XILINX®

XAPP454 (v1.0) December 6, 2004

DDR2 SDRAM Memory Interface for Spartan-3 FPGAs

Author: Karthikeyan Palanisamy

Summary

This application note describes a DDR2 SDRAM memory interface implementation in a Spartan-3 device, interfacing with a Micron DDR2 SDRAM device. This document provides a brief overview of the DDR2 SDRAM device features, followed by a detailed explanation of the DDR2 SDRAM memory interface implementation.

DDR2 SDRAM Device Overview

DDR2 SDRAM devices are the next generation DDR SDRAM devices. The DDR2 SDRAM memory interface is source-synchronous and supports double-data rate like DDR SDRAM memory. DDR2 SDRAM devices use the SSTL 1.8V I/O standard.

DDR2 SDRAM devices use a DDR SDRAM architecture to achieve high-speed operation. The memory operates using a differential clock provided by the controller. (The reference design on the web does not support differential strobes. Support for this is planned to be added later.) Commands are registered at every positive edge of the clock. A bi-directional data strobe (DQS) is transmitted along with the data for use in data capture at the receiver. DQS is a strobe transmitted by the DDR2 SDRAM device during reads, and by the controller during writes. DQS is edge-aligned with data for reads, and center-aligned with data for writes.

Read and write accesses to the DDR2 SDRAM device are burst oriented. Accesses begin with the registration of an active command and are then followed by a read or a write command. The address bits registered coincident with the active command are used to select the bank and row to be accessed. The address bits registered with the read or write command are used to select the bank and starting column location for the burst access.

Interface Model

The DDR2 SDRAM memory interface is layered to simplify the design and make the design modular. Figure 1 shows the layered memory interface. The three layers consist of an application layer, an implementation layer, and a physical layer.

NOTICE OF DISCLAIMER: Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

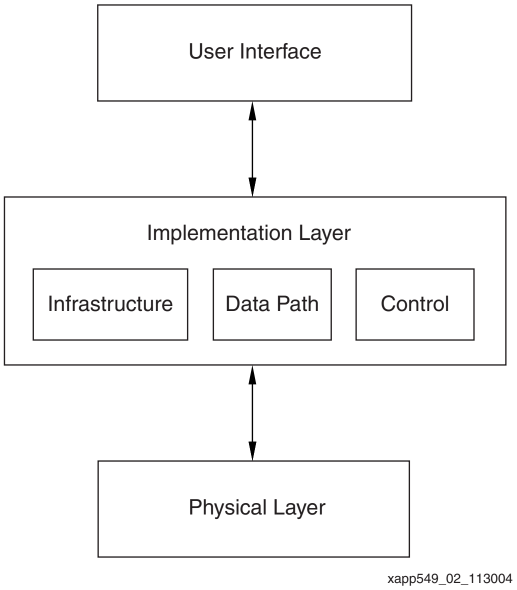

Figure 1: **Interface Layering Model**

DDR2 SDRAM Controller Modules

Figure 2 is a block diagram of the Spartan-3 DDR2 SDRAM memory interface. All four blocks shown in this figure are sub-blocks of the ddr2_top module. The function of each block is explained in the following sections.

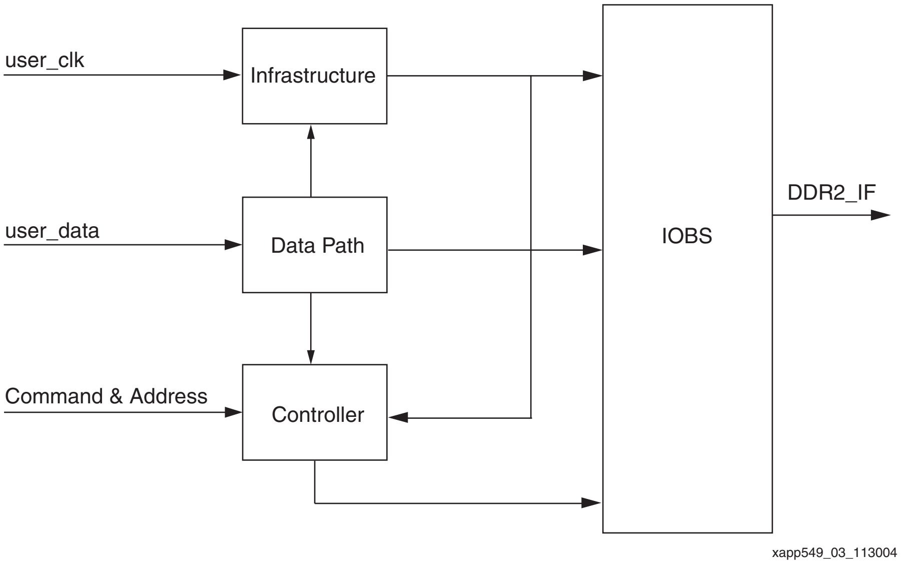

Figure 2: **DDR2 SDRAM Memory Interface Modules**

Controller

The controller's design is based on the design shown in XAPP253, *Synthesizable 400 Mb/s DDR SDRAM Controller*, but is modified to incorporate changes for the DDR2 SDRAM memory. It supports a burst length of four, and CAS latencies of three and four. The design is modified to implement the write latency feature of the DDR2 SDRAM memory. The controller initializes the EMR(2) and EMR(3) registers during the Load Mode command and also generates differential data strobes.

The controller accepts user commands, decodes these user commands, and generates read, write, and refresh commands to the DDR2 SDRAM memory. The controller also generates signals for other modules. Refer to XAPP253 for detailed design and timing analyses of the controller module.

Data Path

The data path module is responsible for transmitting data to and receiving data from the memories. Major functions include:

- Writing data to the memory
- Reading data from the memory
- Transferring the read data from the memory clock domain to the FPGA clock domain

For a description of data write and data read capture techniques, see XAPP768c, *Interfacing Spartan-3 Devices With 166 MHz or 333 Mb/s DDR SDRAM Memories*. The write data and strobe are clocked out of the FPGA. The strobe is center-aligned with respect to the data during writes. For DDR2 SDRAM memories, the strobe is non-free running. To meet the requirements specified above, the write data is clocked out using a clock that is shifted 90° and 270° from the primary clock going to the memory. The data strobes are generated out of primary clocks going to the memory.

Memory read data is edge-aligned with a source-synchronous clock. The DDR2 SDRAM clock is a non-free running strobe. The data is received using the non-free running strobe and transferred to the FPGA clock domain. The input side of the data uses resources similar to the input side of the strobe. This ensures matched delays on data and strobe signals until the strobe is delayed in the strobe delay circuit.

Infrastructure

The Infrastructure module generates the FPGA clocks and reset signals. A Digital Clock Manager (DCM) is used to generate the clock and its inverted version. A delay calibration circuit is also implemented in this module.

The delay calibration circuit is used to select the number of delay elements used to delay the strobe lines with respect to read data. The delay calibration circuit calculates the delay of a circuit that is identical in all respects to the strobe delay circuit. All aspects of the delay are considered for calibration, including all the component and route delays. The calibration circuit selects the number of delay elements for any given time. After the calibration is done, it asserts the select lines for the delay circuit. Refer to XAPP768c for details about delay calibration.

IOBS

All FPGA input and output signals are implemented in the IOBS module. All address and control signals are registered going into and coming out from the IOBS module.

User Interface Signals

Table 1 shows user interface signal descriptions; all signal directions are with respect to the DDR2 SDRAM controller.

Table 1: **User Interface Signals**

Signal Name	Direction	Description
dip1	Input	Clock enable signal for DDR2 SDRAM (active low)
rst_dqs_div_in	Input	This signal enables the dqs_div flop during DDR2 SDRAM memory read.
reset_in	Input	System reset
user_input_data[(2n-1):0]	Input	Write Data for DDR2 SDRAM, where 'n' is the width of the memory interface
user_input_address[addwidth:0]	Input	DDR2 SDRAM row and column address
user_bank_address[bankaddwidth:0]	Input	DDR2 SDRAM bank address
user_config_reg1[14:0]	Input	DDR2 SDRAM configuration data register1
user_config_reg2[12:0]	Input	DDR2 SDRAM configuration data register2
user_command_reg[3:0]	Input	User command register for DDR2 SDRAM controller
burst_done	Input	Burst data transfer done signal
rst_dqs_div_out	Output	This signal is externally connected to rst_dqs_div_in. This signal enables the dqs_div flop.
user_output_data[(2n-1):0]	Output	Read data from DDR2 SDRAM
user_data_valid	Output	This active Low signal indicates that read data from DDR2 SDRAM memory is valid.
user_cmd_ack	Output	Acknowledge signal for user_command
user_ODT_ack	Output	Acknowledge signal for ODT command
init_val	Output	Indicates DDR2 SDRAM is initialized
ar_done	Output	Indicates auto-refresh command is given to DDR2 SDRAM
clk_int	Input	Clock generated by DDR2 SDRAM controller
clk90_int	Input	90 degrees phase-shifted clock generated by DDR2 SDRAM controller
sys_rst	Input	This is generated with system reset input
sys_rst90	Input	90 degrees phase-shifted Reset generated with system reset input
sys_rst180	Input	180 degrees phase-shifted Reset generated with system reset input.
sys_rst270	Input	270 degrees phase-shifted Reset generated with system reset input.

Notes:

1. All signal directions are with respect to DDR2 SDRAM controller.

Signal Descriptions

user_input_data[(2n-1):0]

This is the write data to DDR2 SDRAM from the user interface. The data is valid on a DDR2 SDRAM write command, where n is the width of the DDR2 SDRAM memory. The DDR2 SDRAM controller converts single data rate to double data rate on the physical layer side.

user_input_address[addwidth:0]

This is the sum of row and column address for DDR2 SDRAM writes and reads. Depending on address width variable selection, user_input_address is divided into row and column address bits.

user_bank_address[bankaddwidth:0]

Bank address for DDR2 SDRAM. There is a variable through which the bank address is selectable.

user_config_reg1[14:0]

Configuration data for DDR2 SDRAM memory initialization. The contents of this register are loaded into the mode register during a Load Mode command. The format for user_config_reg1 is as follows:

14	13 11	10	9 7	6 4	3	2 0
PD	WR	TM	Res	Cas_latency	BT	Burst_length

Burst_length[2:0]

The controller supports only a burst length of four.

BT

This bit selects the burst type. The controller supports only sequential bursts. This bit is always set to zero in the controller.

Cas_latency [6:4]

Bits 6:4 select the cas latency. The DDR2 SDRAM controller supports a cas latency of 3 and 4.

Res [9:7]

Bits 9:7 are reserved for future implementation.

TM

This bit is loaded into the TM bit of the Load Mode Register.

WR [13:11]

These three bits are written to WR (write recovery) bits of the Load Mode register.

PD

This bit is written to PD (Power Down Mode) bit of the Load Mode register.

Refer to the Micron DDR2 SDRAM data sheets for details on the Load Mode register.

user_config_reg2[12:0]

DDR2 SDRAM configuration data for the Extended Mode Register. The format of user_config_reg2 is as follows.

12	11	10	9 7	6 4	3 2	1	0
OUT	RDQS	DQS	OCD	Posted CAS	RTT	ODS	Res

Refer to the Micron DDR2 SDRAM data sheets for details on the Extended Mode register.

user_command_reg[3:0]

This is the user command register. Various commands are passed to the DDR2 SDRAM module through this register. Table 2 illustrates the various supported commands.

Table 2: **User Commands**

user_command_reg[3:0]	User Command Description
0000	NOP
0010	Memory (DDR2 SDRAM) initialization
0011	Auto-refresh
0100	Write
0101	Load Mode (Only Load mode)
0110	Read
Others	Reserved

burst_done

Users should enable this signal, for two clock periods, at the end of the data transfer. The DDR2 SDRAM controller supports write burst or read burst for a single row. Users must terminate on a column boundary and reinitialize on a column boundary for the next row of transactions. The controller terminates a write burst or read burst by issuing a pre-charge command to DDR2 SDRAM memory.

user_output_data[(2n-1):0]

This is the read data from DDR2 SDRAM memory. The DDR2 SDRAM controller converts DDR SDRAM data from DDR2 SDRAM memory to SDR data. As the DDR SDRAM data is converted to SDR data, the width of this bus is 2n, where n is data width of DDR2 SDRAM memory.

user_data_valid

The user_output_data[(2n-1):0] signal is valid on assertion of this signal.

user_cmd_ack

This is the acknowledgement signal for a user read or write command. It is asserted by the DDR2 SDRAM controller during a read or write to DDR2 SDRAM. No new command should be given to the controller until this signal is deasserted.

init_val

The DDR2 SDRAM controller asserts this signal after completing DDR2 SDRAM initialization.

ar_done

The DDR2 SDRAM controller asserts this signal for one clock cycle after the auto-refresh command is given to DDR2 SDRAM.

Note: The output clock and reset signals can be used for data synchronization.

Table 3 shows memory interface signals.

Table 3: **Memory Interface Signals**

Signal Name	Direction	Description
ddr_dq[(datawidth –1):0]	Inout	Bidirectional DDR2 SDRAM memory data
ddr_dqs[(dqswidth-1):0]	Inout	Bidirectional DDR2 SDRAM memory data strobe signals. The number of strobe signals depends on the data width and strobe to data ratio.
ddr_cke	Output	Clock enable signal for DDR2 SDRAM memory
ddr_csb	Output	Active low memory chip select signal
ddr_rasb	Output	Active low memory row address strobe
ddr_casb	Output	Actie low memory column address strobe
ddr_web	Output	Active low memory write enable signal
ddr_dm	Output	Memory data mask signal
ddr_ba	Output	Memory bank address
ddr_address	Output	Memory address (both row and column address)
ddr2_clk*	Output	Memory differential clock signals
ddr_odt[4:0]	Output	Memory on-die termination signals.

Initializing DDR2 SDRAM Memory

Before issuing the memory read and write commands, the DDR2 SDRAM memory must be initialized using the memory initialization command. The data written in the Mode Register and in the Extended Mode Register should be placed on user_config_reg1 [14:0] and user_config_reg2 [12:0] until DDR2 SDRAM initialization is completed. Once the DDR2 SDRAM is initialized, the init_val signal is asserted by the DDR2 SDRAM controller. Figure 3 shows a timing diagram of the memory initialization command.

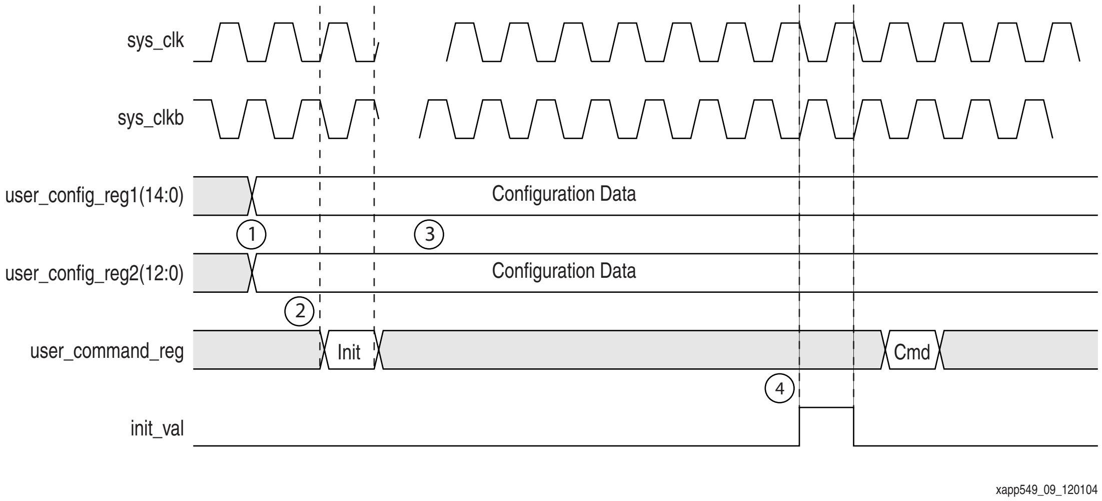

Figure 3: **DDR2 SDRAM Memory Initialization**

1. Two clocks prior to placing the initialization command on command_reg [2:0], the user places valid configuration data on user_config_reg1[14:0] and user_config_reg2[12:0].
2. The user places the initialization command on command_reg [2:0]. This starts the initialization sequence.
3. Data on user_config_reg1[14:0] and user_config_reg2[12:0] should not be changed for any subsequent memory operations.
4. The controller indicates that the configuration is complete by asserting the init_val signal.

DDR2 SDRAM Memory Write

Figure 4 shows a DDR2 SDRAM memory write timing diagram for a burst length of four. The waveform shows two successive bursts. Memory write is preceded by a write command to the DDR2 SDRAM controller. In response to the write command, the DDR2 SDRAM controller acknowledges with a user_cmd_ack signal on the rising edge of SYS_CLKb. Users should wait for a user command acknowledged signal before proceeding to the next step.

Two and a half clock cycles after the user_cmd_ack signal assertion, the memory burst address is placed on user_input_address[addwidth:0] lines. The user_input_address should be asserted on the rising edge of SYS_CLK. The data to be written into memory should be asserted with clk90_int_val and should be given to the controller before placing the memory address on user_input_address. The user data width is twice that of the memory data width. The controller converts it into double data rate before it is passed to memory.

For a burst length of four, two user_input_data[(2n-1):0] pieces of data are given to the DDR2 SDRAM controller with each user address. To terminate the write burst, burst_done is asserted on the rising edge of SYS_CLK for two clocks. The burst_done signal should be asserted for two clocks with the last memory address. Any further commands to the DDR2 SDRAM controller should be given only after the user_cmd_ack signal is deasserted.

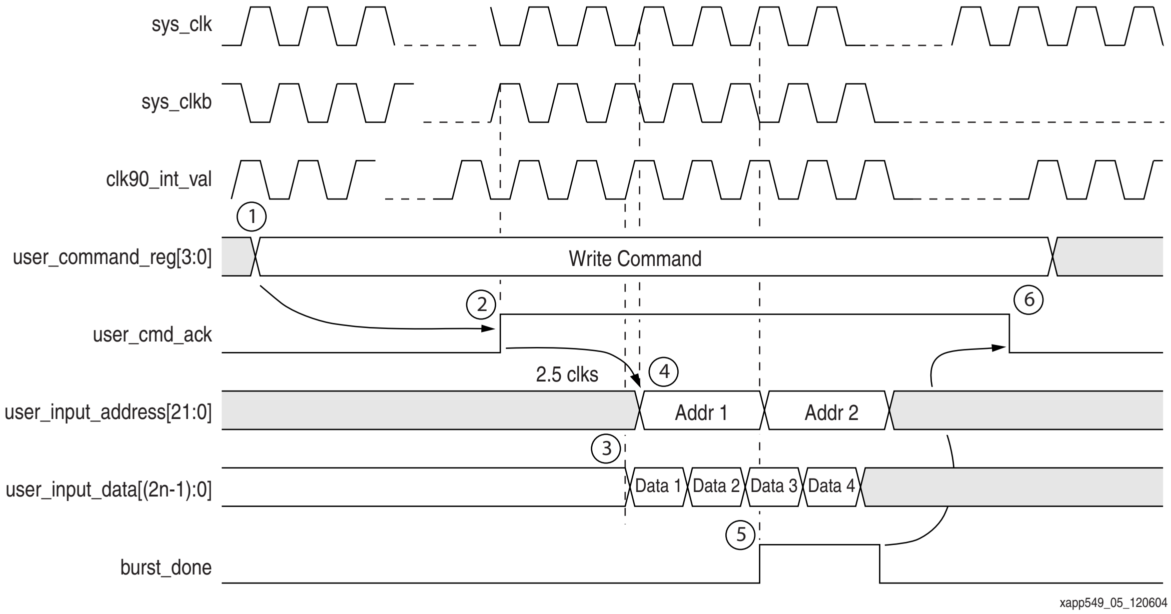

Figure 4: **DDR2 SDRAM Memory Write Burst Length of Four**

1. The user initiates a memory write by issuing a write command to the DDR2 SDRAM controller. The write command must be asserted on the rising edge of the SYS_CLK.
2. The DDR2 SDRAM controller acknowledges the write command by asserting the user_cmd_ack signal on the rising edge of the SYS_CLKb.
3. The user should place the data to be written into the memory onto the user_input_data pins before placing the memory address on the user_input_address. The input data is asserted with the clk90_int_val signal.
4. Two and half clocks after the user_cmd_ack signal assertion, the user should place the memory address on user_input address [21:0]. The user_input_address signal should be asserted on the rising edge of the SYS_CLK.
5. To terminate write burst, the user should assert the burst_done signal for two clocks with the last user_input_address.
6. Any further commands to the DDR2 SDRAM controller should be given only after the user_cmd_ack signal is de-asserted.

DDR2 SDRAM Memory Read

Figure 5 shows a memory read timing diagram for two successive bursts with a burst length of four. The user initiates a memory read by sending a read command to the DDR2 SDRAM controller.

The read command flow is similar to the write command. A read command is asserted on the rising edge of SYS_CLK. The DDR2 SDRAM controller asserts the user_cmd_ack signal in response to the read command on the rising edge of SYS_CLKb. After two and half clock cycles of user_cmd_ack, the memory burst read address is placed on user_input_address[addwidth:0]. The user_input_address signal is asserted on the rising edge of SYS_CLK.

The data read from the DDR2 SDRAM memory is available on user_output_data, which is asserted with clk90_int_val. The data on user_output_data is valid only when user_data_valid signal is asserted. As the DDR SDRAM data is converted to SDR data, the width of this bus is 2n, where n is the data width of the DDR2 SDRAM memory. For a read burst length of four, the DDR2 SDRAM controller outputs only two data with each user address, each of 2n width of DDR2 SDRAM memory. To terminate the read burst, a burst_done signal is asserted for two clock cycles on the rising edge of SYS_CLK. The burst_done signal should be asserted after the last memory address. Any further commands to the DDR2 SDRAM controller should be given after user_cmd_ack signal deassertion.

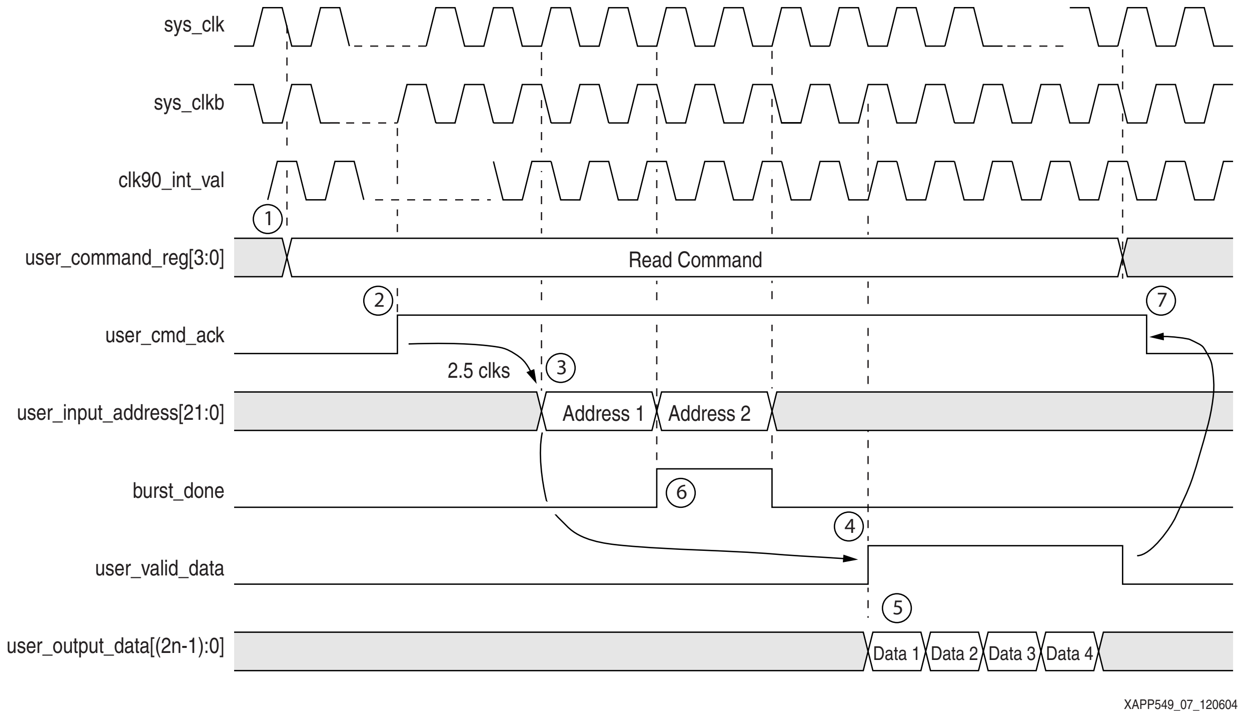

Figure 5: **DDR2 SDRAM Memory Read Burst Length of Four**

The read command flow is similar to the write command flow:

1. The user inputs the read command. It is accepted on the rising edge of the SYS_CLK.
2. The DDR2 SDRAM controller asserts the user_cmd_ack signal on the rising edge of the SYS_CLKb in response to the read command.
3. Two and half clocks after user_cmd_ack, the user places the memory read address on user_input_address [21:0]. The user_input_address signal is then accepted on the rising edge of SYS_CLK.
4. The data on user_output_data is valid only when the user_data_valid signal is asserted.

5. The data read from the DDR2 SDRAM memory is available on user_output_data. The user_output_data is asserted with clk90_int_val. Since the DDR SDRAM data is converted to SDR data, the width of this bus is 2n, where n is the data width of the DDR2 SDRAM memories. For a read burst length of four, with each user address the DDR2 SDRAM controller outputs only two data words.
6. To terminate the read burst, the burst_done signal is asserted for two clocks on the rising edge of SYS_CLK. The burst_done signal should be asserted with the last memory address.
7. Any further commands to the DDR2 SDRAM controller should be given after the user_cmd_ack signal is de-asserted.

DDR2 SDRAM Memory Auto_Refresh

The DDR2 SDRAM controller does not support memory refresh on its own and must periodically be provided with an auto_refresh command. The auto_refresh command is asserted with SYS_CLK. The ar_done signal is asserted by the DDR2 SDRAM controller upon completion of the auto_refresh command. The ar_done signal is asserted with SYS_CLKb.

Physical Layer and Delay Calibration

The physical layer for DDR2 SDRAM is similar to the DDR SDRAM physical layer described in application note XAPP768c. The delay calibration technique described in XAPP768c is also used in the DDR2 SDRAM interface.

Timing Calculations

Write Timing

Table 4: **Write Data**

Parameter	Value (ps)	Leading Edge Uncertainities	Trailing Edge Uncertainities	Meaning
Tclock	6000			Clock period
Tclock_phase	3000			Clock phase
Tdcd	250			Duty cycle distortion of clock to memory
Tdata_period	2750			Total data period, Tclock_phase-Tdcd
Tclock_skew	50	50	50	Minimal skew, since the right/left sides are used and the bits are close together
Tpackage_skew	90	90	90	Skew due to package pins and board layout (This can be reduced further with tighter layout.)
Tsetup	350	350	0	Setup time from memory data sheet
Thold	350	0	350	Hold time from memory data sheet
Tphase_offset_error	140	140	140	Offset error between different clocks from the same DCM
Tjitter	0	0	0	The same DCM is used to generate the clock and data; hence, they jitter together.
Total uncertainties	980	630	630	Worst case for leading and trailing can never happen simultaneously.
Window	1490	630	2120	Total worst-case window is 1490ps.

Read Timing

Table 5: **Read Data**

Parameter	Value (ps)	Leading Edge Uncertainities	Trailing Edge Uncertainities	Meaning
Tclock	6000			Clock period
Tclock_phase	3000			Clock phase
Tclock_duty_cycle_dist	300	0	0	Duty cycle distortion of clock to memory
Tdata_period	2700			Total data period, Tclock_phase-Tdcd
Tdqsq	350	350	0	Strobe to data distortion from memory data sheet
Tpackage_skew	90	90	90	Worst-case package skew
Tds	452	452	0	Setup time from Spartan-3 –5 data sheet
Tdh	-35	0	-35	Hold time from Spartan-3 –5 data sheet
Tjitter	100	0	0	Data and Strobe jitter together, since they are generated off of the same clock.
Tlocal_clock_line	20	20	20	Worst-case local clock line skew
Tpcb_layout_skew	50	50	50	Skew between data lines and strobes on the board
Tqhs	450	0	450	Hold skew factor for DQ from memory data sheet
Total uncertainties		962	575	Worst-case for leading and trailing can never happen simultaneously.
Window for DQS position for normal case	1163	962	2125	Worst-case window of 1163 ps.

Notes:

1. Reference for Tdqsq and Tqhs are from Micron data sheet for MT47H64M4FT-37E, Rev C, 05/04 EN.
2. Reference for Spartan-3 timing is –5 devices, Speeds file version 1.33.

Address and Command Timing

Table 6: **Address and Command Data**

Parameter	Value (ps)	Leading Edge Uncertainties	Trailing Edge Uncertainties	Meaning
Tclock	6000			Clock period
Tclock_skew	50	50	50	Minimal skew, since right/left sides are used and the bits are close together
Tpackage_skew	90	90	65	Using same bank reduces the package skew
Tsetup	500	500	0	Setup time from memory data sheet
Thold	500	0	500	Hold time from memory data sheet
Tphase_offset_error	140	140	140	Offset between different phases of the clock
Tduty_cycle_distortion	0	0	0	Duty cycle distortion does not apply
Tjitter	0	0	0	Since the clock and address are generated using the same clock, the same jitter exists in both; hence, it does not need to be included.
Total uncertainties		780	755	
Command window	3025	2220	5245	Worst-case window of 3025 ps

References

Xilinx Application Notes:

- XAPP253, "Synthesizable 400 Mb/s DDR SDRAM Controller"
- XAPP768c, "Interfacing Spartan-3 Devices With 166 MHz or 333 Mb/s DDR SDRAM Memories" (available under click license)

Xilinx Reference Designs:

- http://www.xilinx.com/bvdocs/appnotes/xapp253.zip
- http://www.xilinx.com/memory

Micron Data Sheet MT47H16M16FG-37E, available online at:
http://www.micron.com/products/dram/ddr2sdram/partlist.aspx?density=256Mb

Conclusion

It is possible to implement a high-performance DDR2 SDRAM memory interface for Spartan-3 FPGAs. This design has been simulated, synthesized (with Synplicity), and taken through the Xilinx Project Navigator flow.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
12/06/04	1.0	Initial Xilinx release.