High-Performance DDR3 SDRAM Interface in Virtex-5 Devices

Author: Matt Dipaolo

XAPP867 (v1.0) September 24, 2007

Summary

This application note describes the controller and the data capture technique for high-performance DDR3 SDRAM interfaces. This data capture technique uses the Input Double Data Rate (IDDR) and Output Double Data Rate (ODDR) features available in every Virtex™-5 FPGA I/O.

Introduction

A DDR3 SDRAM interface is source-synchronous, where the read data and read strobe are transmitted edge aligned. To capture this transmitted data using Virtex-5 FPGAs, either the strobe or the data can be delayed. In this design, the read data is captured in the delayed strobe domain and recaptured in the FPGA clock domain with the IDDR and the registers in the FPGA fabric. The differential strobe is placed on a clock-capable I/O pair to access the BUFIO clock resource. The BUFIO clocking resource routes the delayed read DQS to its associated data IDDR clock inputs. The write data and strobe transmitted by the FPGA use the ODDR.

A brief overview of the DDR3 SDRAM device features and a detailed explanation of the controller operation when interfacing to high-speed DDR3 memories are provided. The backend user interface to the controller is also explained.

DDR3 SDRAM Overview

DDR3 SDRAM devices are the next generation devices in the DDR SDRAM family. DDR3 SDRAM devices use 1.5V signaling. The following section explains the features available in the DDR3 SDRAM devices and the key differences between DDR2 SDRAM and DDR3 SDRAM controllers.

DDR3 SDRAM devices use a DDR architecture to achieve high-speed operation. The memory operates using a differential clock provided by the controller. Commands are registered at every positive edge of the clock. A bidirectional data strobe (DQS) is transmitted along with the data for use in data capture at the receiver. DQS is a strobe transmitted by the DDR3 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 DDR3 SDRAM device are burst oriented. Accesses begin with the registration of an Active command, which is then followed by a Read or Write command. The address bits registered 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 the starting column location for the burst access.

The DDR3 controller design (based on the DDR2 controller and modified for different initialization and mode registers) includes a user backend interface to generate the Write address, Write data, and Read addresses. This information is stored in three backend FIFOs for address and data synchronization between the backend and controller modules. Based on the availability of addresses in the address FIFO, the controller issues the correct commands to the memory, taking into account the timing requirements of the memory. The implementation details of the logic blocks are explained in the following sections.

DDR3 SDRAM Commands Issued by the Controller

Table 1 explains the commands issued by the controller. The commands are detected by the memory using the following control signals: Row Address Select ($\overline{RAS}$), Column Address Select ($\overline{CAS}$), and Write Enable ($\overline{WE}$) signals. Clock Enable (CKE) is held High after device

configuration, and Chip Select ($\overline{CS}$) is held Low throughout device operation. The Mode Register Definition section describes the DDR3 command functions supported in the controller.

Table 1: **DDR3 Commands**

Step	Function	$\overline{RAS}$	$\overline{CAS}$	$\overline{WE}$
1	Load Mode	L	L	L
2	Auto Refresh	L	L	H
3	Precharge [1]	L	H	L
4	Bank Activate	L	H	H
5	Write	H	L	L
6	Read	H	L	H
7	No Operation/IDLE	H	H	H

Notes:

1. Address signal A10 is held High during *Precharge All Banks* and is held Low during single bank precharge.

Mode Register Definition

The Mode register is used to define the specific mode of operation of the DDR3 SDRAM. This includes the selection of burst length, burst type, CAS latency, and operating mode. Figure 1 shows the Mode register features used by this controller. Bank Addresses BA1 and BA0 select the Mode registers.

BA2	BA1	BA0	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	PD		WR		DLL	0		CAS# Latency		0		Burst Length	

A2	A1	A0	Burst Length
0	1	0	4
0	1	1	8
Others			Reserved

A6	A5	A4	CAS Latency
0	0	1	5
0	1	0	6
0	1	1	7
1	0	0	8
1	0	1	9
1	1	0	10
Others			Reserved

A11	A10	A9	Write Recovery
0	0	1	5
0	1	0	6
0	1	1	7
1	0	0	8
1	0	1	10
1	1	0	12
Others			Reserved
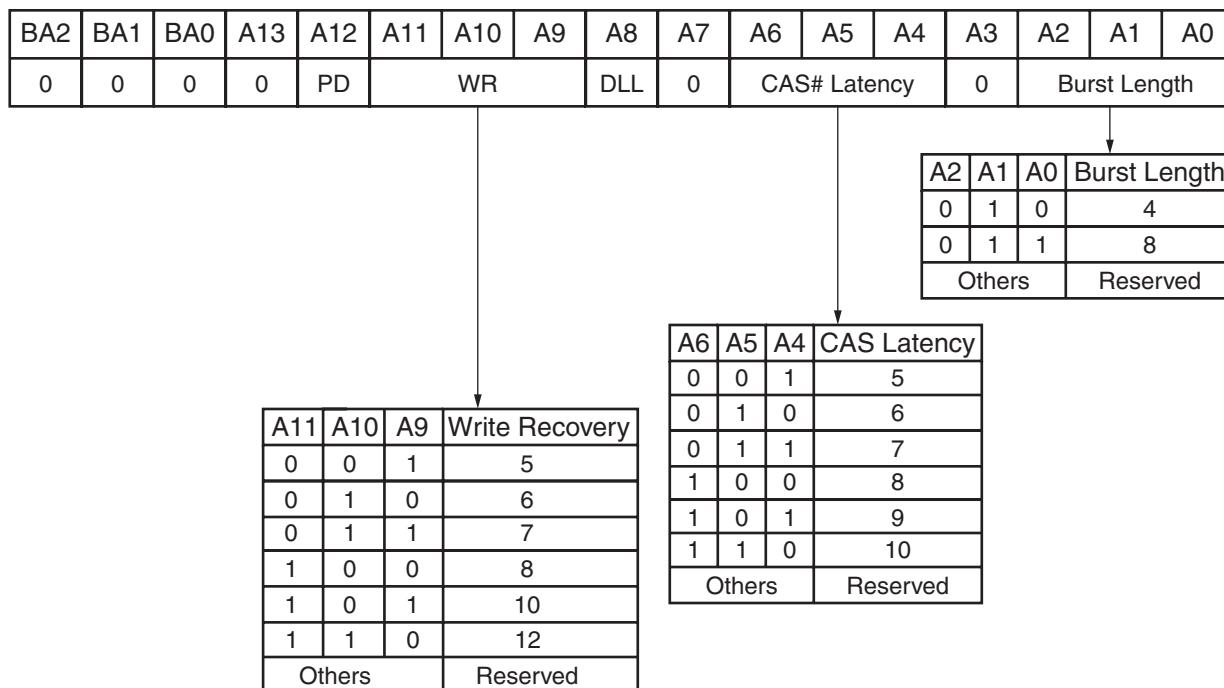
X867_01_082207

Figure 1: **Mode Register**

Table 2 shows the Bank Address bit configuration.

Table 2: **Bank Address Bit Configuration**

BA1	BA0	Mode Register
0	0	Mode Register (MR)
0	1	EMR1
1	0	EMR2
1	1	EMR3

Extended Mode Register Definition

In addition to the functions controlled by the Mode register, the Extended Mode register (Table 3) controls these functions: DLL enable/disable; output drive strength; On-Die Termination (ODT); Posted CAS Additive Latency (AL); off-chip driver impedance calibration (OCD); DQS enable/disable; RDQS/RDQS enable/disable; and OUTPUT disable/enable. OCD is not used in this reference design.

Table 3: **Extended Mode Register**

BA2	BA1	BA0	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	1	0	Qoff	TDQS	0	RTT	OCD Program		R_{TT}	ODS	AL		R_{TT}	ODS	DLL

Extended Mode Register 2 (EMR2)

Table 4: **Extended Mode Register**

BA2	BA1	BA0	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	0	0	0	0	R_{TT} WR		0	SRT	ASR	CWL			0	0	0

Extended Mode Register 3 (EMR3)

Bank Address bits are set to `11` (BA1 and BA0 are set High). Address bits are all set Low.

Initialization Sequence

The initialization sequence used in the controller state machine follows the DDR3 SDRAM specifications. The voltage requirements of the memory must be met by the interface. The following is the sequence of commands issued for initialization.

1. After stable power and clock, a NOP or Deselect command is applied for 200 µs.
2. CKE is asserted.
3. Precharge All command is executed after 400 ns.
4. EMR (2) command is executed. BA0 and BA2 are held Low, and BA1 is held High.
5. EMR (3) command is executed. BA2 is held low, while BA0 and BA1 are both held High.
6. EMR command is executed to enable the memory DLL. BA1, BA2, and A0 are held Low, and BA0 is held High.
7. Mode Register Set command is executed for DLL reset. To lock the DLL, 200 clock cycles are required.
8. ZQ is initialized, and a delay of 200 clock cycles is required.
9. Precharge All command is executed.
10. Two Auto Refresh commands are executed.

11. EMR command is executed to enable OCD default by setting bits E7, E8, and E9 to `1`.

12. EMR command is executed to enable OCD exit by setting bits E7, E8, and E9 to `0`.

After the initialization sequence is complete, the controller issues a dummy write followed by dummy reads to the DDR3 SDRAM memory for the datapath module to select the right number of taps in the Virtex-5 input delay block. The datapath module determines the right number of delay taps required and then asserts the phy_init_done signal to the controller. The controller then moves into the IDLE state.

Precharge Command

The Precharge command is used to deactivate the open row in a particular bank. The bank is available for a subsequent row activation for a specified time (t_{RP}) after the Precharge command is issued. Input A10 determines whether one or all banks are to be precharged.

Auto Refresh Command

DDR3 devices need to be refreshed every 7.8 μs. The circuit to flag the Auto Refresh commands is built into the controller. The controller uses a system clock, divided by 16, to drive the refresh counter. When asserted, the auto_ref signal flags the need for Auto Refresh commands. The auto_ref signal is held High 7.8 µs after the previous Auto Refresh command. The controller then issues the Auto Refresh command after it completes its current burst. Auto Refresh commands are given the highest priority in the design of this controller.

Active Command

Before any Read or Write commands can be issued to a bank within the DDR3 SDRAM memory, a row in the bank must be activated using an Active command. After a row is opened, Read or Write commands can be issued to the row subject to the t_{RCD} specification. DDR3 SDRAM devices also support posted CAS additive latencies; these allow a Read or Write command to be issued prior to the t_{RCD} specification by delaying the actual registration of the Read or Write command to the internal device using additive latency clock cycles.

When the controller detects a conflict, it issues a Precharge command to deactivate the open row and then issues another Active command to the new row. A conflict occurs when an incoming address refers to a row in a bank other than the currently opened row.

Read Command

The Read command is used to initiate a burst read access to an active row. The values on BA0 and BA1 select the bank address. The address inputs provided on $A_0 - A_i$ select the starting column location. After the read burst is over, the row is still available for subsequent access until it is precharged.

Figure 2 shows an example of a Read command with an additive latency of zero. Hence, in this example, the Read latency is five, the same as the CAS latency.

www.xilinx.com
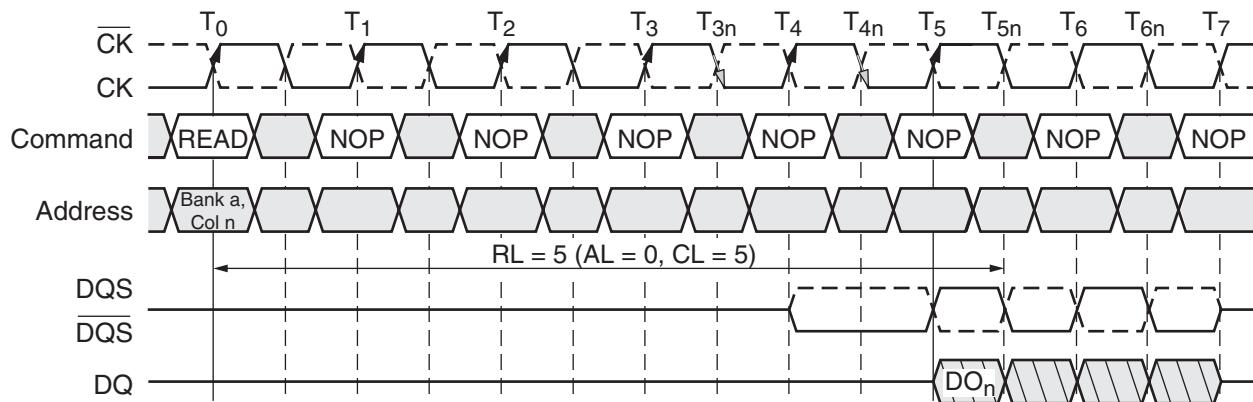
Figure 2: **Read Command Example**

Write Command

The Write command is used to initiate a burst access to an active row. The values on BA0 and BA1 select the bank address while the value on address inputs $A_0 - A_i$ select the starting column location in the active row. DDR3 SDRAMs use a Write Latency (WL) equal to Read Latency (RL).

Write Latency = Read Latency = (Additive Latency + CAS Latency)

Figure 3 shows the case of a Write burst with a WL of 5. The time between the Write command and the first rising edge of the DQS signal is determined by the WL.
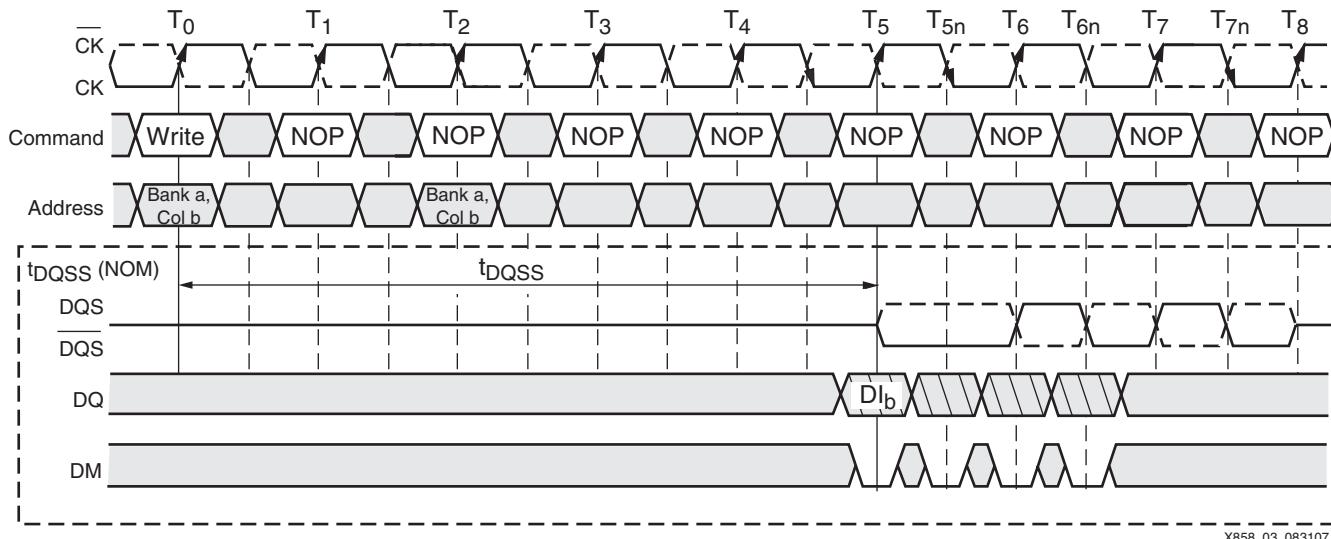
Figure 3: **Write Command Example**

DDR3 SDRAM Interface Design

DDR3 interface block diagram is shown in Figure 4. All the FIFOs in the user interface are asynchronous FIFOs, allowing the user's backend to operate at any frequency.
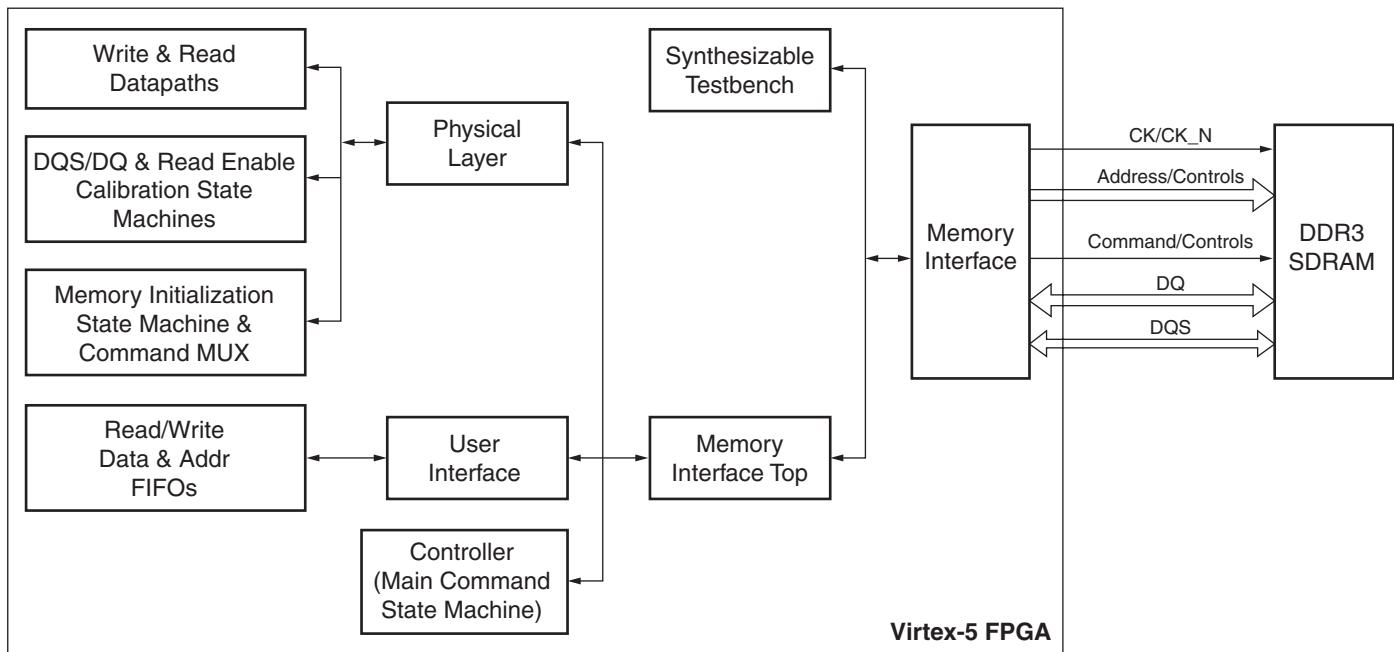
X858_04_090707

Figure 4: **DDR3 Complete Interface Block Diagram**

Sample User Backend and Synthesizable Testbench

A sample user backend and synthesizable testbench block is provided as part of the DDR3 reference design. The backend provides address and data patterns to test read and write accesses between the memory device and the memory interface (DDR3 controller and Physical layer). The backend includes the following blocks: backend state machine, read data comparator, and a data generator module. The data generation module generates the various address and data patterns that are written to the memory. The address locations are pre-stored in a block RAM, being used in this design as a ROM. The address values stored have been selected to test accesses to different rows and banks in the DDR3 SDRAM device. The data pattern generator includes a state machine that issues patterns of data. The backend state machine emulates a user backend. This state machine issues the write or read enable signals to determine the specific FIFO to be accessed by the data generator module.

User Interface

The backend user interface has three FIFOs: the Address FIFO, the Write Data FIFO, and the Read Data FIFO. The first two FIFOs are accessed by the user backend modules, while the Read Data FIFO is accessed by the datapath module used to store the captured Read data.

User-to-Controller Interface

Table 5 lists the signals between the user interface and the controller.

Table 5: **Signals Between User Interface and Controller**

Port Name	Port Width (in bits)	Port Description	Notes
app_af_addr	36	Output of the Address FIFO in the user interface. Mapping of these address bits: • Memory Address 31:0], (CS, Bank, Row, Column)[• Reserved [33:32] • Command Request [35:34]	Monitor FIFO-full status flag to write address into the address FIFO.
af_empty	1	The user interface Address FIFO empty status flag output.	FIFO16 Empty Flag. The controller processes the address on the output of the FIFO when this signal is deasserted.
af_rden	1	Read Enable input to address FIFO in the user interface.	This signal is asserted for one clock cycle when the controller state is Write or Read.
wdf_rden	1	Read Enable input to Write Data FIFO in the user interface.	This signal is asserted for four clock cycles for a burst length of 8. Sufficient data must be available in Write Data FIFO associated with a write address for the required burst length before issuing a Write command. For example, for a 64-bit data bus and a burst length of 4, the user should input two 128-bit data words in the Write Data FIFO for every write address before issuing the Write command.

www.BDTIC.com/XILINX

The memory address (Af_addr) includes the column address, row address, bank address, and chip-select width for deep memory interfaces (Table 6).

Table 6: **Af_addr Memory Address**

Address	Description
Column Address	COL_WIDTH – 1:0
Row Address	COL_WIDTH + ROW_WIDTH – 1:COL_WIDTH
Bank Address	COL_WIDTH + ROW_WIDTH + BANK_WIDTH – 1:COL_WIDTH + ROW_WIDTH
Chip Select	COL_WIDTH + ROW_WIDTH + BANK_WIDTH + CS_BITS – 1:COL_WIDTH + ROW_WIDTH + BANK_WIDTH

Command Request

Table 7 lists the Read and Write command request formats.

Table 7: **Optional Commands**

Command	Description
00	Write
01	Read
10	NOP
11	NOP

Figure 5 shows four consecutive Writes followed by four consecutive Reads with a burst length of 8. Table 8 lists the state signal values for Figure 5.
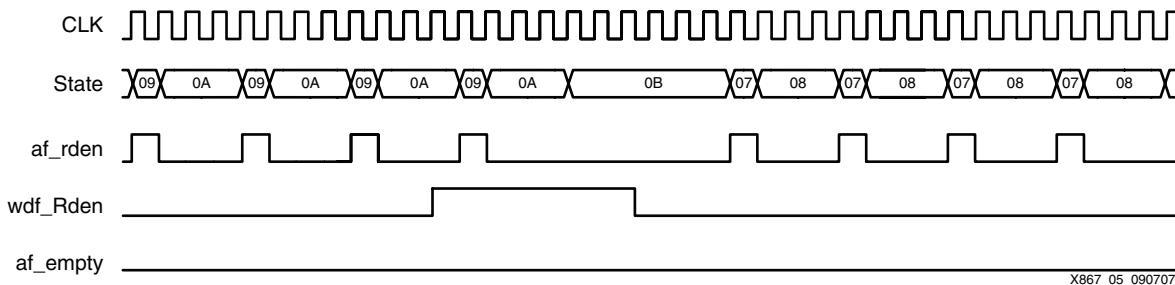
Figure 5: **Consecutive Reads Followed by Consecutive Writes with Burst Length of 8**

Table 8: **Values for State Signals in Figure 5**

State	Description
09	Burst Write
0A	Write Wait
07	Burst Read
0B	Write Read
08	Read Wait

Physical Layer

The physical layer comprises the write datapath, the read datapath, the calibration state machine for DQS and DQ calibration, calibration logic for read enable alignment, and the memory initialization state machine. The write datapath generates the data and strobe signals transmitted during a Write command. The read datapath captures the read data in the read strobe domain.

Write Datapath

The write datapath uses the built-in ODDR available in every Virtex-5 FPGA I/O. The ODDR transmits the data (DQ) and strobe (DQS) signals. The memory specification requires DQS to be transmitted center aligned with DQ. The strobe (DQS) forwarded to the memory is 180° out of phase with CLK0. Therefore, the write data transmitted using ODDR must be clocked by CLK90 as shown in Figure 6. The timing diagram for write DQS and DQ is shown in Figure 7.
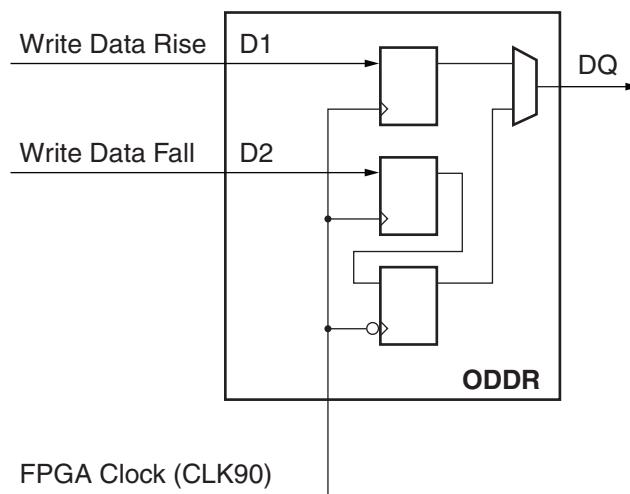

Figure 6: **Write Data Transmitted Using ODDR**
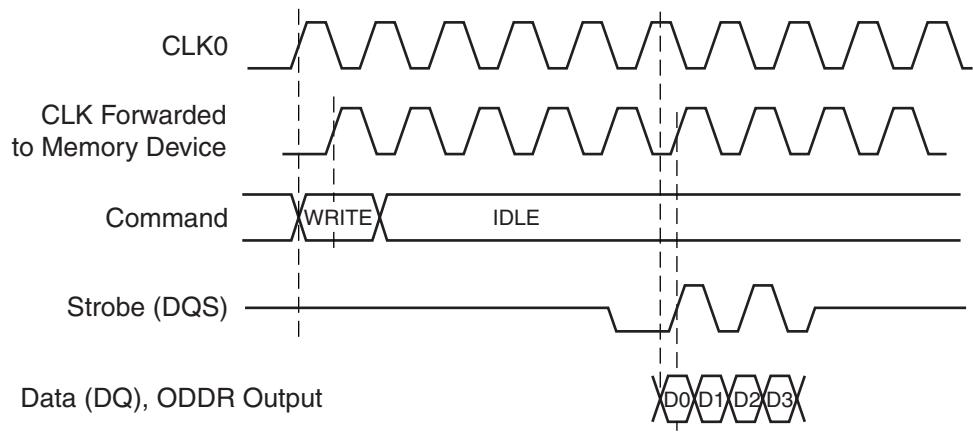
Figure 7: **Write Strobe (DQS) and Data (DQ) Timing for a Write Latency of Five**

Read Datapath

The read datapath compromises the various register stages to capture the read data from the memory and transfer it to the internal FPGA clock domain. This is accomplished by using a combination of ChipSync elements available in each I/O and flip-flops located in the FPGA fabric.

The synchronization stages are:

- First stage: The DQ is captured by the input DDR flop (IDDR) of each DQ I/O. The differential DQS strobe is placed on a clock capable I/O pin pair, drives an IDELAY element and BUFIO local clock network, and clocks each DQ IDDR. The input of each DQ IDDR is a delayed version of the DQ IDDR, delayed using the built-in IDELAY element. The DQ IDELAY is adjusted to provide sufficient timing between the delayed DQ and DQS inputs to the IDDR. The IDELAY setting for each DQ is determined by a timing calibration routine executed one time after system reset.

- Second stage: The outputs of the IDDR (for rising and falling data) are routed to flip-flops located in the FPGA fabric, close to each DQ I/O. The fabric flops are clocked with the core (FPGA) clock. Synchronization is achieved by using the DQ and DQS IDELAY elements to adjust the output of the IDDR relative to the core clock. The IDELAY settings are also determined during the initial timing calibration routine. The output of the flip-flops is now synchronous with the clock used for the rest of the DDR3 interface logic.

Controller Implementation

The controller has the ability to keep four banks open at a time. The banks are opened in the order of the commands that are presented to the controller. In the event that four banks are already opened and an access arrives to the fifth bank, the least recently used bank will be closed and the new bank will be opened. All the banks are closed during auto refresh and will be opened as commands are presented to the controller.

The controller state machine manages issuing the commands in the correct sequencing order while determining the timing requirements of the memory.

Before the controller issues the commands to the memory:

1. The controller decodes the address located in the FIFO.

2. The controller opens a row in a bank if that bank and row are not already opened. In the case of an access to a different row in an already opened bank, the controller closes the row in that bank and opens the new row. The controller moves to the Read/Write states after opening the banks if the banks are already opened.

3. After arriving in the Write state, if the controller gets 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 sees a Write command from the command logic block, the controller waits for the read_to_write time before issuing the Write command. In the Read or Write state, the controller also asserts the read enable to the address FIFO to get the next address.

4. The commands are pipelined to synchronize with the Address signals before being issued to the DDR3 memory.

Reference Design

The reference design for the Virtex-5 DDR3 SDRAM memory controller is based on the DDR2 SDRAM memory controller released through the Memory Interface Generator (MIG) tool. The reference design files can be downloaded from:
http://www.xilinx.com/bvdocs/appnotes/xapp867.zip.

Reference Design Utilization

Table 9 lists the resource utilization for a 32-bit interface, including the physical layer, the controller, the user interface, and a synthesizable testbench.

Table 9: **Resource Utilization for a 32-Bit Interface**

Resources	Utilization	Notes
Slices	1760	Includes the controller, synthesizable testbench, and the user interface.
BUFGs	4	Includes one BUFG for the 200 MHz reference clock for the IDELAY block.
BUFIOs	4	Equals the number of strobes in the interface.
DCMs	1	–
Memory Device	2	The XC5VLX50T interfaces to two 16-bit-wide memory devices.

Conclusion

The DDR3 SDRAM controller along with the data capture technique using DDR IOBs provide a good margin for high-performance memory interfaces. A high margin is achieved when data capture in the DQS domain and data transfer to the FPGA clock domain occurs in the IDDR.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
09/24/07	1.0	Initial Xilinx release.

Notice of Disclaimer

www.BDTIC.com/XILINX