



XAPP853 (v1.3) June 7, 2010

QDR II SRAM Interface for Virtex-5 Devices

Author: Lakshmi Gopalakrishnan

Summary

This application note describes the implementation and timing details of a Quad Data Rate (QDR II) SRAM interface for Virtex®-5 devices. The synthesizable reference design leverages the unique I/O and clocking capabilities of the Virtex-5 family to achieve performance levels of 300 MHz (600 Mb/s), resulting in an aggregate throughput for each 36-bit memory interface of 43.2 Gb/s.

The design greatly simplifies the task of read data capture within the FPGA while minimizing the number of resources used. A straightforward user interface is provided to allow simple integration into a complete FPGA design utilizing one or more QDR II interfaces.

Introduction

QDR SRAM devices were developed in response to the demand for higher bandwidth memories targeted at networking and telecommunications applications. The basic QDR architecture has independent read and write datapaths for simultaneous operation. Both paths use Double Data Rate (DDR) transmission to deliver two words per clock cycle, one word on the rising clock edge and another on the falling edge. The result is that four bus widths of data (two read and two write) are transferred during each clock period, hence the quad data rate name.

Table 1 summarizes the Virtex-5 FPGA QDR II reference design described in this application note, including performance goals and device utilization details.

Table 1: Virtex-5 FPGA QDR II Reference Design Specifications

Parameters		Specification Details
		Speed Grade / Performance
Maximum Frequency, by speed grade (over $\pm 3\%$ voltage range)		-1 / 250 MHz
		-2 / 300 MHz
		-3 / 300 MHz
Device Utilization	Slices	653
	GCLK Buffers	3
	FIFO36	5
QDR II SRAM Operation		Four-word and two-word burst
Bus Widths Supported		18, 36, and 72 bits
I/O Standard		HSTL_I_18 (1.8V Signaling)
HDL Language Support		Verilog and VHDL
Target Memory Device for Verification		Four-word burst: Samsung K7R643684M-FC30 (2Mx36)
		Two-word burst: Cypress CY7C1520JV18-300BZXC (2Mx36)

The QDR I and QDR II memory specifications were jointly defined and developed by the QDR consortium (Cypress, IDT, NEC, Samsung, and Renesas). References to additional information about the QDR specifications, the QDR consortium, and QDR memory products are in [“Additional Resources,” page 18](#).

QDR memory devices are offered in both two-word burst and four-word burst architectures. The two-word burst devices transmit two words per Read or Write request. A DDR address bus is used to allow Read requests during the first half of the clock period and Write requests during the second half of the clock period. In contrast, four-word burst devices transmit four words per Read or Write request, and hence only require a Single Data Rate (SDR) address bus to maximize data bandwidth. Read and Write operations must be requested on alternating clock cycles (i.e., non-overlapping), allowing the address bus to be shared.

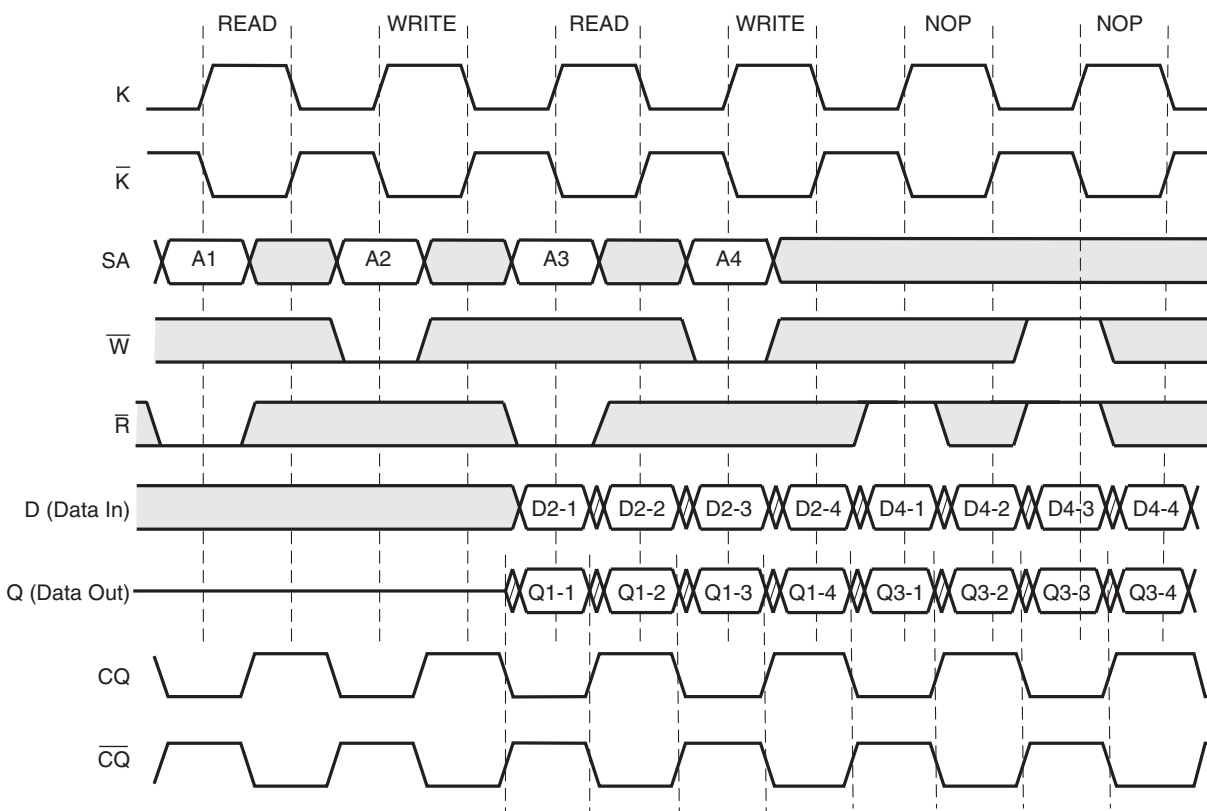
One of the unique features of the QDR II architecture is the echo-clock (CQ) output, which is frequency locked to the device input clock (K) but edge aligned to the data transmitted on the Read path outputs (Q). The CQ clock output is re-timed to align with the Q data outputs using a delay-locked loop (DLL) circuit internal to the QDR II memory device. This clock forwarding, or source-synchronous, interface method allows greater timing margin for the read data capture operation at the far-end device.

[Figure 1, page 3](#) is a timing diagram showing concurrent Read/Write operations on a four-word burst QDR II memory interface. All inputs to the QDR II memory are synchronous to the input clocks (K and \bar{K}) and are typically presented to the memory, center aligned with respect to the K and \bar{K} clock edges.

Note: The active-Low Read Control (\bar{R}) and Write Control (\bar{W}) pins alternate clock cycles to enable a single shared SDR address bus (SA).

The Write bus Data In (D) values are transmitted to the memory in DDR mode beginning on the next rising edge of K clock after the Write Control pin is active. The Read bus Data Out (Q) values are transmitted from the memory in DDR mode, edge aligned with the CQ and \bar{CQ} echo clock outputs. The first word on the Read bus is transmitted on the rising edge of the \bar{CQ} clock output following the next rising edge of the \bar{K} input clock.

QDR II memories also have active-Low Byte Write (\bar{BW}) enable pins to use when selecting specific bytes from the Data In (D) word to write to the memory. These signals are omitted from [Figure 1, page 3](#) for clarity.



X853_01_042706

Figure 1: Four-Word Burst QDR II SRAM Timing Diagram with Concurrent Read and Write Operations

In [Figure 1](#), the DDR address bus allows Read addresses to be presented to the memory during the first half of the clock period and Write addresses to be presented during the second half of the clock period. Thus, the active-Low Read Control (\bar{R}) and Write Control (\bar{W}) pins can be asserted on the same clock cycle.

The two Write bus Data In (D) values are transmitted to the memory in DDR mode starting on the rising edge of K clock prior to the Write address assertion. The Read bus Data Out (Q) values are transmitted from the memory in DDR mode, edge aligned with the CQ and \bar{CQ} echo clock outputs. The first word on the Read bus is transmitted on the rising edge of the \bar{CQ} clock output following the next rising edge of the /K input clock.

Design Overview

Figure 2 is a high-level block diagram of the Virtex-5 FPGA QDR II memory reference design, showing both the external connections to the QDR II memory device and the internal FPGA fabric interface for initiating Read/Write commands.

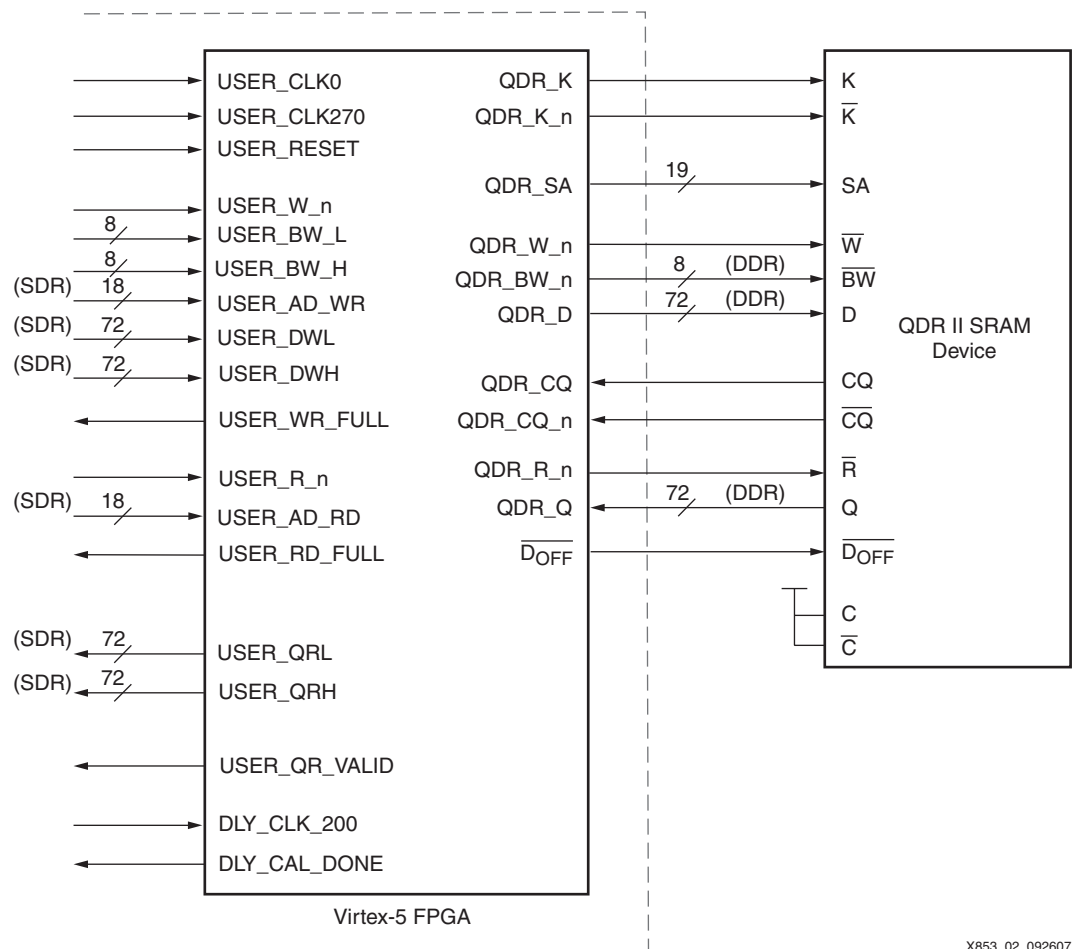
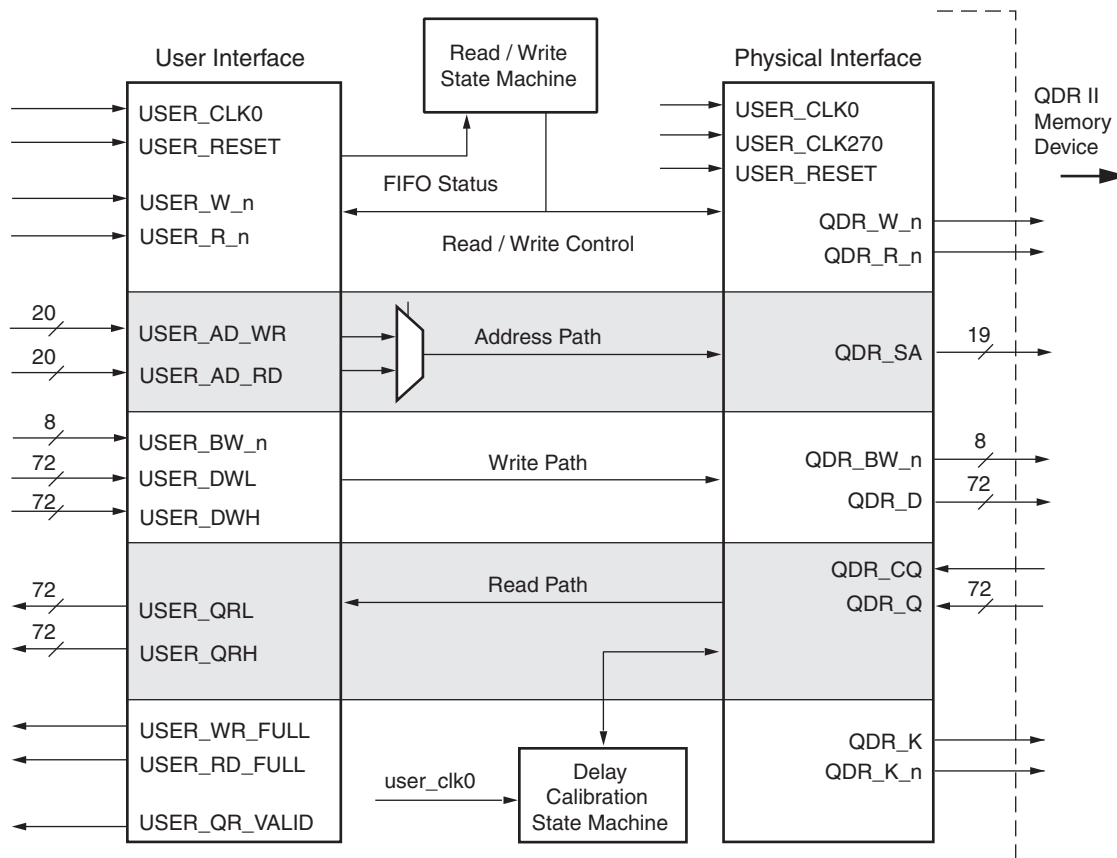


Figure 2: Virtex-5 FPGA QDR II Memory Reference Design Block Diagram

The C and \bar{C} pins of the QDR II memory device are tied High in Figure 2. The Memory Interface Generator (MIG) tool generates the C and \bar{C} pins, which can be removed by the user if they are tied High on the board. The D_{OFF} pin is driven by the interface design and is therefore allocated by the MIG tool. This can also be changed to be tied High at the memory following the memory vendor specifications.

As shown in Figure 3, page 5, the Virtex-5 FPGA QDR II reference design is composed of four main elements:

- User Interface
- Physical Interface
- Read/Write State Machine
- Delay Calibration State Machine



X853_03_092707

Figure 3: Components of the QDR II Reference Design

The user interface uses a simple protocol based entirely on SDR signals to make Read/Write requests. This module is constructed primarily from FIFO16 primitives and is used to store the address and data values for Read/Write operations before and after execution. Refer to “[User Interface](#),” page 6 for more details on the user interface timing protocol.

The Read/Write state machine is responsible for monitoring the status of the FIFOs within the user interface module, coordinating the flow of data between the user interface and physical interface, and initiating the actual Read/Write commands to the external memory device. It ensures execution of Read/Write operations with minimal latency in a concurrent manner as per the requirements of the QDR II memory specification.

The physical interface is responsible for generating the proper timing relationships and DDR signaling to communicate with the external memory device in a manner that conforms to its command protocol and timing requirements.

The delay calibration state machine is an integral component of the physical layer used to achieve maximum performance while greatly simplifying the task of read data capture inside the FPGA. Each input pin on a Virtex-5 device has a programmable delay element (IDELAY) that can be dynamically adjusted to control the amount of delay on the input path across a 5 ns window. The delay calibration state machine leverages this unique capability to adjust the timing of the read data returning from the memory device so that it can be synchronized directly to the global FPGA system clock (USER_CLK0) without any complex data capture techniques.

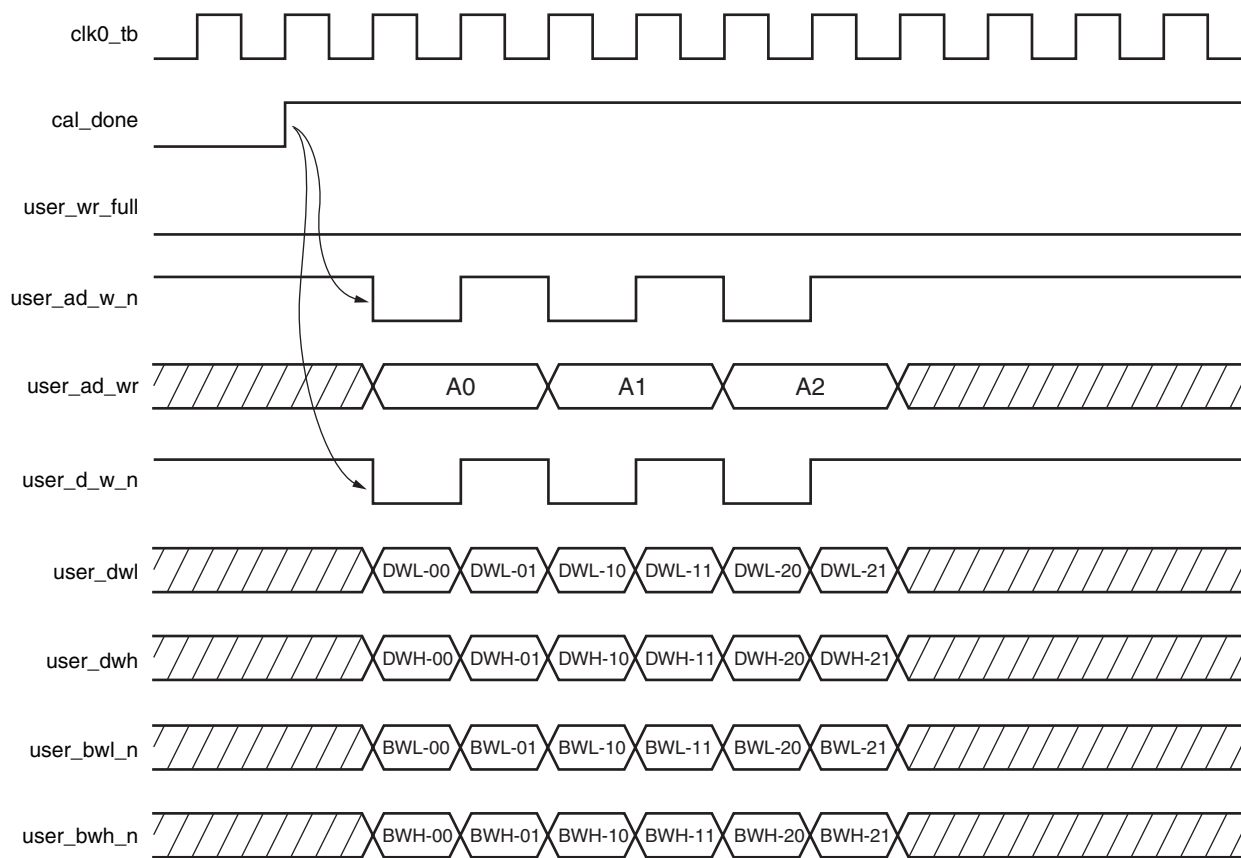
Implementation Details

The QDR II memory reference design was implemented to take advantage of the unique capabilities of the Virtex-5 family. Advances in I/O, clocking, and storage element technology enable the high-performance, turnkey operation of this design. The following sections describe the design implementation in further detail.

User Interface

The user interface module utilizes FIFO36 blocks to store the address and data values for Read/Write operations. For Write commands, three FIFO36 blocks are used, one to store the Write address (USER_AD_WR) and byte write enable (USER_BW_n) signals, and two to store the Low (USER_DWL) and High (USER_DWH) 72-bit data words to be written to the memory. Read commands also use FIFO36 blocks, one to store the Read address (USER_AD_RD) and two FIFO36 blocks to store the Low (USER_QRL) and High (USER_QRH) read data words captured in each bank.

Figure 4 shows the timing protocol required to issue Read/Write requests to the user interface when using the four-word burst reference design. As mentioned previously, the interface uses all SDR signals synchronized to the main FPGA design system clock (USER_CLK0).



X853_04_110107

Figure 4: **Four-Word Burst User Interface Timing Protocol**

Write requests are made via an active-Low USER_W_n signal during the rising edge of USER_CLK0. The Write address (USER_AD_WR) must be presented on this same clock edge. The first and second 36-bit data words to be written to the memory are also presented at this time to the 36-bit USER_DWL and USER_DWH input buses, respectively. The third and fourth words of the four-word burst are presented to USER_DWL and USER_DWH, respectively, on the next rising edge of USER_CLK0.

Read requests are made via an active-Low USER_R_n signal during the rising edge of USER_CLK0. The 18-bit Read address (USER_AD_RD) must be presented on this same clock edge. Read data in the system clock domain is provided to the user backend along with the USER_QR_VALID signal and is not stored in the read data FIFOs.

Unlike the QDR II memory itself, the user interface can accept Read and Write requests on the same clock cycle, as shown on the third cycle of [Figure 4, page 6](#). The Read/Write state machine manages the interleaving of Read and Write requests to the external memory device, relieving the user interface of this responsibility.

The user interface also includes a number of signals not shown in [Figure 4, page 6](#) that indicate the status of the Read/Write FIFOs. An active-High USER_WR_FULL output indicates that the Write FIFOs are full. No more Write requests are allowed under this condition until the Write request queue is reduced. Any Write requests made while USER_WR_FULL is High are ignored. A similar situation applies to the USER_RD_FULL signal for Read requests.

[Figure 5](#) shows the timing protocol required to issue Read/Write requests to the user interface when using the two-word burst reference design.

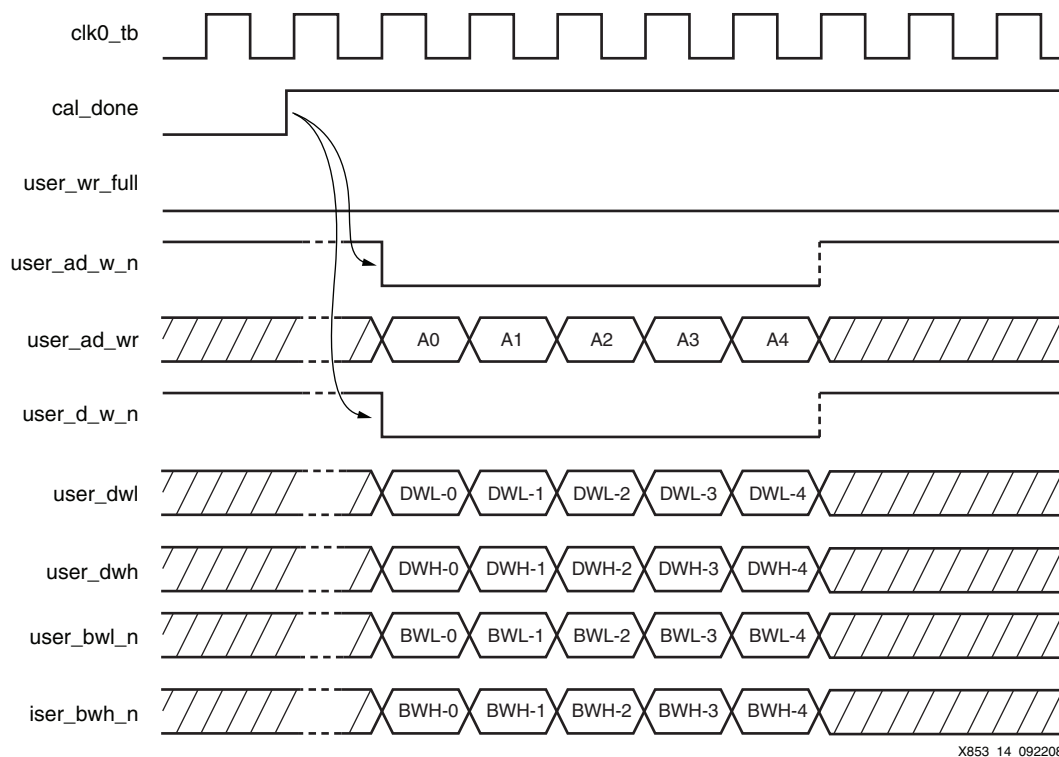


Figure 5: Two-Word Burst User Interface Timing Protocol

Read/Write State Machine

The state diagram for the four-word burst Read/Write state machine is shown in [Figure 6, page 8](#). This state machine is responsible for coordinating the flow of data between the user interface and physical interface. It initiates the Read/Write commands to the external memory device based on the requests stored in the user interface FIFOs.

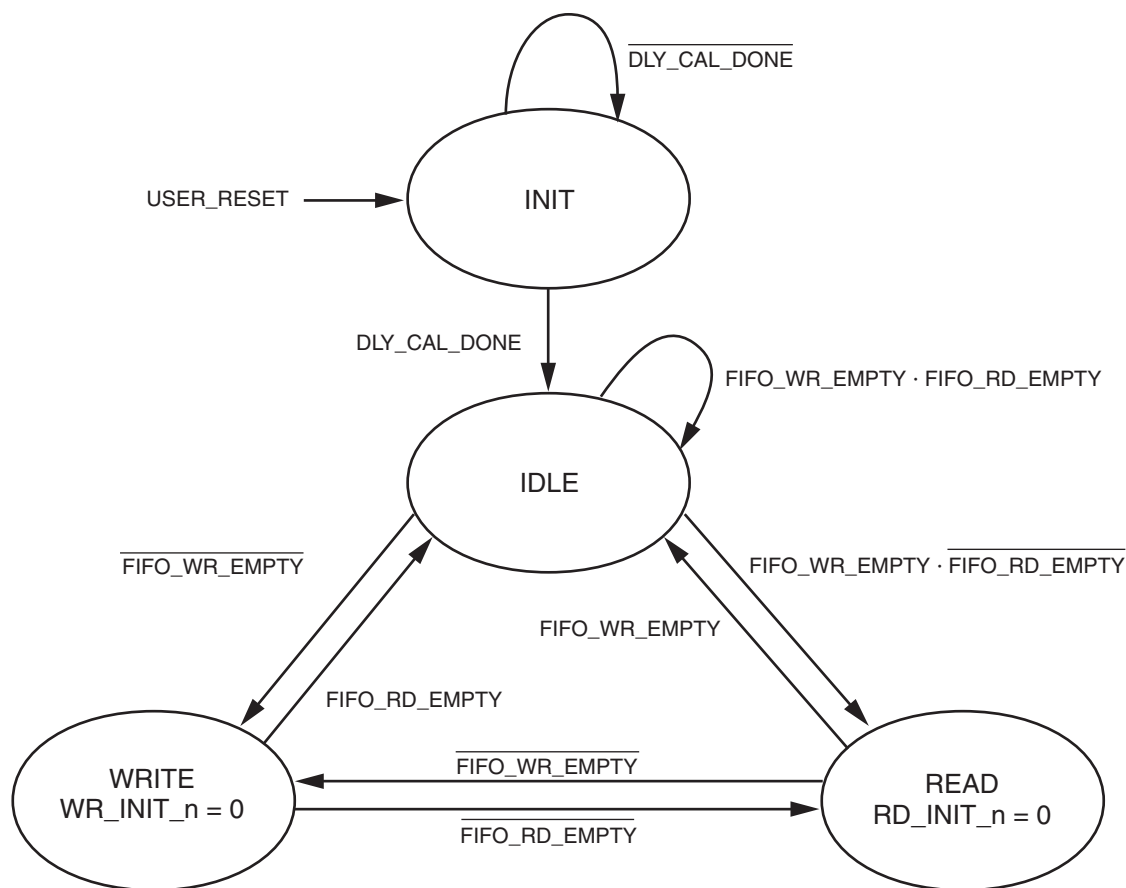
A USER_RESET always returns the state machine to the INIT state. The memory operations are suspended until the delay-calibration state machine has completed adjusting the delay on the IDELAY blocks. Completion of the calibration operation is signaled by an active-High DLY_CAL_DONE input that transitions the Read/Write state machine to the Idle state to await Read/Write requests from the user interface.

From the Idle state, Write commands take precedence on the presumption that a Write to memory must always occur before any valid Read data. When no Read or Write requests are pending, the state machine loops in the Idle state.

A Write request pending in the user interface FIFOs causes transition to the Write state where a Write command is initiated via the internal WR_INIT_n strobe. This strobe pulls the Write address and data values from the FIFO and results in the initiation of the external QDR_W_n Write control strobe to the memory device.

Assuming there is a pending Read request, the state machine then transitions to the Read state where the internal RD_INIT_n strobe is activated. This strobe pulls the Read address from the FIFOs and launches an external QDR_R_n strobe to the memory device.

The Read/Write state machine continuously monitors the user interface FIFO status signals to determine if there are any pending Read/Write requests. A continuous flow of concurrent Read/Write requests causes the state machine to simply alternate between the Read and Write states, ensuring properly interleaved requests to the external memory. A stream of Write only requests results in alternating Idle and Write states, while a stream of Read requests similarly alternates between Idle and Read states.



X853_05_092208

Figure 6: Four-Word Burst Read/Write State Machine

The state diagram for the two-word burst Read/Write state machine while using the FIFO interface is shown in [Figure 7](#).

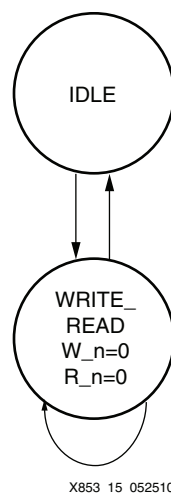


Figure 7: Two-Word Burst Read/Write State Machine with FIFO Interface

Two-Word Burst Design without FIFO Interface

In a two-word burst design, user commands can be transferred directly to the physical layer. The design does not rely on the controller state machine to provide the write and read commands; instead, commands are forwarded directly to the physical layer. The user can select this option by setting the BURST2_FIFO_INTERFACE parameter in the `design_top` file to FALSE. This option can be used when a Write address and a Read address need to be issued to the memory on the same cycle, the empty flag status of the Read Address FIFO and Write Address FIFO can be different, and the command order needs to be preserved.

Physical Interface

The physical interface of the Virtex-5 FPGA QDR II memory reference design generates the actual I/O signaling and timing relationships for communication of Read/Write commands to the external memory device, including the DDR data signals. It provides the necessary timing margins and I/O signaling standards required to meet the overall design performance specifications. All I/O signals for the QDR II memory design use HSTL-I signaling. This section details each component of the physical interface.

Clocking Scheme

The QDR II memory design uses Output DDR (ODDR) primitives found in all Virtex-5 FPGA I/O blocks. These built-in DDR register functions greatly simplify the task of generating the proper clock, address, data, and control signaling for communication to the QDR II memory device. Both the IDDR and ODDR primitives have various modes of operation to determine how the captured or transmitted DDR data is presented to the FPGA fabric and I/O pins, respectively.

The clocking scheme ([Figure 8, page 10](#)) in the QDR II design uses the ODDR registers in opposite-edge mode to generate the QDR_K and QDR_K_n clocks for the memory device. This clock forwarding methodology effectively removes the clock-to-out parameter of the FPGA from timing margin considerations because the clock signals have nearly identical timing in comparison to the QDR II memory address, data, and control signals. All externally transmitted signals are therefore matched with respect to the clock-to-out parameter.

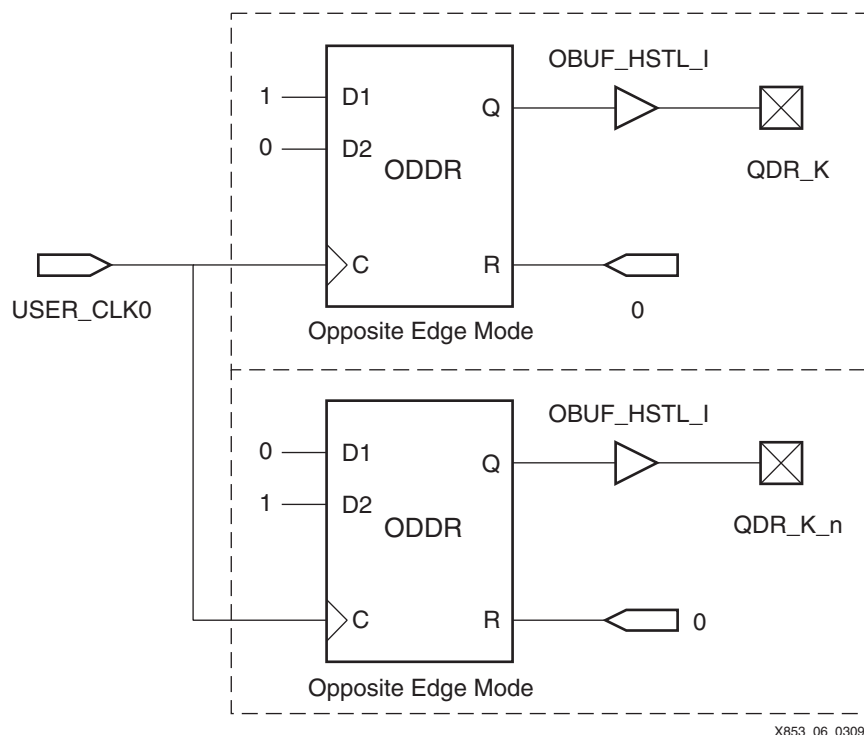


Figure 8: Clock Forwarding Scheme Based on ODDR Register Function

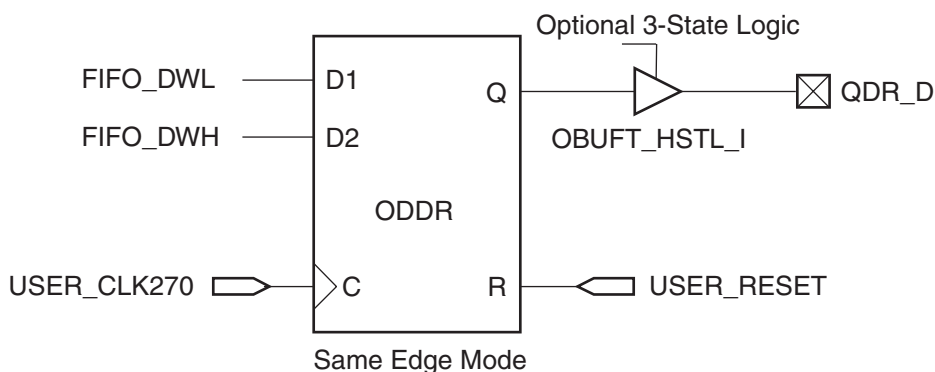
Write Path

The Write path to the QDR II memory includes the address, data, and control signals necessary to execute a Write operation. The Write address (QDR_AD_WR), control strobe (QDR_W_n), and byte write enable (QDR_BW_n) signals all use SDR formatting. However, the Write data values (QDR_D) utilize DDR signaling to achieve the required two-word or four-word burst within the allotted clock periods.

All of these Write path signals must be presented center aligned with respect to the QDR_K and QDR_K_n clock edges. For this reason, the output registers for these signals are synchronized to the USER_CLK270 clock. This signal operates at the same frequency but is 270° (75% of the clock period) out-of-phase with respect to USER_CLK0. This ensures adequate setup and hold margins for the memory device with respect to the incoming QDR_K and QDR_K_n clock edges.

Figure 9, page 11 demonstrates the use of USER_CLK270 and the ODDR registers to generate the DDR signaling required for the QDR_D Write datapath. The ODDR register is configured in same-edge mode, allowing both 72-bit data words (FIFO_DWL and FIFO_DWH) to be captured from the FPGA fabric on the same rising edge of USER_CLK270. The FIFO_DWL value is transmitted immediately after this rising edge onto the QDR_D Write data bus, while the FIFO_DWH value is subsequently transmitted out of the ODDR block on the next falling edge of USER_CLK270. This process repeats to generate a four-word Write data burst.

The Read/Write address, byte write enables, and Read/Write control strobes are generated in a similar manner using a single flip-flop within the I/O block to create SDR signals synchronized to USER_CLK270.



X853_07_092607

Figure 9: Write Datapath Implementation

Read Datapath

The CQ-based data capture scheme enables capture of read data from the memory at very high clock rates. This scheme of data capture uses the ISERDES feature available in every I/O. The incoming clock (CQ) and data (Q) are delayed and synchronized to the system clock (CLK0) inside the ISERDES module.

The read datapath comprises two stages: read data capture and read data recapture. Both stages are implemented inside the built-in ISERDES available in every Virtex-5 FPGA I/O. The CQ signal routed through the BUFIO captures the incoming read data (Q) in the first set of registers inside the ISERDES block. The second set of registers is used to transfer the data in the CQ domain to the system clock domain.

The ISERDES has three clock inputs: CLK, OCLK, and CLKDIV. The read data (Q) capture is done in the CLK (CQ) domain and transferred to the system clock using OCLK and CLKDIV (Figure 10, page 12):

- CLK: The read clock (CQ), routed through the BUFIO, provides the CLK clock input.
- OCLK and CLKDIV: These clocks serve to deserialize the incoming data. The system clock (CLK_0) provides inputs to OCLK and CLKDIV because the data transfer happens at the same frequency as the interface speed.

The data captured in the ISERDES can be written into built-in FIFO36 modules available inside the Virtex-5 FPGAs.

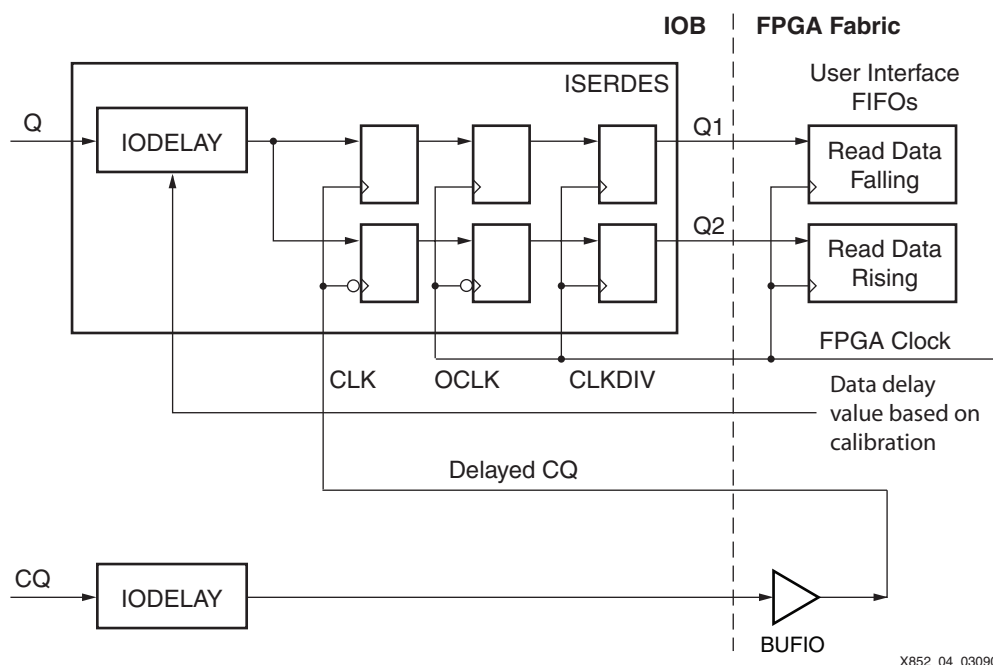


Figure 10: Read Data Capture Using ISERDES

The CQ-based data capture scheme requires placing the read clocks (CQ and \overline{CQ}) on clock-capable I/Os (CCIOs), from where they can access the BUFIOs available inside each bank.

Cascaded Digitally Controlled Impedance for 36-Bit Component Interfaces

In Virtex-5 devices, I/O banks that need Digitally Controlled Impedance (DCI) reference voltages can be cascaded with other DCI I/O banks. One set of VRN/VRP pins can be used to provide a reference voltage to several I/O banks. With DCI cascading, one bank (the master bank) must have its VRN/VRP pins connected to external reference resistors. Other banks in the same column (slave banks) can use DCI standards with the same impedance as the master bank without connecting the VRN/VRP pins on these banks to external resistors. DCI impedance control in cascaded banks is received from the master bank. This results in more usable pins and reduced power usage because fewer VR pins and DCI controllers are used. DCI enables all 36 bits of read data, as well as the CQ and CQ# clocks, to be placed in the same bank when interfacing with 36-bit QDR II components.

The MIG software has partial support for Cascaded DCI. When Cascaded DCI is selected, the MIG tool utilizes VRP and VRN pins in all the banks for user pin allocation. Thus, VRN and VRP pins for the master bank must be reserved using reserve pin options in the MIG tool. After the design is generated with Cascaded DCI support, the .ucf file must be modified manually to specify the master and slave banks. The master bank provides the reference VRN/VRP voltage to all the slave banks. The syntax for representing the Cascaded DCI in the .ucf file is:

```
CONFIG DCI_CASCADE = "<master> <slave1> <slave2> ...";
```

These rules need to be followed to use Cascaded DCI:

1. The master and slave banks must all reside on the same column (left, center, or right) on the device.
2. Master and slave banks must have the same V_{CCO} and V_{REF} (if applicable) voltage.

For more information about Cascaded DCI, refer to DCI Cascading in [UG190](#), Virtex-5 FPGA User Guide.

CQ/CQ# Implementation

The controller uses CQ and CQ# for capturing read data of a 36-bit component. CQ and CQ# are placed on the P pins of the clock-capable I/Os. For a 36-bit component, CQ is used to capture the first 18 bits of the read data, and CQ# is used to capture the second 18 bits of the read data. For an 18-bit component, only CQ is used to capture the read data. CQ# is not used and is connected to dummy logic. This dummy logic is only used to retain the CQ# pin during place and route (PAR). The CQ# pin can be used, if needed.

Delay Calibration

The delay calibration logic is responsible for providing the required amount of delay on the Read data (Q) and the Read Clock (CQ) to align the clock in the data valid window.

The delay calibration is enabled due to the available IODELAY elements in all the I/Os in the Virtex-5 device. The IODELAY elements delay the input read data by increments of 75 ps, up to a maximum delay of 5 ns. IDELAYCTRLs are available in every bank in Virtex-5 devices and help to maintain the resolution of the IODELAY elements.

Calibration begins when the IDELAYCTRL ready signal has been asserted. The calibration is performed in three stages:

1. Calibration of CQ with respect to Q.
2. Calibration of CQ and Q with respect to the FPGA clock.
3. Read enable calibration that determines when read data is valid. This helps to generate the write enable to the read data FIFOs.

Calibration of CQ and Q

This stage of calibration helps to align CQ inside the data valid window. CQ is delayed more than the read data by the delay on the BUFIO and the route delay of CQ before it clocks the read data in the ISERDES. In a case where the data valid window is considerably reduced, this delay on the BUFIO can move the edge of the CQ or the CQ_n clock outside of the valid window. This calibration stage helps to avoid the desynchronization of the clock and data. The calibration stage includes a four-word dummy write to the memory with a constant rise data pattern of 1s and a constant fall data pattern of 0s followed by constant reads to the same location until the calibration is completed. The non-transitioning rise and fall data pattern helps to avoid any metastability caused by the FPGA clock in the second and third register stages in the ISERDES.

The steps involved in this stage include:

1. Increment CQ to see if it is within the valid window. If it is, continue to increment CQ until the hold window range is measured.
2. Reset CQ.
3. Increment Q to determine the setup window with respect to CQ.
4. Reset Q.
5. Based on the window detected in steps 1 and 3, delay either CQ or Q to center align CQ inside the valid window.

Calibration of CQ/Q with FPGA Clock

This calibration stage helps to align CQ and Q with respect to the FPGA clock. For reliable data capture and transfer of the data into the FPGA fabric, the calibration is required to align the FPGA clock inside of the data captured by CQ.

This stage includes a four-word dummy write to the memory. The dummy write includes the following pattern: FF/00-55/AA, where FF and 55 refer to the rising data pattern during a four-word burst, and 00 and AA refer to the falling data pattern.

The steps involved in this calibration stage are:

1. Increment CQ and Q and determine the valid window range.
2. If this window is insufficient, which indicates that the CQ and FPGA clock edge are aligned very closely to each other, continue to increment CQ and Q to determine the valid window with respect to the next FPGA clock edge.
3. The calibration is complete when at least 15 taps of window are available at frequencies above 250 MHz and half the clock period worth of taps are available at frequencies lower than 250 MHz between CQ and the FPGA clock.

Read Enable Calibration

This stage is required to generate the write enable for the read data FIFOs. This includes non-contiguous read commands to align the data valid signal to the read data at the output of the ISERDES.

The calibration logic, built using an SRL16 as a shift register, helps to determine the number of register stages required by the read command signal to generate the correct write enable signals. One read enable signal is generated for data captured by each CQ or CQ_n clock. Based on the count value on the SRL shift register from all the banks, the read data from all the banks is aligned and presented to the user backend along with a user_qr_valid signal.

Board Design Considerations

While the Virtex-5 family offers many advanced I/O and clocking-related features to greatly simplify memory interface design, attention must still be paid to basic board design criteria for a reliable and high-performance interface.

Specifically, the source synchronous nature of the Read and Write path interfaces requires matched board trace lengths for the interface clock, data, and control signals.

For example, the trace lengths of the QDR II memory device input signals (QDR_K, QDR_K_n, QDR_W_n, QDR_R_n, QDR_SA, QDR_BW_n, and QDR_D) must be well matched to present the control, address, and data lines to the memory device with adequate setup and hold margins. The implementation of the physical interface ensures these signals are center aligned to the QDR_K and QDR_K_n clock edges when leaving the FPGA device outputs. The board traces must ensure that this relationship continues to the memory device inputs.

Similarly, the QDR II memory device output signals (QDR_Q, QDR_CQ) must have well-matched trace lengths for the signals to all arrive edge aligned at the inputs to the Virtex-5 device. This trace length matching is critical to the implementation of the direct-clocking Read data capture methodology. Any reasonable board design tool can match these traces within an acceptable tolerance with little effort.

Timing Analysis

The Virtex-5 FPGA QDR II memory reference design leverages the unique I/O and clocking features of the device to maximize performance and timing margins, while greatly reducing the need for detailed placement and pinout analysis.

This section presents an example timing analysis for the address/control paths, the Write datapath, and the Read (or capture) datapath.

Address/Control Paths

As discussed previously, the Read/Write address bus, byte write enable signals, and Read/Write control strobes are all synchronized to the USER_CLK270 clock. This ensures that these SDR signals have adequate setup and hold margins to the memory device with respect to the incoming QDR_K and QDR_K_n clock edges derived from USER_CLK0.

Table 2 shows an example timing analysis for these signals based on an interface to a 300 MHz QDR II memory device implemented with a Virtex-5 device.

Table 2: Address and Control Signal Timing Analysis

Parameter	Value (ps)	Leading-Edge Uncertainties	Trailing-Edge Uncertainties	Description
T _{CLOCK}	3333	–	–	Clock period at 300 MHz
T _{CLOCK_SKEW_FPGA}	100	50	50	Clock skew from TRACE analysis
T _{PACKAGE_SKEW}	60	30	30	Maximum package skew within bank
T _{SETUP}	400	400	0	Setup time from memory data sheet
T _{HOLD}	400	0	400	Hold time from memory data sheet
T _{PCB_LAYOUT_SKEW}	±50	50	50	Maximum skew between board traces based on estimated match tolerance
T _{JITTER}	100	50	50	Jitter component associated with the difference between USER_CLK0 and USER_CLK270
Total Uncertainties	–	580	580	
Valid Window	2173	–	–	Worst-case window

Write Datapath

The Write datapath (QDR_D) is also synchronized to USER_CLK270. However, the Write data words are transmitted as DDR values, and therefore must have adequate setup and hold margins with respect to both the rising edge of QDR_K and the rising edge of QDR_K_n. Accordingly, the timing analysis for the Write datapath shown in Table 3 incorporates the maximum duty cycle distortion of the memory clocks and is based on a 300 MHz QDR II memory device and a Virtex-5 device, -2 speed grade.

Table 3: Write Datapath Timing Analysis

Parameter	Value (ps)	Leading-Edge Uncertainties	Trailing-Edge Uncertainties	Description
T _{CLOCK}	3333	–	–	Clock period at 300 MHz
T _{CLOCK_PHASE}	1666.5	–	–	Clock phase (50% of clock period)
T _{DCD}	150	–	–	Duty cycle distortion of memory clock
T _{DATA_PERIOD}	1516.5	–	–	Total data period, T _{CLOCK_PHASE} – T _{DCD}
T _{CLOCK_SKEW_FPGA}	100	50	50	Clock skew from TRACE analysis
T _{PACKAGE_SKEW}	60	30	30	Maximum package skew within bank
T _{SETUP}	300	300	0	Setup time from memory data sheet
T _{HOLD}	300	0	300	Hold time from memory data sheet
T _{PCB_LAYOUT_SKEW}	±50	50	50	Maximum skew between board traces based on estimated match tolerance
T _{OUT_OFFSET}	280	140	140	Maximum offset between different outputs of DCM
T _{JITTER}	100	50	50	Jitter component associated with difference between USER_CLK0 and USER_CLK90

Table 3: Write Datapath Timing Analysis (Cont'd)

Parameter	Value (ps)	Leading-Edge Uncertainties	Trailing-Edge Uncertainties	Description
Total Uncertainties	–	620	620	Worst-case leading and trailing uncertainties cannot occur simultaneously
Valid Window	276.5	–	–	Worst-case window

Read Datapath (Data Capture)

The Read datapath (QDR_Q) values are captured directly into the USER_CLK0 clock domain inside the SERDES. Table 4 presents the timing analysis for Read data capture on a -2 speed grade Virtex-5 FPGA.

Table 4: Read Timing Analysis at 300 MHz for a -2 Speed Grade Virtex-5 FPGA

Parameter	Value (ps)	Comments
Clock Period	3333	
Data Period	1666.5	
Memory Uncertainties		
$T_{CQ_Q_SKEW}$	540	Includes t_{CQHqV} and t_{CQHqX} .
FPGA Uncertainties		
T_{SAMP_BUFIO}	400	T_{SAMP_BUFIO} parameter from DS202 , Virtex-5 FPGA Data Sheet: DC and Switching Characteristics.
T_{BUFIO_SKEW}	70	T_{BUFIO_SKEW} parameter from the Virtex-5 FPGA data sheet.
$T_{PKGSKEW}$	50	Accounts for any variation after the package flight times have been matched.
$T_{PCB_LAYOUT_SKEW}$	50	Accounts for any board layout skew after careful matching of traces close to ± 20 ps variation, including package flight times.
T_{DCD_BUFIO}	100	T_{DCD_BUFIO} parameter from the Virtex-5 FPGA data sheet.
Tap Jitter	360	This is the jitter introduced by delaying data through the IODELAY. The worst-case jitter value is calculated using the $T_{IDELAYPAT_JIT}$ parameter from the Virtex-5 FPGA data sheet.
Data Valid Window		
Data Window	96.5	

Notes:

- Inter-symbol interference (ISI), crosstalk, user input clock jitter, and contributors to dynamic skew are not considered in this analysis.

Reference Design

The reference design for interfacing Virtex-5 FPGAs to QDR II memories is available both in Verilog and VHDL, and has been integrated with the MIG tool. This tool has been integrated with the Xilinx® CORE Generator™ software. For the latest version of the design, download the IP Update on the Xilinx website at:

http://www.xilinx.com/xlnx/xil_sw_updates_home.jsp

Table 5 shows the reference design matrix.

Table 5: Reference Design Matrix

Parameter	Description
General	
Developer Name	Xilinx
Target Devices (Stepping Level, ES, Production, Speed Grades)	Virtex-5 FPGA
Source Code Provided?	Yes, available through the MIG tool in CORE Generator software
Source Code Format	VHDL, Verilog
Design Uses Code or IP from Existing Reference Design, Application Note, third party, or CORE Generator™ Software?	Y
Simulation	
Functional Simulation Performed?	Y
Timing Simulation Performed?	N
Testbench Provided for Functional and Timing Simulations?	Y
Testbench Format	VHDL, Verilog
Simulator Software and Version	ModelSim 6.3c
SPICE/IBIS Simulations?	Y
Implementation	
Synthesis Software Tools and Version	XST, version 10.1 Synplify Pro 8.8.0.4
Implementation Software Tools and Version	ISE® Design Suite 10.1
Static Timing Analysis Performed?	Y
Hardware Verification	
Hardware Verified?	Y
Hardware Platform Used for Verification	ML561 memory interfaces development board

Conclusion

This application note describes the implementation and timing details of two-word and four-word burst QDR II SRAM interfaces for Virtex-5 devices. The data capture scheme used greatly simplifies the task of read data capture within the FPGA while providing a high-performance, robust, and scalable memory interface solution for current- and next-generation QDR II SRAM memory devices.

Additional Resources

QDR II SRAM Memory Device Vendors:

- Cypress Semiconductor: <http://www.cypress.com/>
- Renesas Technology: <http://www.renesas.com/>
- IDT, Inc.: <http://www.idt.com/>
- Samsung Semiconductor: <http://www.samsung.com/>
- NEC Corporation: <http://www.necel.com/>

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
05/12/06	1.0	Initial Xilinx release.
01/30/08	1.1	<ul style="list-style-type: none"> • Updated all tables. • Updated Figure 2, Figure 3, Figure 4, Figure 6, Figure 8, Figure 9, and Figure 10. • Deleted Figure 9, Figure 10, and Figure 11. • Updated “Design Overview,” “User Interface,” “Read/Write State Machine,” “Read Datapath,” “Delay Calibration,” “Read Datapath (Data Capture),” and “Conclusion” sections. • Made minor text edits.
10/06/08	1.2	<ul style="list-style-type: none"> • Updated description of two-word and four-word burst QDR II SRAM interfaces. • Added Figure 5 and Figure 7. • Updated Table 1, Table 4, and Figure 6. • Moved Table 1 to “Introduction” section. • Added Reference Design Matrix (Table 5). • Updated “Calibration of CQ and Q” section. • Updated “Additional Resources” section.
06/07/10	1.3	Updated Figure 7 . Added “ Two-Word Burst Design without FIFO Interface ” section.

Notice of Disclaimer

Xilinx is disclosing this Application Note to you “AS-IS” with no warranty of any kind. This Application Note is one possible implementation of this feature, application, or standard, and is subject to change without further notice from Xilinx. You are responsible for obtaining any rights you may require in connection with your use or implementation of this Application Note. XILINX MAKES NO REPRESENTATIONS OR WARRANTIES, WHETHER EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL XILINX BE LIABLE FOR ANY LOSS OF DATA, LOST PROFITS, OR FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR INDIRECT DAMAGES ARISING FROM YOUR USE OF THIS APPLICATION NOTE.