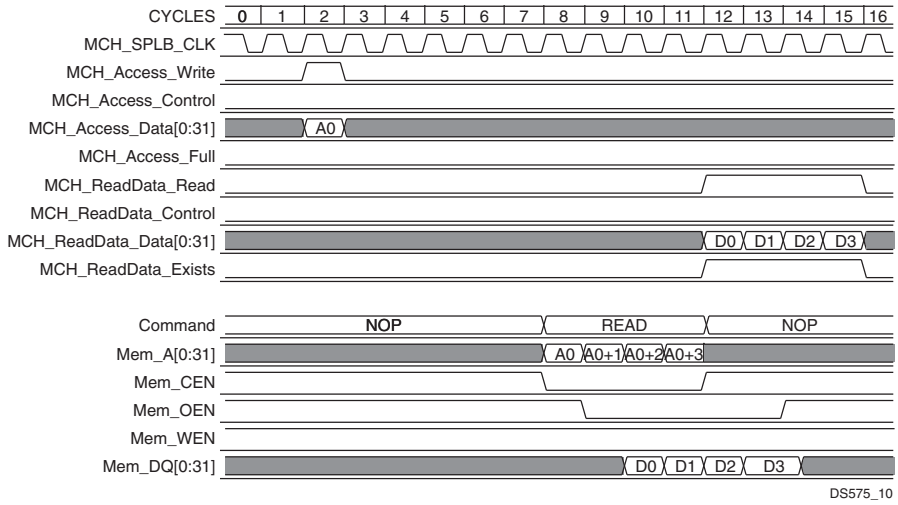

Figure 10: **XCL Cacheline Read Operation With 32-Bit Synchronous SRAM**

XCL Protocol Cacheline Write Read Timing Diagrams

Figure 11 shows an XCL channel performing a cacheline write followed by a cacheline read operation on a 32-bit Asynchronous SRAM.

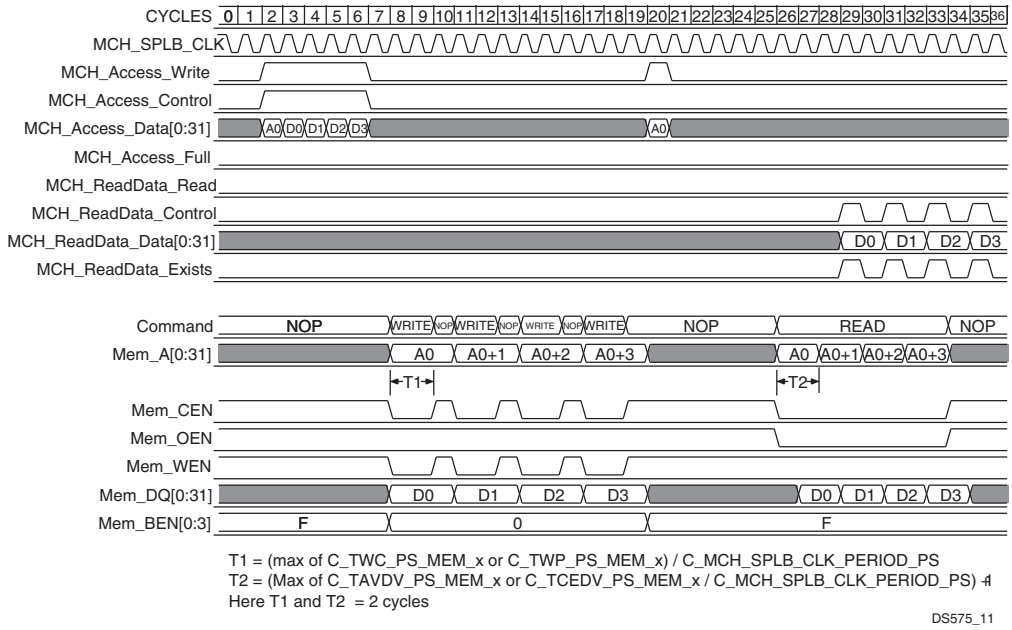

Figure 11: **XCL Cacheline Write Read Operation With 32-Bit Asynchronous SRAM**

Figure 12 shows an XCL channel performing a cacheline write followed by a cacheline read operation on a 32-bit Synchronous SRAM.

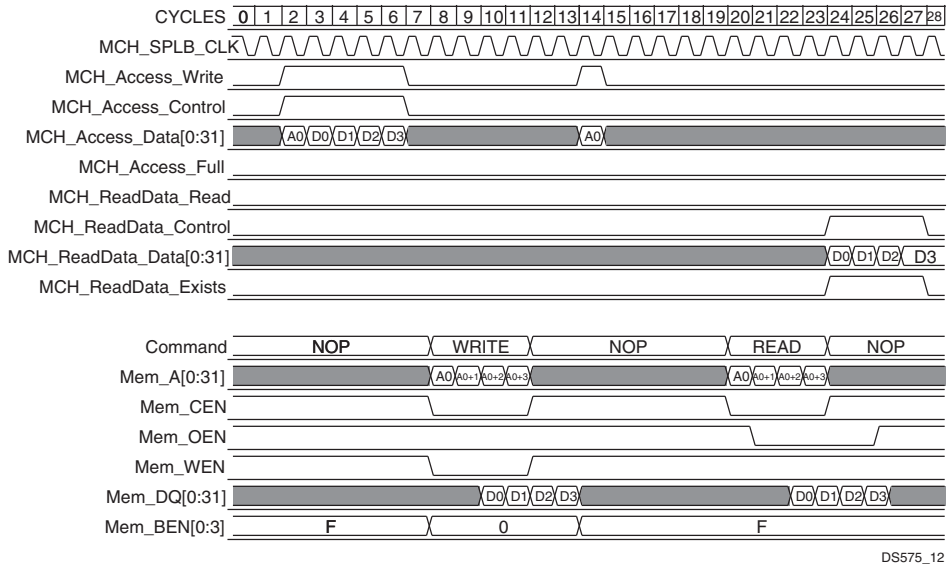

Figure 12: **XCL Cacheline Write Read Operation With 32-Bit Synchronous SRAM**

PLB Burst Write Timing Diagrams

Figure 13 shows a PLB burst write operation on a 32-bit Asynchronous SRAM.

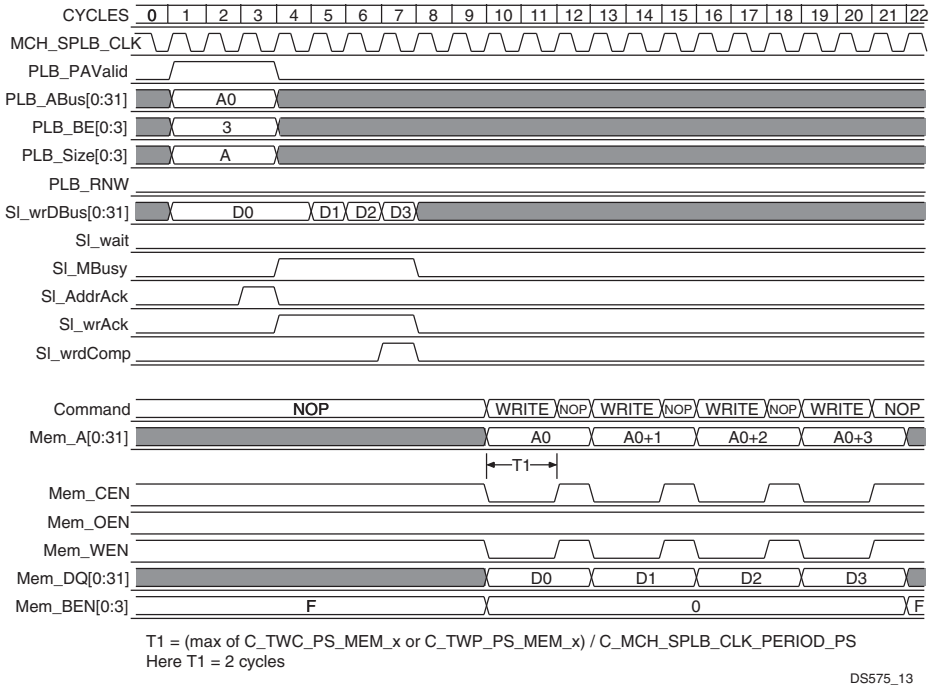

Figure 13: **PLB Burst Write Operation With 32-Bit Asynchronous SRAM**

Figure 14 shows a PLB burst write operation on a 32-bit Synchronous SRAM.

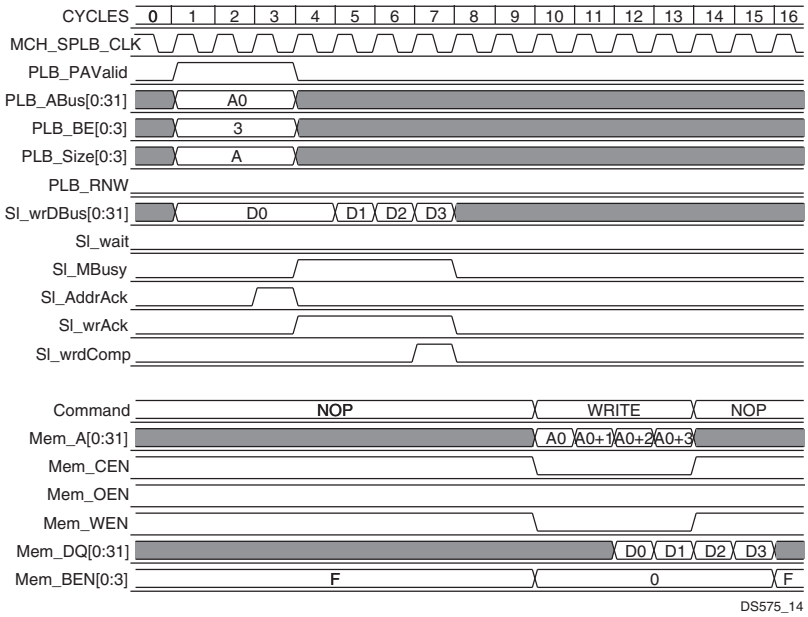

Figure 14: **PLB Burst Write Operation With 32-Bit Synchronous SRAM**

PLB Burst Read Timing Diagrams

Figure 15 shows a PLB burst read operation on a 32-bit Asynchronous SRAM.

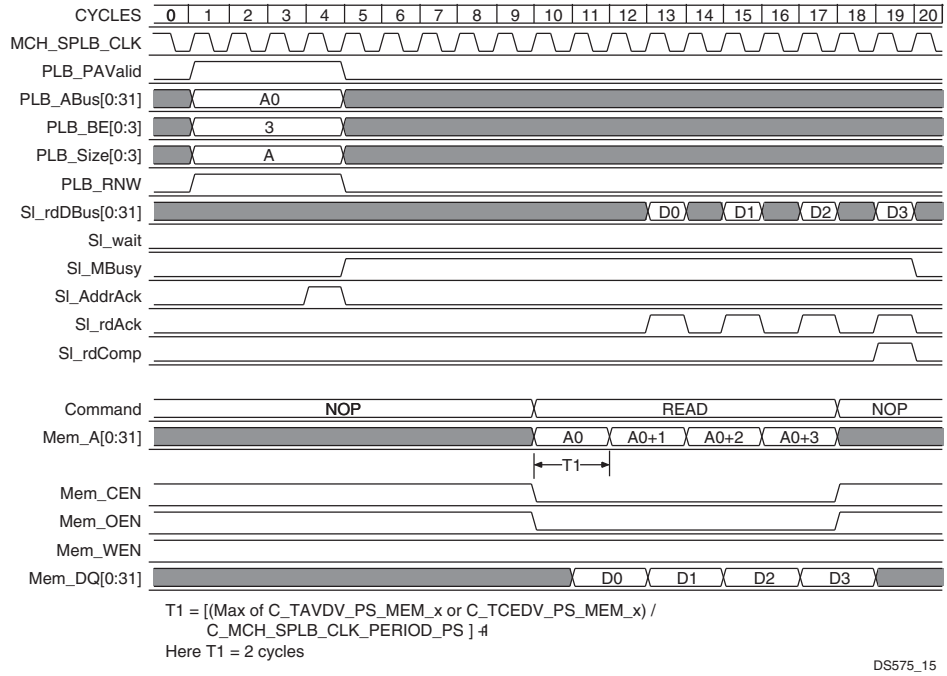

Figure 15: **PLB Burst Read Operation With 32-Bit Asynchronous SRAM**

Figure 16 shows a PLB burst read operation on a 32-bit Synchronous SRAM.

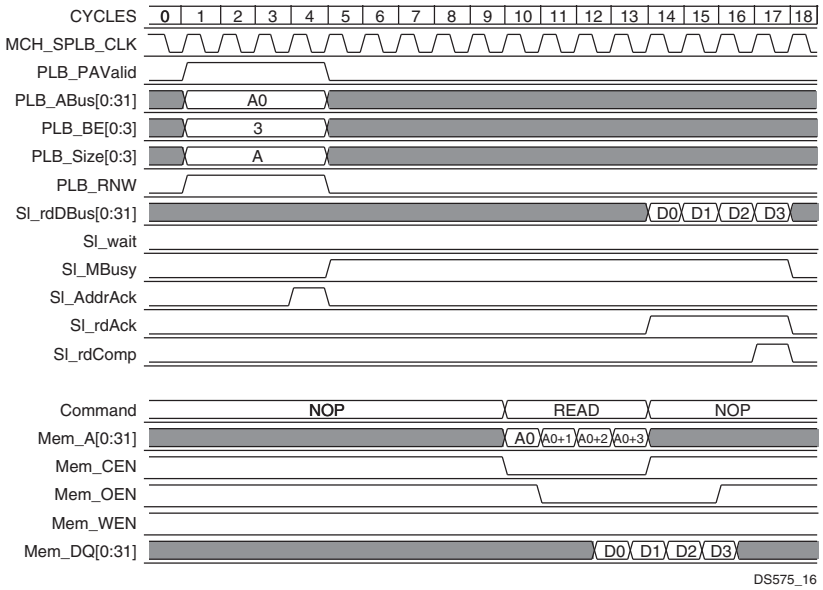

Figure 16: **PLB Burst Read Operation With 32-Bit Synchronous SRAM**

PLB Single Read Timing Diagrams

Figure 17 shows a PLB Single read operation on an 8-bit Synchronous SRAM.

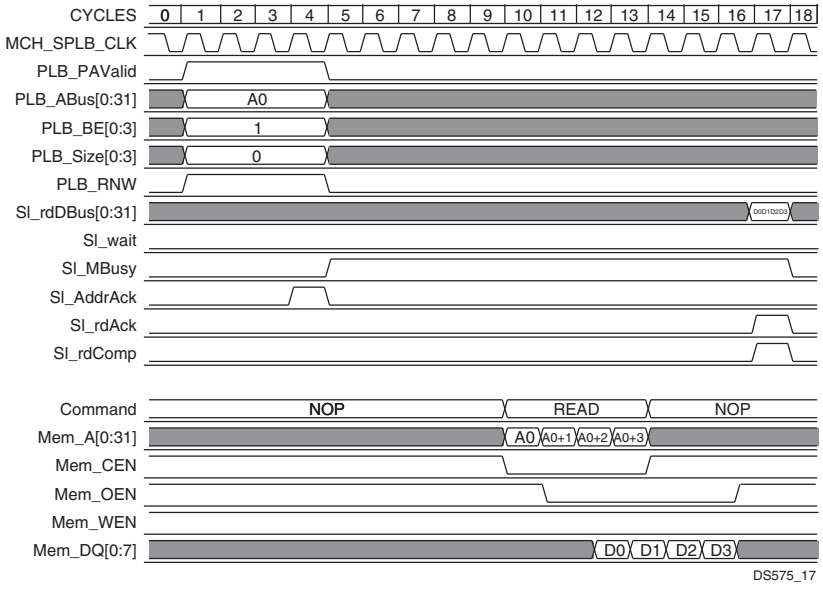

Figure 17: **PLB Single Read Operation With 8-Bit Synchronous SRAM**

Figure 18 shows a PLB Single read operation on a 16-bit Synchronous SRAM.

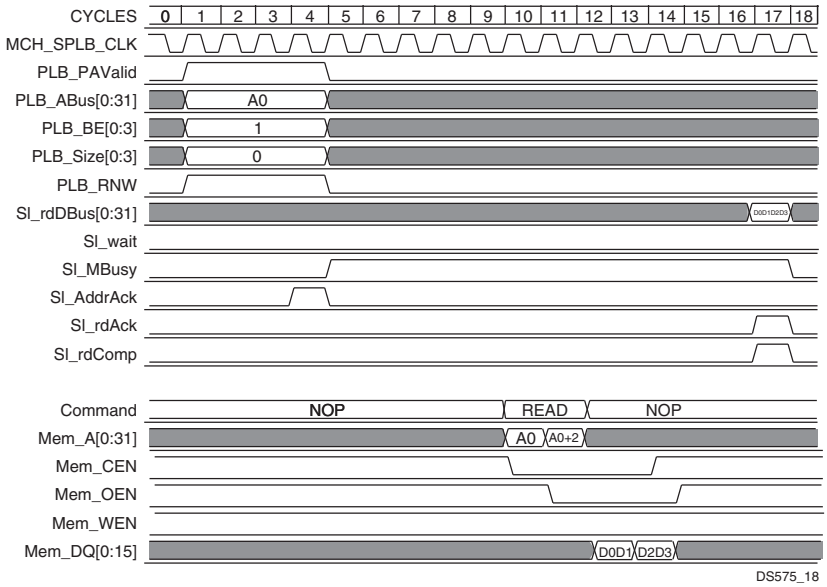

Figure 18: **PLB Single Read Operation With 16-Bit Synchronous SRAM**

PLB Single Write Timing Diagrams

Figure 19 shows a PLB Single Write operation on an 8-bit Synchronous SRAM.

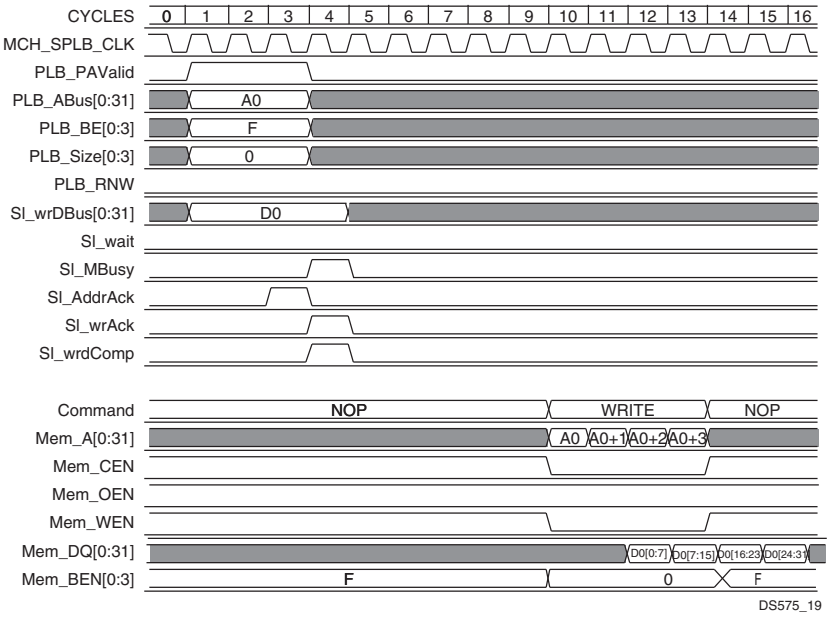

Figure 19: **PLB Single Write Operation With 8-Bit Synchronous SRAM**

Figure 20 shows a PLB Single read operation on a 16-bit Synchronous SRAM.

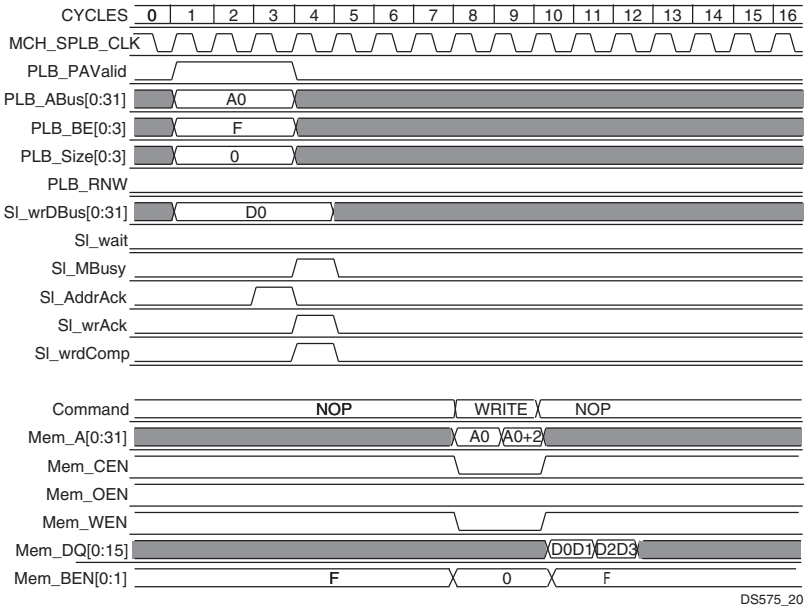

Figure 20: **PLB Single Write Operation With 16-Bit Synchronous SRAM**

PLB Burst Read IPIF 64 Timing Diagrams

Figure 21 shows a PLB Single read operation on a 32-bit Synchronous SRAM.

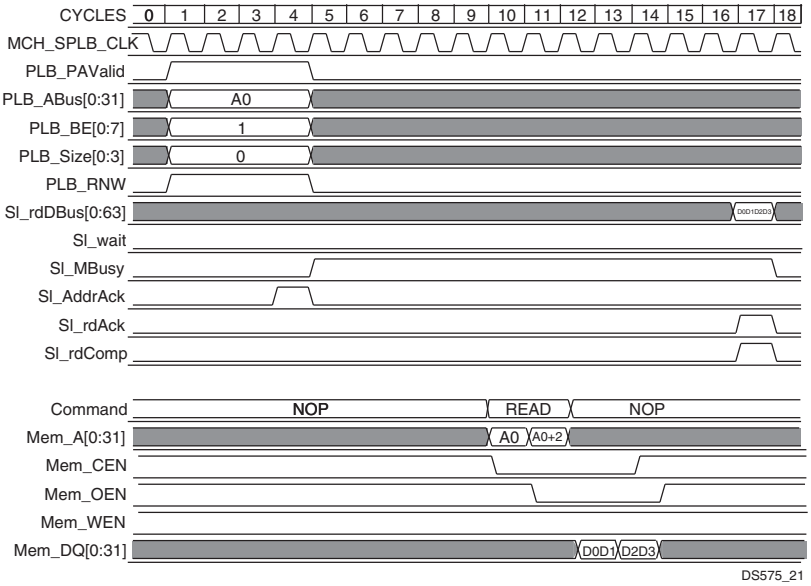

Figure 21: **PLB Burst Read Operation With 32-Bit Synchronous SRAM**

Page Mode Read Timing Diagrams

Figure 22 shows a PLB burst read operation on a 16-bit Page Mode flash.

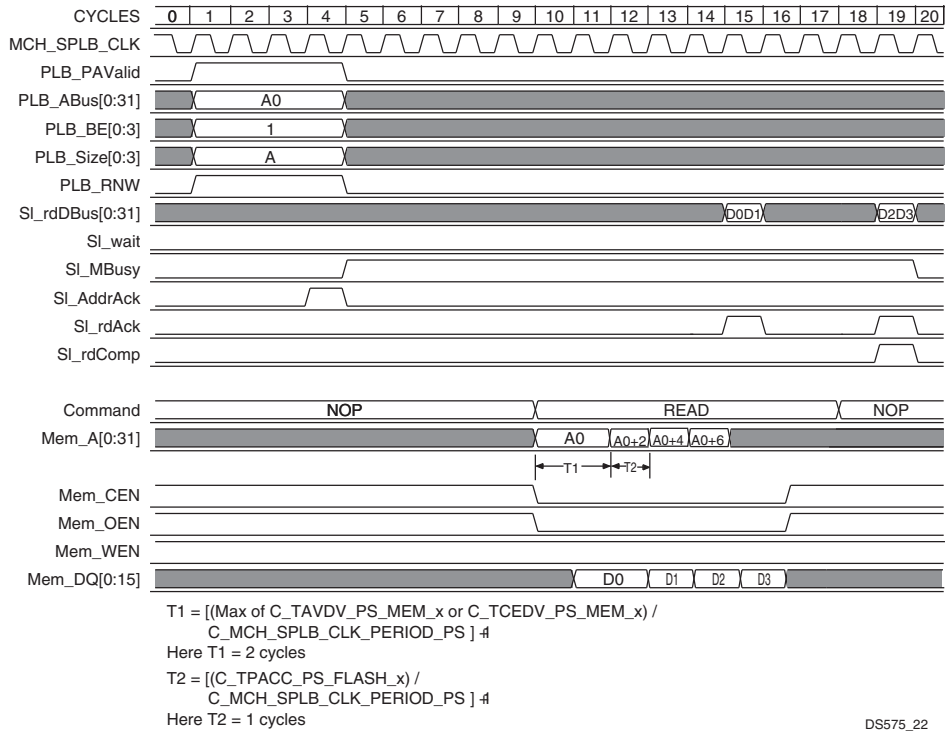

Figure 22: **PLB Burst Read Operation With 16-Bit Page Mode Flash**

Design Constraints

Timing Constraints

A timing constraint should be placed on the system clock to set the frequency to meet the bus timing requirements. An example is shown in Figure 23.

```
NET "MCH_SPLB" TNM_NET = "MCH_PLB_CLK";
TIMESPEC "TS_MCH_SPLB_Clk" = "MCH_PLB_Clk" 10 ns HIGH 50%
```

DS575_23

Figure 23: **XPS MCH EMC Timing Constraints**

Pin Constraints

If external pullups/pulldowns are not available on the MEM_DQ signals, those pins should be specified to use pullup or pulldown resistors. An example is shown in Figure 24.

```
NET "MEM_DQ<0>" PULLDOWN;
NET "MEM_DQ<1>" PULLDOWN;
NET "MEM_DQ<2>" PULLDOWN;
. . . .

NET "MEM_DQ<31>" PULLDOWN;
```

DS575_24

Figure 24: **XPS MCH EMC Pin Constraints**

Design Implementation

Target Technology

The target technology is an FPGA listed in the Supported Device Family field of the LogiCORE Facts.

Device Utilization and Performance Benchmarks

Core Performance

Because the XPS MCH EMC core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When the XPS MCH EMC core is combined with other designs in the system, the utilization of FPGA resources and timing of the XPS MCH EMC design will vary from the results reported here.

The XPS MCH EMC resource utilization benchmarks for various parameter combinations measured with the Virtex®-4 (xc4vlx25-11-ff676) FPGA as the target device are detailed in Table 16.

Table 16: **Performance and Resource Utilization Benchmarks for the Virtex-4 (XC4VLX100-10-FF1513) FPGA**

Parameter Values										Device Resources			F_{MAX} (MHz)
C_SPLB_DWIDTH, C_MCH_NATIVE_DWIDTH	C_NUM_BANKS_MEM	C_NUM_CHANNELS	C_INCLUDE_PLB_IPIF	C_INCLUDE_WRITE_BUFFER	C_MAX_MEM_WIDTH	C_PAGEMODE_FLASH_x	C_MEMx_WIDTH	C_SYNCH_MEM_x	C_SYNCH_PIPEDELAY_x	Slices	Slice Flip-Flops	LUTs	
32, 32	1	1	0	0	8	0,0,00	8	0	1	242	321	271	154
32, 32	1	4	1	1	8	0,0,00	8	1	2	1026	923	1575	136
32, 32	2	4	1	1	32	0,0,00	16,32	1,1	2,2	1042	1069	1619	134
32, 32	3	3	1	0	32	0,0,00	32,8,16	0,1,1	1,2,2	825	942	1246	131
32, 32	3	4	1	1	32	0,0,00	8,16,32	1,0,0	2,1,1	1421	1095	1753	125
32, 32	4	3	1	0	32	0,0,00	32,8,16,32	1,0,1,0	2,1,2,1	952	954	1267	127
32, 32	4	4	1	1	32	0,0,00	8,16,8,32	0,0,0,0	1,1,1,1	1250	1032	1478	143
32, 32	4	0	1	1	32	0,0,00	32,8,16,32	1,1,1,1	2,2,2,2	425	616	492	142
128,32	4	4	1	1	32	0,0,00	8,16,8,32	1,0,1,0	2,1,2,1	1205	1132	1874	126
32, 32	4	0	1	1	64	0,0,00	8,16,32,64	1,0,1,0	1,2,1,2	605	816	759	125
64, 64	4	0	1	1	64	0,0,00	8,16,32,64	0,0,0,0	1,2,1,2	827	836	839	125
64, 64	4	0	1	1	32	0,0,00	8,16,8,32	1,0,1,0	1,2,1,2	728	811	895	129
64, 64	4	0	1	1	64	0,0,00	8,16,32,664	1,1,1,1	1,2,1,2	809	965	939	126
64, 32	4	4	1	1	16	1,1,1,1	8,16,8,16	0,0,0,0	1,2,1,2	1084	1037	1887	126

Notes:

1. For all above cases, C_INCULDE_DATAWIDTH_MATCHING_x = 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
2. For all above cases, the rest of the parameters listed in Table 1 are assigned with the default values
3. All above cases for Virtex-4 are executed by providing MCH_SPLB_Clk = 8 ns (125 MHz) period constraint in UCF

The XPS MCH EMC resource utilization for various parameter combinations measured with Virtex-5 (XC5VLX110-1-FF1760) FPGA as the target device is detailed in Table 17.

Table 17: **Performance and Resource Utilization Benchmarks for the Virtex-5 (XC5VLX110-1-FF1760) FPGA**

Parameter Values										Device Resources			F_{MAX} (MHz)
C_SPLB_DWIDTH, C_MCH_NATIVE_DWIDTH	C_NUM_BANKS_MEM	C_NUM_CHANNELS	C_INCLUDE_PLB_IPIF	C_INCLUDE_WRITE_BUFFER	C_MAX_MEM_WIDTH	C_PAGEMODE_FLASH_x	C_MEMx_WIDTH	C_SYNCH_MEM_x	C_SYNCH_PIPEDELAY_x	Slices	Slice Flip-Flops	LUTs	
32, 32	1	1	0	0	8	0,0,00	8	0	1	170	357	302	254
32, 32	1	4	1	1	8	0,0,00	8	1	2	589	950	1219	176
32, 32	2	4	1	1	32	0,0,00	16,32	1,1	2,2	597	1107	1312	176
32, 32	3	3	1	0	32	0,0,00	32,8,16	0,1,1	1,2,2	581	1000	1080	183
32, 32	3	4	1	1	32	0,0,00	8,16,32	1,0,0	2,1,1	488	1143	1413	170
32, 32	4	3	1	0	32	0,0,00	32,8,16,32	1,0,1,0	2,1,2,1	449	1004	1096	176
32, 32	4	4	1	1	32	0,0,00	8,16,8,32	0,0,0,0	1,1,1,1	477	1078	1262	179
32, 32	4	0	1	1	32	0,0,00	32,8,16,32	1,1,1,1	2,2,2,2	276	667	466	190
128,32	4	4	1	1	32	0,0,00	8,16,8,32	1,0,1,0	2,1,2,1	748	1188	1511	167
32, 32	4	0	1	1	64	0,0,00	8,16,32,64	1,0,1,0	1,2,1,2	351	869	624	170
64, 64	4	0	1	1	64	0,0,00	8,16,32,64	0,0,0,0	1,2,1,2	375	840	733	175
64, 64	4	0	1	1	32	0,0,00	8,16,8,32	1,0,1,0	1,2,1,2	322	813	714	175
64, 64	4	0	1	1	64	0,0,00	8,16,32,664	1,1,1,1	1,2,1,2	325	971	767	168
64, 32	4	4	1	1	16	1,1,1,1	8,16,8,16	0,0,0,0	1,2,1,2	541	033	1363	170

Notes:

1. For all above cases, C_INCULDE_DATAWIDTH_MATCHING_x = 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
2. For all above cases, the rest of the parameters listed in Table 1 are assigned with the default values
3. All above cases for Virtex-5 are executed by providing MCH_SPLB_Clk = 6.66 ns (150 MHz) period constraint in UCF

The XPS MCH EMC resource utilization for various parameter combinations measured with the Virtex-6 (XC6VLX130-1-TFF1156) FPGA as the target device is detailed in Table 18.

Table 18: **Performance and Resource Utilization Benchmarks for the Virtex-6 (XC6VLX130-1-TFF1156) FPGA**

Parameter Values										Device Resources			F_{MAX} (MHz)
C_SPLB_DWIDTH, C_MCH_NATIVE_DWIDTH	C_NUM_BANKS_MEM	C_NUM_CHANNELS	C_INCLUDE_PLB_IPIF	C_INCLUDE_WRITE_BUFFER	C_MAX_MEM_WIDTH	C_PAGEMODE_FLASH_x	C_MEMx_WIDTH	C_SYNCH_MEM_x	C_SYNCH_PIPEDELAY_x	Slices	Slice Flip-Flops	LUTs	
32, 32	1	1	0	0	8	0,0,00	8	0	1	123	330	317	219
32, 32	1	4	1	1	8	0,0,00	8	1	2	200	499	547	190
32, 32	3	4	1	1	32	0,0,00	8,16,32	1,0,0	2,1,1	409	835	1072	167
32, 32	4	3	1	0	32	0,0,00	32,8,16,32	1,0,1,0	2,1,2,1	589	921	1270	190
32, 32	4	4	1	1	32	0,0,00	8,16,8,32	0,0,0,0	1,1,1,1	159	483	449	170
32, 32	4	0	1	1	32	0,0,00	32,8,16,32	1,1,1,1	2,2,2,2	296	632	799	166
128,32	4	4	1	1	32	0,0,00	8,16,8,32	1,0,1,0	2,1,2,1	497	968	1300	174
32, 32	4	0	1	1	64	0,0,00	8,16,32,64	1,0,1,0	1,2,1,2	642	1033	1434	167
64, 64	4	0	1	1	64	0,0,00	8,16,32,64	0,0,0,0	1,2,1,2	701	1107	1653	167
64, 32	4	4	1	1	16	1,1,1,1	8,16,8,16	0,0,0,0	1,2,1,2	241	651	639	168

Notes:

1. For all above cases, C_INCULDE_DATAWIDTH_MATCHING_x = 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
2. For all above cases, the rest of the parameters listed in Table 1 are assigned with the default values
3. All above cases for Virtex-6 are executed by providing MCH_SPLB_Clk =6 ns (166 MHz) period constraint in UCF

The XPS MCH EMC resource utilization for various parameter combinations measured with Spartan-3E (XC3S1600E-4-FG484) FPGA as the target device is detailed in Table 19.

Table 19: **Performance and Resource Utilization Benchmarks for the Spartan-3E (XC3S1600E-4-FG484) FPGA**

Parameter Values										Device Resources			F_{MAX} (MHz)
C_SPLB_DWIDTH, C_MCH_NATIVE_DWIDTH	C_NUM_BANKS_MEM	C_NUM_CHANNELS	C_INCLUDE_PLB_IPIF	C_INCLUDE_WRITE_BUFFER	C_MAX_MEM_WIDTH	C_PAGEMODE_FLASH_x	C_MEMx_WIDTH	C_SYNCH_MEM_x	C_SYNCH_PIPEDELAY_x	Slices	Slice Flip-Flops	LUTs	
32, 32	1	1	0	0	8	0,0,00	8	0	1	282	292	264	105
32, 32	1	4	1	1	8	0,0,00	8	1	2	1238	947	1642	102
32, 32	2	4	1	1	32	0,0,00	16,32	1,1	2,2	1296	1077	1757	104
32, 32	3	3	1	0	32	0,0,00	32,8,16	0,1,1	1,2,2	1009	959	1300	100
32, 32	3	4	1	1	32	0,0,00	8,16,32	1,0,0	2,1,1	1009	959	1300	100
32, 32	4	3	1	0	32	0,0,00	32,8,16,32	1,0,1,0	2,1,2,1	1223	1094	1725	100
32, 32	4	4	1	1	32	0,0,00	8,16,8,32	0,0,0,0	1,1,1,1	977	968	1329	101
128,32	4	4	1	1	32	0,0,00	8,16,8,32	1,0,1,0	2,1,2,1	1251	1122	1799	100
32, 32	4	0	1	1	64	0,0,00	8,16,32,64	1,0,1,0	1,2,1,2	623	831	748	100
64, 64	4	0	1	1	64	0,0,00	8,16,32,64	0,0,0,0	1,2,1,2	732	836	865	100
64, 64	4	0	1	1	32	0,0,00	8,16,8,32	1,0,1,0	1,2,1,2	806	812	868	100
64, 64	4	0	1	1	64	0,0,00	8,16,32,664	1,1,1,1	1,2,1,2	865	961	935	100
32, 32	4	0	1	1	16	1,1,1,1	8,16,8,16	0,0,0,0	1,2,1,2	489	555	506	101

Notes: Notes:

1. For all above cases, C_INCULDE_DATAWIDTH_MATCHING_x = 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
2. For all above cases, the rest of the parameters listed in Table 1 are assigned with the default values
3. All above cases for Spartan-3e are executed by providing MCH_SPLB_Clk = 10 ns (100 MHz) period constraint in UCF

The XPS MCH EMC resource utilization for various parameter combinations measured with Spartan-6 (XC6SLX6-2SG324) FPGA as the target device is detailed in Table 20.

Table 20: **Performance and Resource Utilization Benchmarks for the Spartan-6 (XC6SLX16-2-CSG324) FPGA**

Parameter Values										Device Resources			F_{MAX} (MHz)
C_SPLB_DWIDTH, C_MCH_NATIVE_DWIDTH	C_NUM_BANKS_MEM	C_NUM_CHANNELS	C_INCLUDE_PLB_IPIF	C_INCLUDE_WRITE_BUFFER	C_MAX_MEM_WIDTH	C_PAGEMODE_FLASH_x	C_MEMx_WIDTH	C_SYNCH_MEM_x	C_SYNCH_PIPEDELAY_x	Slices	Slice Flip-Flops	LUTs	
32, 32	1	1	0	0	8	0,0,00	8	0	1	118	330	305	141
32, 32	1	4	1	1	8	0,0,00	8	1	2	153	499	506	120
32, 32	2	4	1	1	32	0,0,00	16,32	1,1	2,2	372	836	998	114
32, 32	3	3	1	0	32	0,0,00	32,8,16	0,1,1	1,2,2	362	921	1115	116
32, 32	3	4	1	1	32	0,0,00	8,16,32	1,0,0	2,1,1	143	481	418	131
32, 32	4	3	1	0	32	0,0,00	32,8,16,32	1,0,1,0	2,1,2,1	199	635	717	111
32, 32	4	4	1	1	32	0,0,00	8,16,8,32	0,0,0,0	1,1,1,1	481	968	1233	111
128,32	4	4	1	1	32	0,0,00	8,16,8,32	1,0,1,0	2,1,2,1	549	1109	1531	112
32, 32	4	0	1	1	64	0,0,00	8,16,32,64	1,0,1,0	1,2,1,2	511	1032	1321	112
64, 64	4	0	1	1	64	0,0,00	8,16,32,64	1,1,1,1	1,2,1,2	257	649	664	111
32, 32	4	0	1	1	16	1,1,1,1	8,16,8,16	0,0,0,0	1,2,1,2	508	1141	1559	112

Notes:

1. For all above cases, C_INCULDE_DATAWIDTH_MATCHING_x = 1 when C_MEMx_WIDTH not equal to C_MCH_NATIVE_DWIDTH
2. For all above cases, the rest of the parameters listed in Table 1 are assigned with the default values
3. All above cases for Spartan-6 are executed by providing MCH_SPLB_Clk = 9ns (110 MHz) period constraint in UCF

System Performance

To measure the system performance (F_{MAX}) of this core, it was added to a Virtex-4 system, a Virtex-5 system, Spartan-3A system, a Spartan-6 system, and a Virtex-6 system as the Device Under Test (DUT) as shown in Figure 25, Figure 26, Figure 27, Figure 28 and Figure 29

Because the XPS MCH EMC core will be used with other design modules in the FPGA, the utilization and timing numbers reported in this section are estimates only. When the XPS MCH EMC core is combined with other designs in the system, the utilization of FPGA resources and timing of the XPS MCH EMC design will vary from the results reported here.

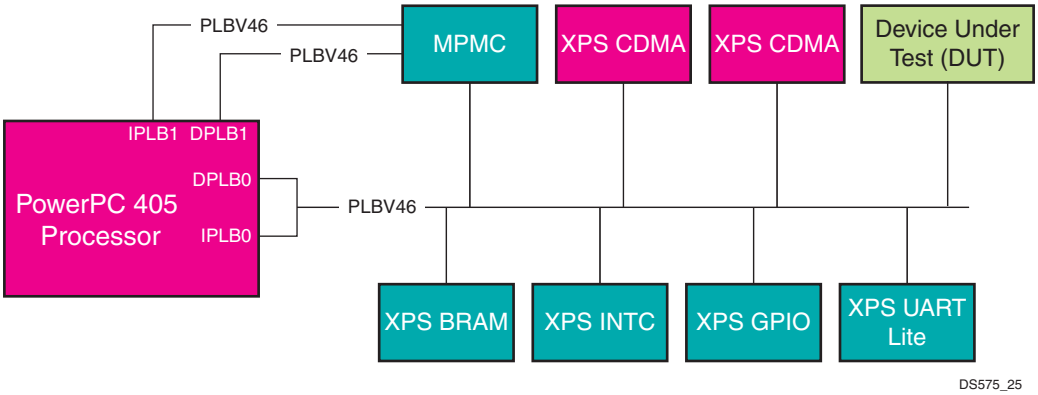

Figure 25: **Virtex-4 FX FPGA System with the XPS MCH EMC core as the DUT**

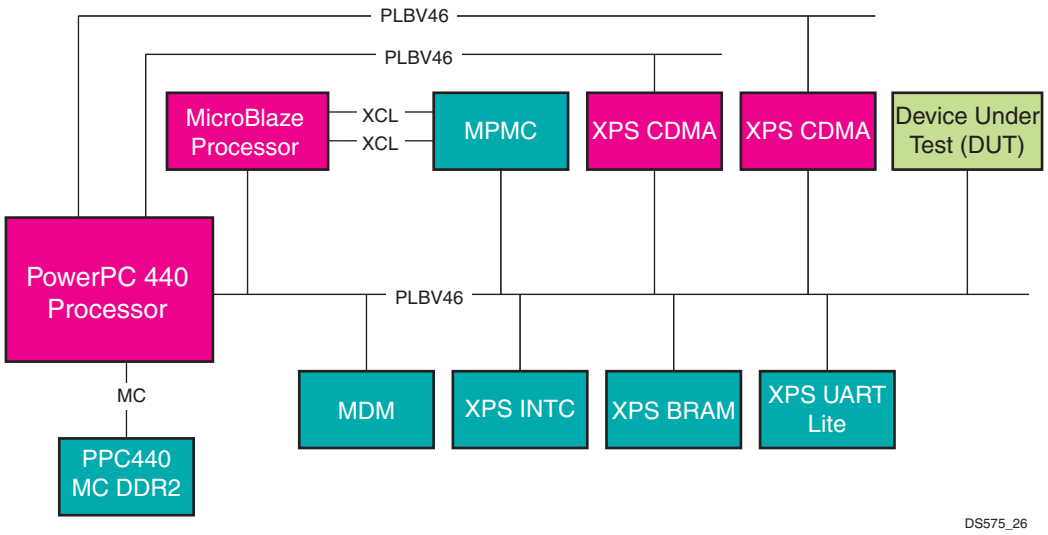

Figure 26: **Virtex-5 FXT FPGA System with the XPS MCH EMC core as the DUT**

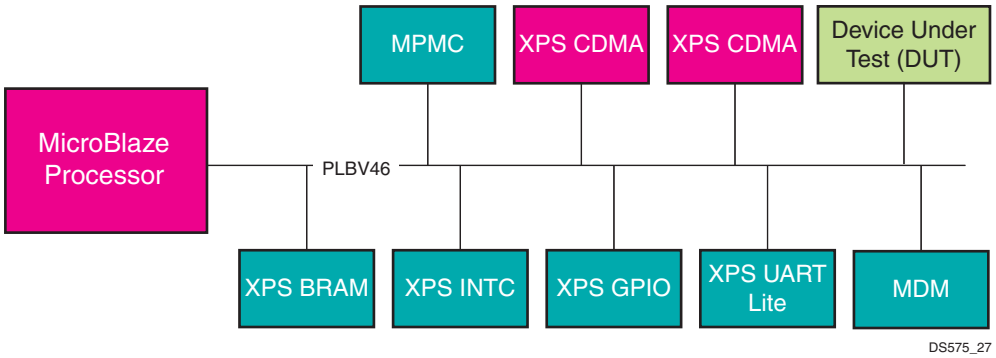

Figure 27: **Spartan-3A DSP FPGA System with the XPS MCH EMC core as the DUT**

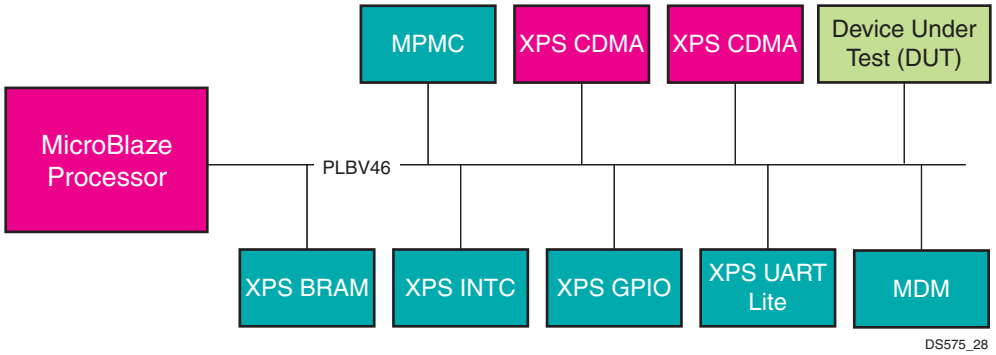

Figure 28: **Spartan-6 FPGA System with the XPS MCH EMC core as the DUT**

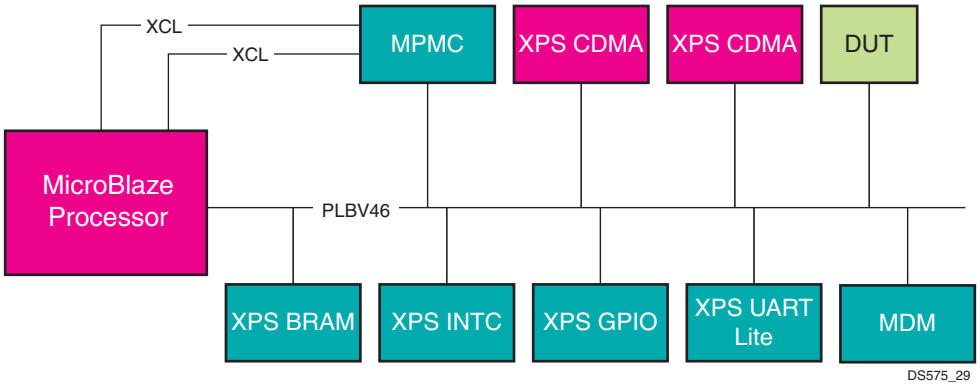

Figure 29: **Virtex-6 FPGA System with the XPS MCH EMC core as the DUT**

The target FPGA was then filled with logic to drive the LUT and BRAM utilization to approximately 70% and the I/O utilization to approximately 80%. Using the default tool options and the slowest speed grade for the target FPGA, the resulting target F_{MAX} numbers are shown in Table 21.

Table 21: XPS EMC System Performance

Target FPGA	Target F_{MAX} (MHz)
S3AD3400 -4	90
V4FX60 -10	100
V5FXT70 -1	120
S6LX16 -2	100
V6LX130t -1	150

The target F_{MAX} is influenced by the exact system and is provided for guidance. It is not a guaranteed value across all systems.

Support

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

Reference Documents

1. UG081 MicroBlaze Processor Reference Guide
2. DS626 MCH PLBV46 Slave Burst Design Specification
3. DS562 PLBV46 SLAVE BURST Design specification
4. IBM CoreConnect 128-Bit Processor Local Bus: Architecture Specification version 4.6

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
05/08/08	1.0	Initial release
06/10/08	1.1	Updated the timing diagrams for page mode flash, 64 bit IPIF Native DWidth, 8-bit, 16-Bit memories Added the constraints for 64 bit IPIF Native DWidth
4/24/09	1.2	Replaced references to supported device families and tool name(s) with hyperlink to PDF file.
4/28/09	1.3	Updated the document with Margin system and system performance table
7/10/09	1.4	Updated for resource utilization tables and updated the core version
4/19/10	1.5	Updated for resource utilization tables for Virtex-6 and Spartan-6 families; updated table 11 for address position
12/14/10	1.6	Incorporated CR565060; converted to current data sheet template.
12/01/10	1.7	Incorporated CR#584962; updated the default value of MCHx_ReadData_Control in Table 2 of XPS MCH EMC I/O Signal.s
6/22/11	1.8	Updated for 13.2 release.

Notice of Disclaimer