XILINX®

XAPP851 (v1.1) July 14, 2006

DDR SDRAM Controller Using Virtex-5 FPGA Devices

Author: Toshihiko Moriyama and Rich Chiu

Summary

This application note describes a 200-MHz DDR SDRAM (JEDEC DDR400, PC3200 standard) controller implemented in a Virtex™-5 device. This implementation uses IDELAY elements to adjust read data timing. Read data timing calibration and adjustment is done in this controller.

DDR SDRAM devices are low-cost, high-density storage resources that are widely available from many memory vendors. This reference design has been developed using DDR400 SDRAM components.

DDR SDRAM Description

The DDR SDRAM specification details are available from JEDEC organization, part of the Electronic Industries Alliance (EIA), at http://www.jedec.org/. The DDR SDRAM specifications are published in the JEDEC document, under the reference JESD79E.

DDR SDRAM devices are the silicon memory resource most frequently used in systems today, with applications ranging from consumer products to video systems. DDR SDRAM device frequencies range to 200 MHz or DDR400. DRAM devices are available in component or module configurations.

DDR Controller Commands

Table 1 presents the commands issued by the controller. These commands are passed to the memory using the following control signals:

- Row Address Select ($\overline{RAS}$)
- Column Address Select ($\overline{CAS}$)
- Write Enable ($\overline{WE}$)
- Clock Enable (CKE) (always held High after device configuration)
- Chip Select ($\overline{CS}$) (always held Low during device operation)

Table 1: **DDR SDRAM Commands**

Signal No.	Function	$\overline{RAS}$	$\overline{CAS}$	$\overline{WE}$
1	Load Mode Register	L	L	L
2	Auto Refresh	L	L	H
3	Precharge(1)	L	H	L
4	Select Bank Activate Row	L	H	H
5	Write Command	H	L	L
6	Read Command	H	L	H
7	No Operation (NOP)	H	H	H

Notes:

1. Address signal A10 is held High during PRECHARGE ALL BANKS and is held Low during single bank precharge.

Command Functions

Mode Register

The Mode register is used to define the specific mode of DDR SDRAM operation, including the selection of burst length, burst type, CAS latency, and operating mode. Figure 1 shows the Mode register features that this controller uses.

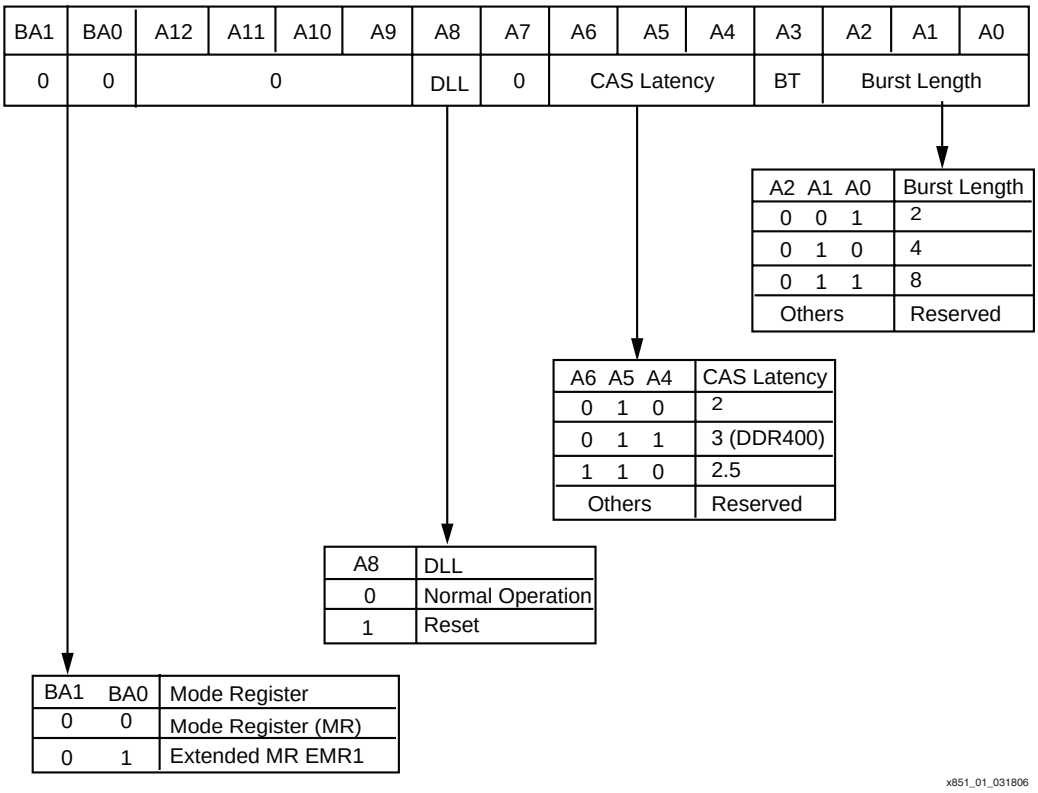

Figure 1: **Mode Register Definition for DDR400**

Bank Addresses BA1 and BA0 select the Mode registers. Figure 1 shows the Bank Address bits configuration.

Extended Mode Register

The Extended Mode register controls functions beyond those controlled by the Mode register. These additional functions are DLL enable/disable and output drive strength for DDR SDRAM interfaces shown in Figure 2.

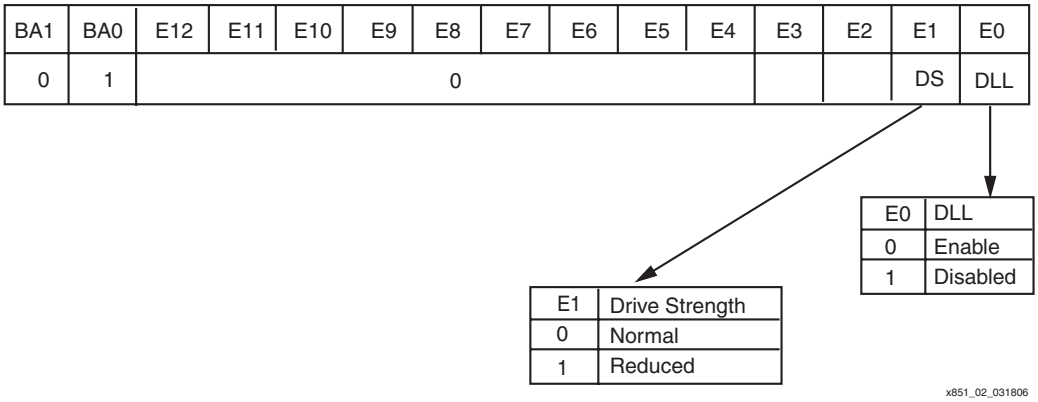

Figure 2: **Extended Mode Register for DDR400**

DDR SDRAM Memory Controller Reference Design

This DDR SDRAM Memory Controller Reference Design consists of a PHY layer and a Main controller layer as shown in Figure 3. The PHY layer consists of memory initialization logic, and address/command/data I/O logic. The read-data capture-timing calibration is also done within the PHY layer. The Main controller layer consists of the DDR SDRAM controller state machine and FIFO logic for address/command/data.

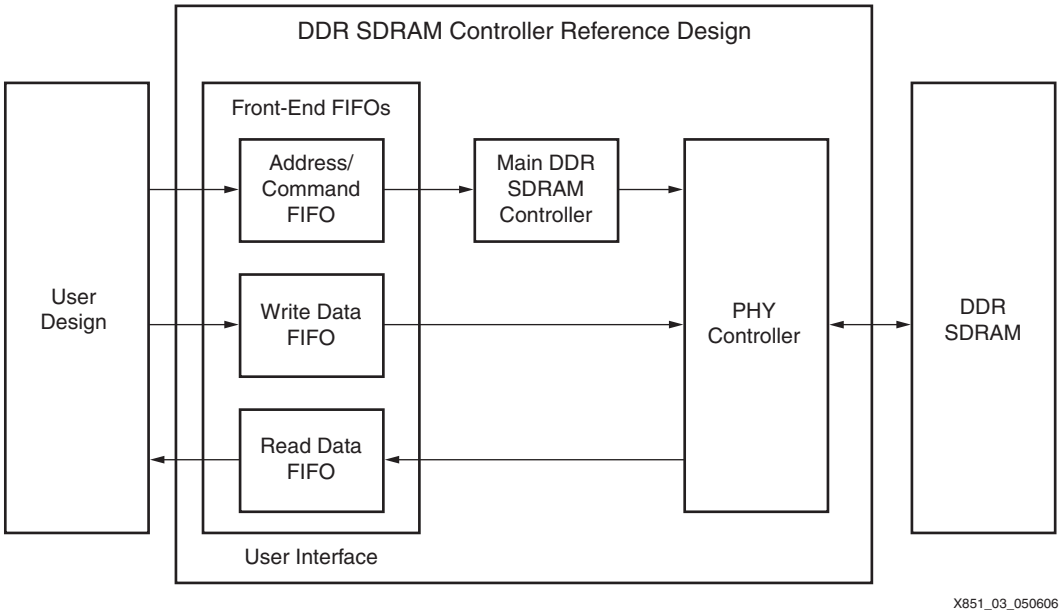

Figure 3: **Reference-Design DDR SDRAM Memory-Controller Structure**

DDR SDRAM Interface Design

The User Interface to the DDR controller provides a basic FIFO-like interface through which the user issues commands, provides write data to, and receives read data from the DDR memory. The data width of the of User Interface is twice that of the DDR memory bus, and provides the DDR Memory controller with two data words every FPGA clock cycle.

DDR SDRAM User Interface

The backend user interface has three FIFOs:

- Address/Command FIFO
- Write Data FIFO
- Read Data FIFO

The first two FIFOs are loaded by the user-specific backend logic, and the Read Data FIFO is accessed by the PHY Controller to store the captured data on each read cycle.

Table 2: **User Interface Port Descriptions**

Port Name	I/O	Width	Description	Notes
APP_ADDR	I	36	Instruction code and address for command for controller to execute. The bits in this port are mapped as follows: [31:0] Memory Address (CS, Bank, Row, Column) [34:32] Dynamic Command Request (see Table 4) [35] Unused – Reserve for future functionality.	Monitor APP_ADDR_AF almost full flag before writing to this FIFO
APP_ADDREN	I	1	Write strobe for APP_ADDR	Active-High
APP_ADDR_AF	O	1	Address/Command FIFO almost full flag	Active-High
APP_WR_DATA	I	data_width x 2	Write data for write burst	
APP_DATAMASK	I	data_mask_width x 2	Data mask corresponding to write data.	
APP_DATAEN	I	1	Write strobe for APP_WR_DATA/APP_DATAMASK	Active-High
APP_WRDATA_AF	O	1	Write Data FIFO almost full flag	Active-High
APP_RD_DATA	I	data_width x 2	Read Data FIFO output (captured read data).	
APP_RD_VALID	O	1	When asserted, indicates captured read data presented on APP_RD_DATA is valid on the current clock cycle.	Active-High
CTRL_RDY	O	1	When asserted, indicates the PHY Interface logic has finished SDRAM initialization and read datapath calibration	Active-High
PHY_ERROR	O	1	When asserted, indicates an error occurred during read datapath calibration	Active-High

The memory address (APP_ADDR) includes the column address, row address, bank address, and chip-select width for deep memory interfaces as shown in Table 3.

Caution! The memory controller does not support auto-precharge, and the user must always ensure that APP_ADDR[10] is Low for both read and write commands.

Table 3: **User Interface Address Bit Assignments**

Address	Bit Assignments
Column Address	col_ap_width – 1 : 0
Row Address	col_ap_width + row_address – 1 : col_ap_width

Table 3: **User Interface Address Bit Assignments *(Continued)***

Address	Bit Assignments
Bank Address	col_ap_width + row_address + bank_address – 1 : col_ap_width + row_address
Chip Select	col_ap_width + row_address + bank_address + chip_address – 1 : col_ap_width + row_address + bank_address

Dynamic Command Request

Table 4 lists the commands supported by the memory controller via the User Interface. Note that the Load Mode Register, Auto Refresh, Precharge, and Activate are automatically issued by the Memory Controller at the appropriate times. However, these commands can also be manually issued via the User Interface.

Table 4: **Controller Supported Commands**

APP_ADDR[34:32]	Description
000	Load Mode Register
001	Auto Refresh
010	Precharge All
011	Activate
100	Write
101	Read
110	NOP
111	NOP

DDR SDRAM Controller Interface

Figure 4 presents the state machine of the DDR SDRAM command generation state machine.

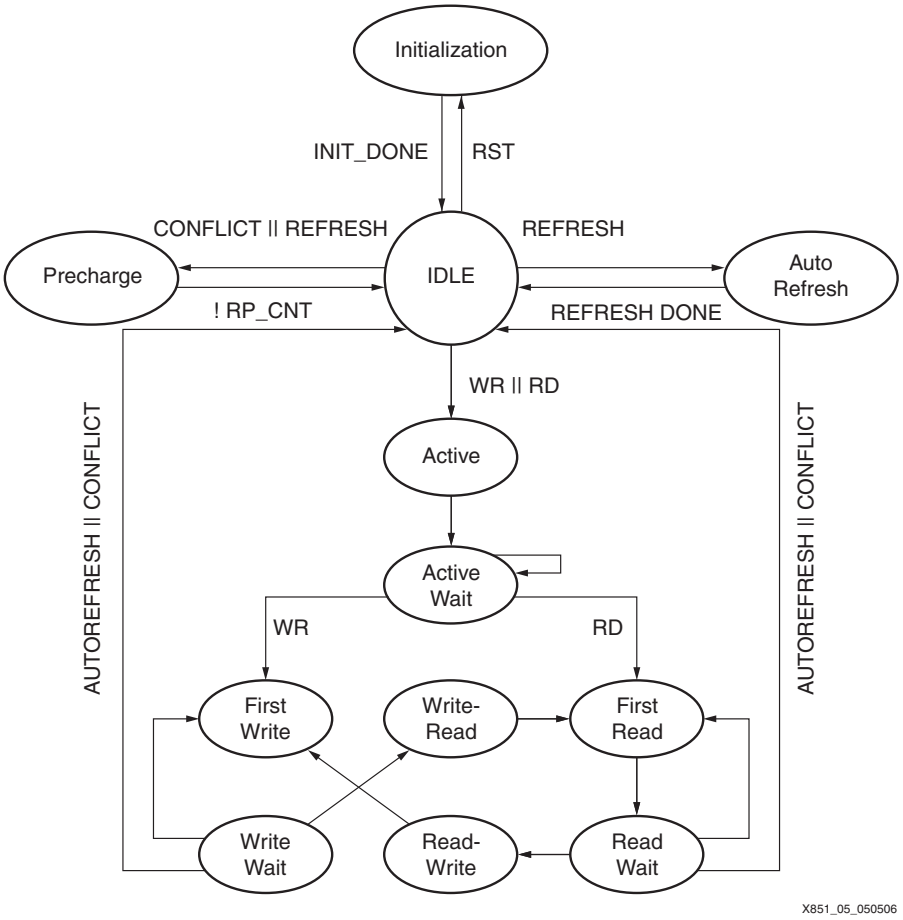

Figure 4: **Main Controller State Machine**

These steps are followed before the controller issues the commands to the memory:

1. The command logic block generates a Read/Write command.
2. The controller issues a Read-enable signal to the Read/Write address FIFO.
3. The controller activates a row in the corresponding bank if all banks have been precharged, or it compares the bank and row addresses to the already open row and bank address. If there is a conflict, the controller precharges the open bank, and then issues an active command before moving to the Read/Write states.
4. After entering the Write state, if the controller detects a Read command, the controller waits for the Write_to_Read time before issuing the Read command. Similarly, in the read state, when the controller detects a Write command from the command logic block, the controller waits for the Read_to_Write time before issuing the Write command.
5. The commands are pipelined to synchronize with the address signals before being issued to the DDR memory.

Table 5 lists the design files for the SDRAM Controller Interface.

Table 5: **DDR SDRAM Controller Design Files**

Module Name	File Name	Description
DDR1_TOP	ddr1_top.vhd	Top Module
DDR1_PARAMETERS	ddr1_parameters.vhd	DDR SDRAM memory parameters
DDR1_CONTROLLER	ddr1_controller.vhd	DDR SDRAM memory main controller.
DDR1_BACKEND_FIFOS	ddr1_backend_fifos.vhd	Instantiates ddr1_rd_wr_addr_fifo and ddr1_wr_data_fifo_16 modules
DDR1_RD_WR_ADDR_FIFO	ddr1_rd_wr_addr_fifo.vhd	Read/Write address FIFO
DDR1_WR_DATA_FIFO_16	ddr1_wr_data_fifo_16.vhd	Write Data FIFO

Table 6 lists the top-level I/O ports for the SDRAM Controller Interface.

Table 6: **DDR SDRAM Controller Top-Level Port Descriptions**

Port Name	I/O	Description
RST	I	See "PHY Interface," page 8 for signal descriptions
CLK0	I	
CLK90	I	
CKE	O	
CK	O	
AD	O	
BA	O	
CS_n	O	
RAS_n	O	
CAS_n	O	
WE_n	O	
DM	O	
DQ	I/O	
DQS	I/O	
APP_ADDR	I	See "DDR SDRAM User Interface," page 4 for signal descriptions
APP_ADDR_EN	I	
APP_WR_DATA	I	
APP_DATA_MASK	I	
APP_DATA_EN	I	
APP_RD_DATA	O	
APP_RD_VALID	O	
APP_ADDR_AF	O	
APP_WR_DATA_AF	O	
CTRL_RDY	O	
PHY_ERROR	O	

PHY Interface

The PHY layer contains the DDR SDRAM memory-initialization state machine and read data capture timing calibration logic. After power-on, DDR SDRAM memory initialization starts. After initialization is done, read data capture timing calibration starts.

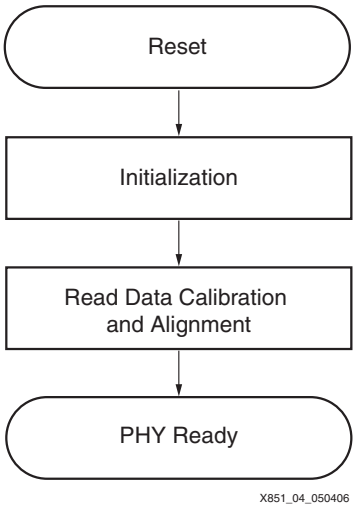

Figure 5: **PHY Initialization State-Machine Sequence**

Initialization

DDR SDRAM must be initialized before read and write operation. As shown in Figure 6, when the active-High Reset signal is set from High-to-Low, the controller starts memory initialization. The memory initialization sequence is defined by a JEDEC specification.

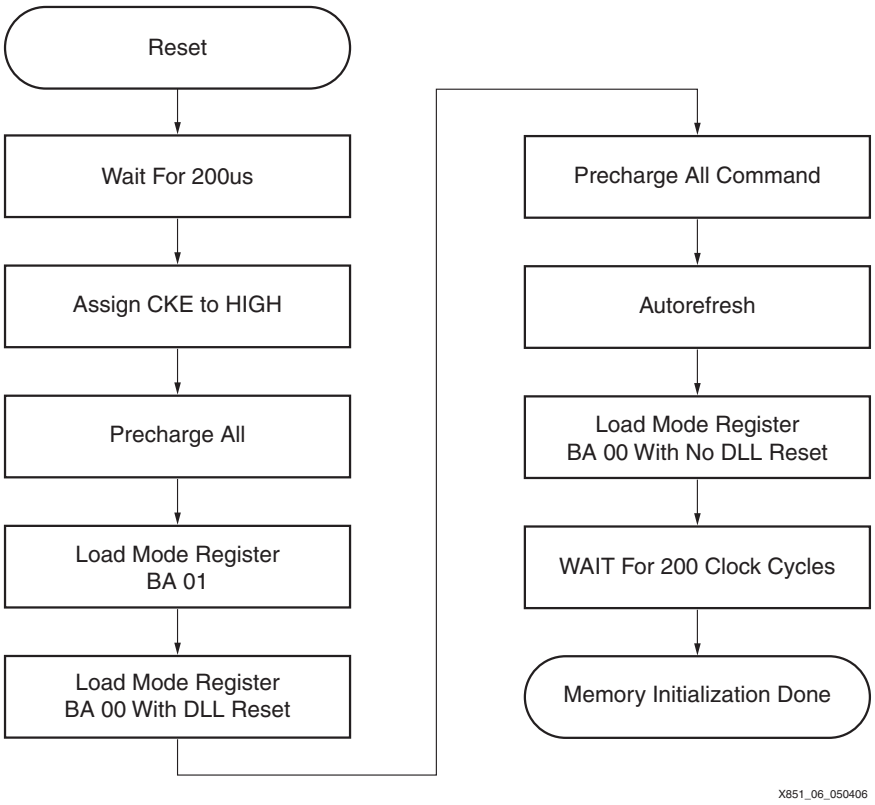

Figure 6: **Memory Initialization State-Machine Sequence**

Read Data Capture Timing Calibration

After calibration, Read data is captured with the DQS strobe signal. The read data must then be transferred from the DQS clock domain to the FPGA clock (CLK0) domain. However, DQS does not have a pre-determined relationship with the FPGA clock. To do this transfer, the DQ/DQS must be phase-shifted to allow the FPGA clock to capture the DQ data without timing violations. As shown in Figure 7, the DQ data is captured/synchronized by the DQS signal with the IDDR register.

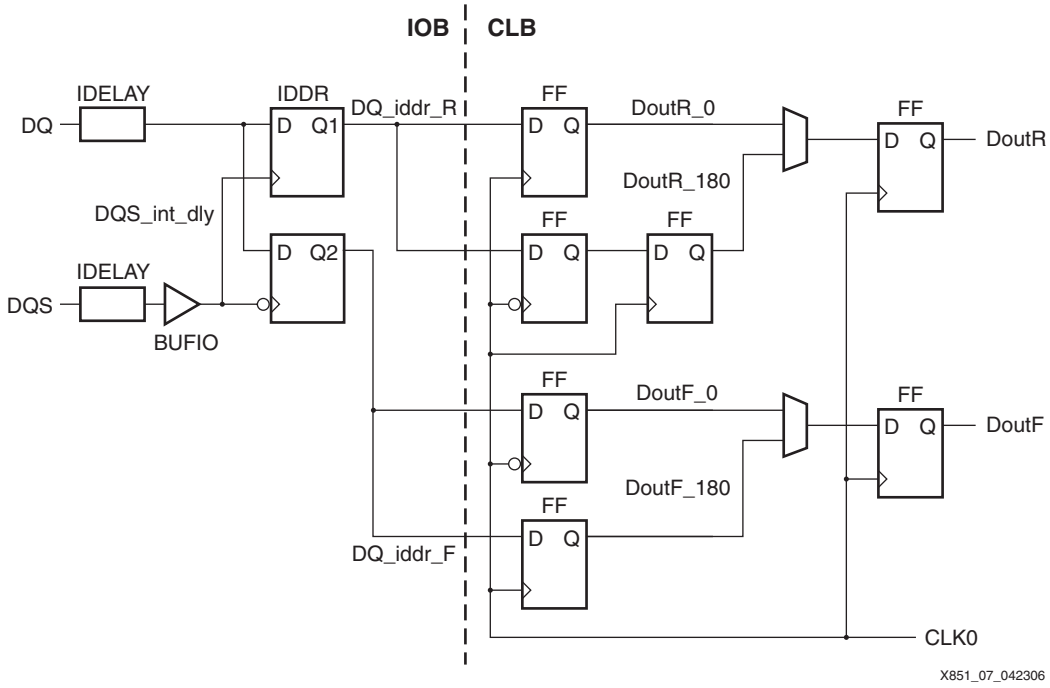

Figure 7: **Read Data Capture Block**

The DQS signal is routed through a delay circuit and the BUFIO and provides the IDDR clock input. The DQ_iddr_R is the output of IDDR captured by the DQS rising edge. The DQ_iddr_F is the output signal of IDDR captured by the DQS falling edge. These DQ_iddr_R and DQ_iddr_F signals are not phase-aligned with CLK0. The calibration logic in this reference design delays DQ and DQS signals to synchronize with the CLK0 clock. There are four possible cases for this alignment.

Case 1. CLK0 is within 90° and 180° position of DQS. In this case, add 0° to 90° delays to DQ and DQS by using IDELAY.

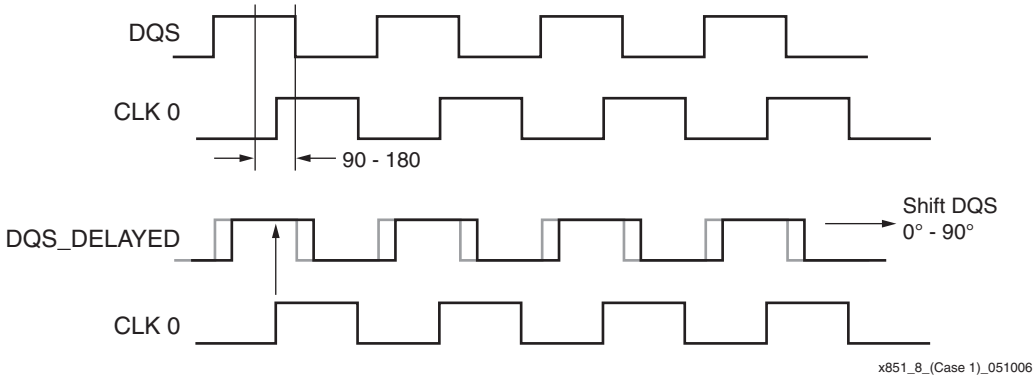

Figure 8: **Case 1 – DQS and System Clock Phase Relationship**

Case 2. CLK0 is within 180° to 270° position of DQS. In this case, add 90° to 180° delays to DQ and DQS by using IDELAY.

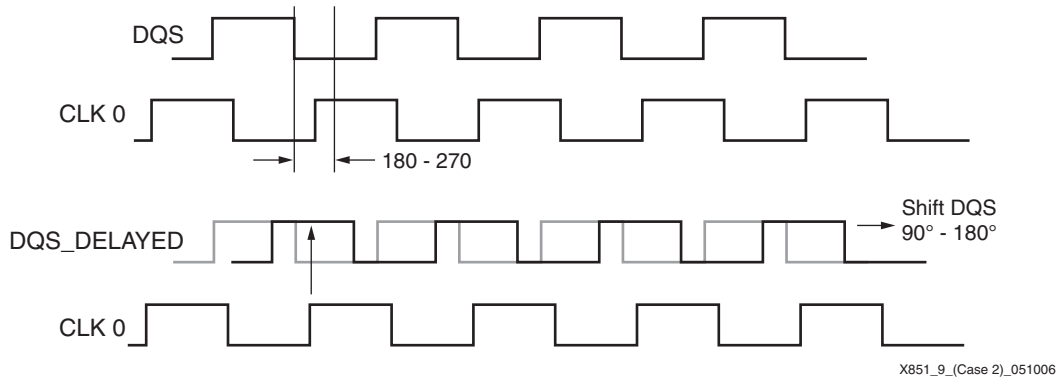

Figure 9: **Case 2 – DQS and System Clock Phase Relationship**

Case 3. CLK0 is with 270° and 360° position of DQS. In this case, add 0° to 90° delays to DQ and DQS by using IDELAY, and use the opposite edge of CLK0 to capture DQ.

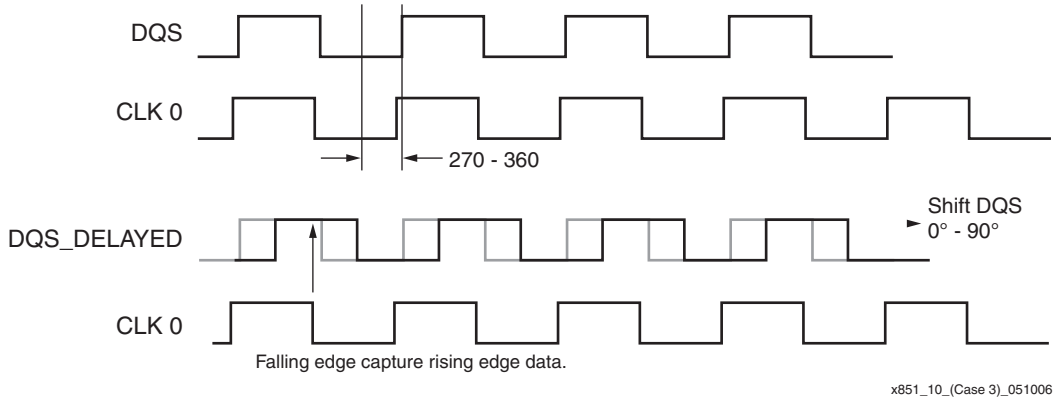

Figure 10: **Case 3 – DQS and System Clock Phase Relationship**

Case 4. clk0 is within 0° and 90° position of DQS. In this case, add 90° to 180° delays to DQ/DQS by using IDELAY and use opposite edge of clk0 to capture DQ.

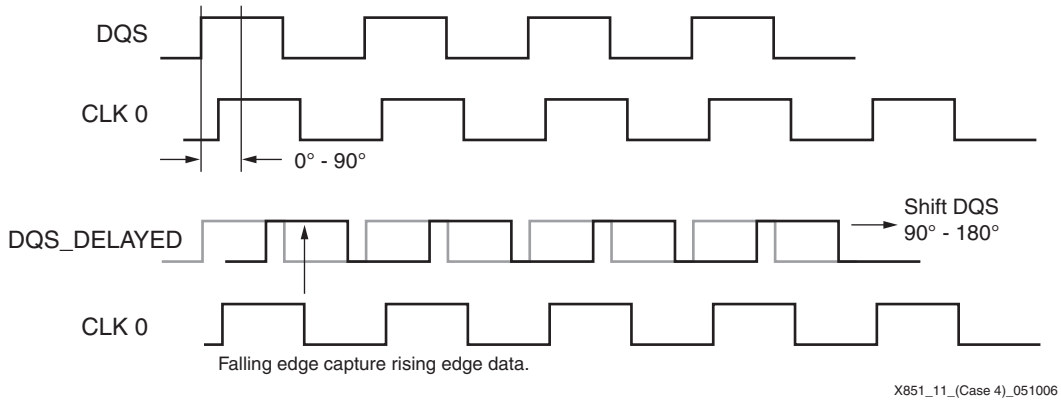

Figure 11: **Case 4 – DQS and System Clock Phase Relationship**

Read Enable Timing Calibration

The amount of delay between the FPGA and memory is dependent on various environmental factors (e.g., customer board layout and PCB trace lengths). Therefore, the controller does not know exactly on which FPGA clock cycle the valid data will arrive at the FPGA when it issues a read command to the memory. Because the DDR SDRAM device does not provide a read valid or read-enable signal along with the read data, it is necessary to perform calibration to determine on which FPGA clock cycle the read data is valid. This read-enable signal is based on the CAS latency and burst length, and compensates for customer-specific delays between the memory and FPGA. The number of register stages required to align the read-enable signal to the Read Data Capture Block output is determined during calibration. One internal read-enable signal is generated for each data byte. Figure 12 shows the read enable logic block diagram. This reference design includes logic that can adjust the read-enable timing by doing training during the initialization phase.

a. The controller writes a fixed data pattern to memory. This serves as a “training” pattern during read enable calibration.
b. Data is read back from memory and the read data is compared to the original training pattern.
c. The Read Enable signal is delayed until the received data output from the Read Data Capture Block matches the training pattern.
d. Since it is possible that different bytes may produce different read enable latencies, it may be necessary to delay the read data output from the Read Data Capture block for certain bytes such that the entire read word arrives at the internal Read Data FIFO on the same FPGA clock cycle.

After Read Data Capture and Read Enable calibration is complete, the PHY controller is ready to take user commands from the Main controller.

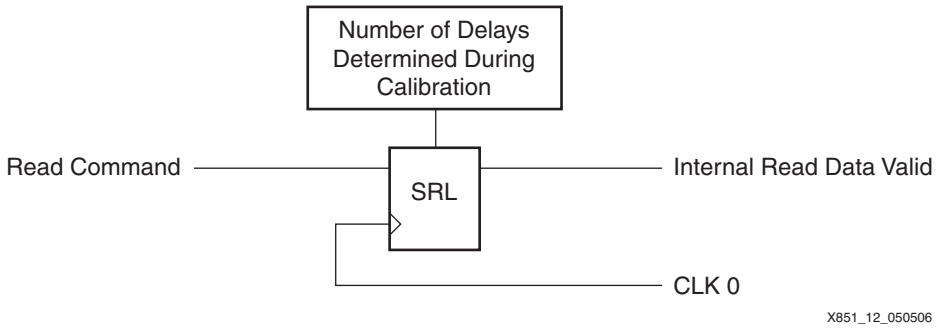

Figure 12: **Read Enable**

Timing Analysis

The read data DQ is captured by DQS and transferred to the FPGA clock domain as shown in “Read Data Capture Timing Calibration,” page 9. The Read Data and Clock timing relationships are shown in Table 7.

Table 7: **Read Data Timing Analysis**

Parameters	Symbol	Time (ps)
Clock period	t_{CK}	5000
DDR SDRAM Memory		
Data period (duty cycle 0.45 : 0.55)	$t_{CKx0.45}$	2250
Access window of DQS from CK/$\overline{CK}$ total	t_{DQSCK}	1200
DRAM uncertainty total		1200
FPGA		
BUFIO clock tree skew		TBD
System clock jitter	$t_{PERJITT\_0}$	TBD
IDDR out to CLB FF skew		TBD
Tap uncertainty (±1 IDELAY tap count)	$t_{IDELAYRESOLUTION}$	TBD
FPGA uncertainty total		TBD
Uncertainties total		TBD
DQ window margin		TBD

PHY Code Structure

Since the PHY layer is separate from the Main controller, the PHY layer can be used independently. When the PHY layer of the DDR controller design is used independently, the PHY layer structure as shown in Table 8 and Figure 13 must be included in an independent controller. In this case, functions like open/close row management, memory refresh, and read and write access timing must be managed by the independent controller.

Table 8: **PHY Design Files**

Module Name	File Name	Description
PHY_TOP	phy_top.vhd	PHY interface top
PHY_ADR_OUT	phy_adr_out.vhd	Address and bank signals IOB FF
PHY_CTRL_OUT	phy_ctrl_out.vhd	Control signals IOB FF
PHY_DATA_WRITE	phy_data_write.vhd	Write data path
PHY_DATA_READ	phy_data_read.vhd	Read data path
PHY_DQ_ALIGN	phy_dq_align.vhd	Read data-capture timing-alignment logic
PHY_RDEN_ALIGN	phy_rden_align.vhd	Read enable alignment signal
PHY_PTN_GEN	phy_ptn_gen.vhd	Pattern generator for Read-capture timing calibration
PHY_INIT	phy_init.vhd	DDR SDRAM memory initialization state machine

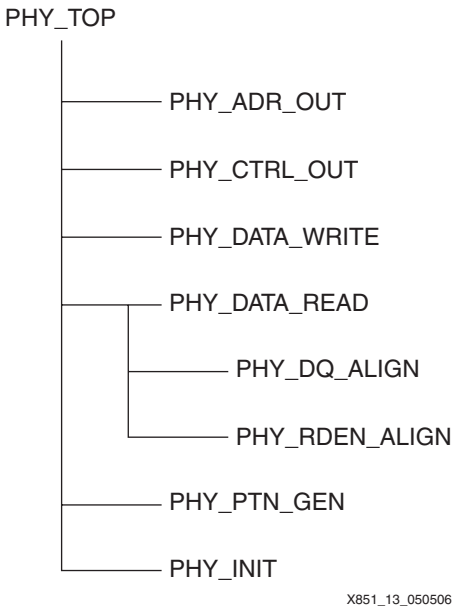

Figure 13: **PHY Layer Code Structure**

The PHY Layer contains all of the controls for the I/O ports used to communicate with the DDR SDRAM. The list of these ports and their descriptions are shown in Table 9.

Table 9: **PHY Layer I/O Port and Signal Descriptions**

Port Name	I/O	Description
RST	I	Synchronous Reset
CLK0	I	Main Clock (BUFG clock)
CLK90	I	90-degree phase shifted clock (BUFG clock)
PHY_ADDR_IN	I	Row address / Column address IOB FF
PHY_BANK_IN	I	Bank select
PHY_CS_N_IN	I	CS_N signal definition is same as memory signal
PHY_RAS_N_IN	I	RAS_N signal definition is same as memory signal
PHY_CAS_N_IN	I	CAS_N signal definition is same as memory signal
PHY_WE_N_IN	I	WE_N signal definition is same as memory signal
PHY_WR_DATA_IN	I	Write data
PHY_WR_EN_IN	I	Write data is enabled when this signal is High
PHY_WR_DM_IN	I	Data Mask bit
PHY_RD_DATA_O	O	Read data
PHY_RD_VALID_O	O	Read data valid
CKE	O	Connect to CKE pin of memory
CK	O	Connect to CK pin of memory
AD	O	Connect to AD pin of memory
BA	O	Connect to BA pin of memory
CS_N	O	Connect to CS_n pin of memory
RAS_N	O	Connect to RAS_n pin of memory

Table 9: **PHY Layer I/O Port and Signal Descriptions *(Continued)***

Port Name	I/O	Description
CAS_N	O	Connect to CAS_n pin of memory
WE_N	O	Connect to WE_n pin of memory
DM	O	Connect to DM pin of memory
DQ	I/O	Connect to DQ pin of memory
DQS	I/O	Connect to DQS pin of memory

PHY User Interface

After the PHY has completed initialization and calibration, the controller layer can issue commands. Some of the available commands are shown in the following sections:

- "General Command Timing"
- "Data Write"
- "Data Read"

General Command Timing

Timing for DDR SDRAM commands, such as for Refresh and Activate is shown in Figure 14. Refer to Table 1 on page 1 for signal logic levels for the various DDR SDRAM commands.

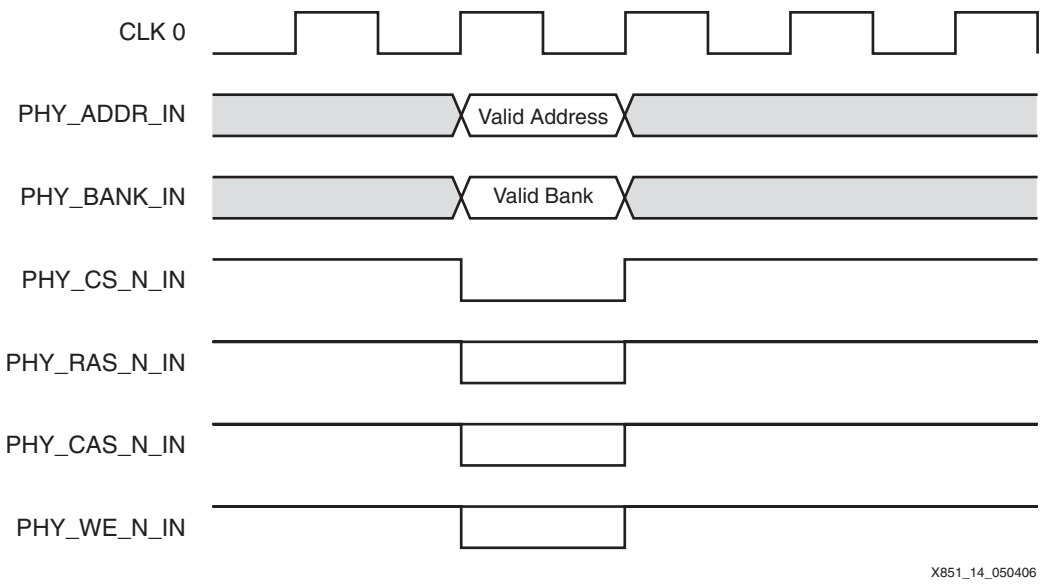

Figure 14: **DDR SDRAM Access Timing**

Data Write

When issuing a write command to the PHY interface, the controller layer sends valid address, bank, control signals, and valid data to the PHY. These signals should be issued on the same clock cycle. When the burst length is 4 or 8, input write data in consecutive clock cycles while asserting PHY_WR_EN_IN.

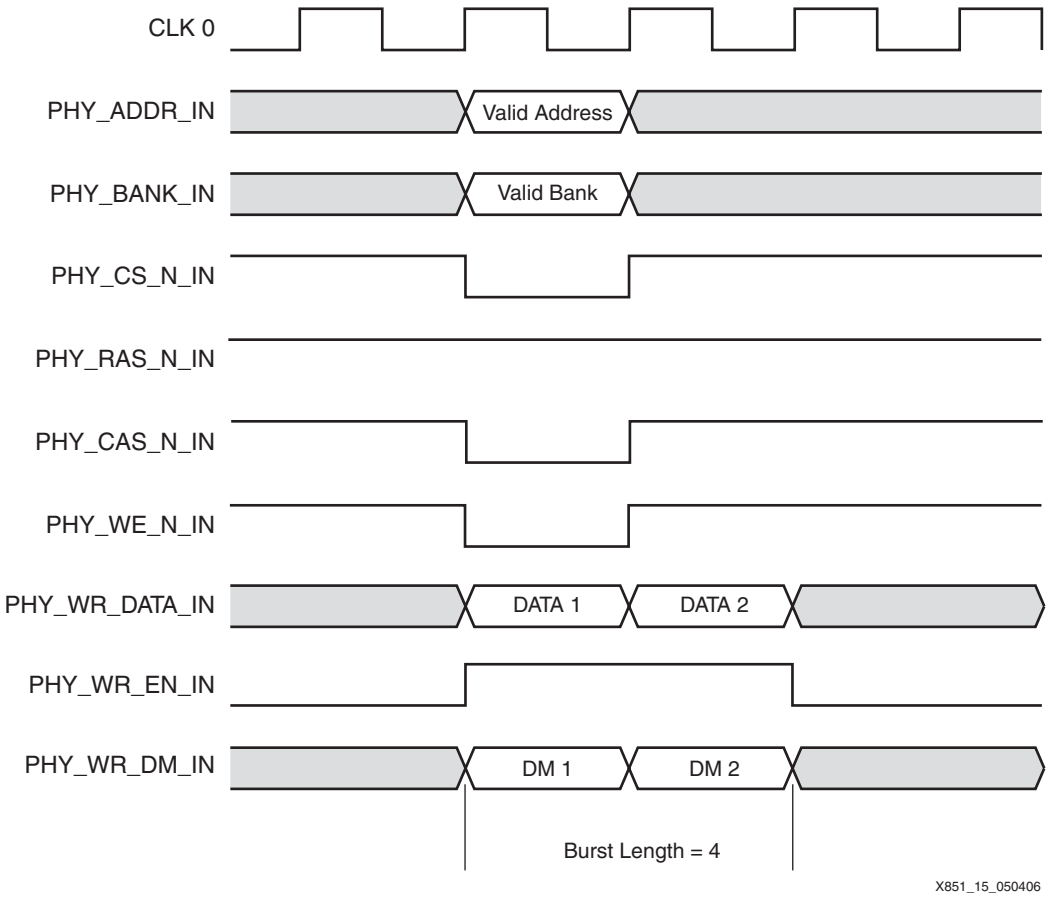

Figure 15: **Write Command Timing (Burst Length = 4)**

Data Read

After issuing READ command to the PHY layer, the PHY layer returns read data from memory. The read data is valid on the PHY_RD_DATA_O port only when PHY_RD_VALID_O is asserted during the same clock cycle. The latency from read command varies based on the results of read enable calibration.

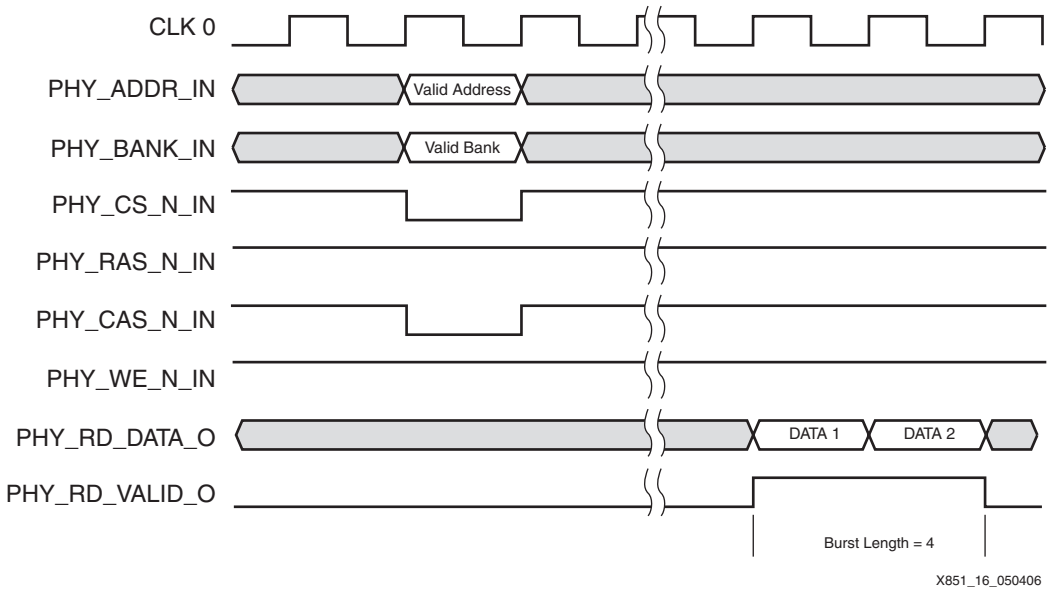

Figure 16: **Read Command Timing (Burst Length = 4)**

Reference Design Specification

The reference design for implementing at 200-MHz DDR SDRAM controller is available at:

http://www.xilinx.com/bvdocs/appnotes/xapp851.zip

Table 10 lists the specifications for this reference design.

Table 10: **Reference Design Utilization**

Parameter	Specifications/Details
Frequency of operation	200-MHz (DDR400 – PC3200)
Supported CAS latency	2, 2.5, and 3
HDL language	VHDL
Bus width	16-bit
Device used for verification for components	Micron MT46V32M16FN-5

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
05/12/06	1.0	Initial Xilinx release.
07/14/06	1.1	Added link to reference design file. Added APP_DATAEN to Table 2. Rewrote introduction in "Read Data Capture Timing Calibration."