Synthesizable QDR SRAM Controller

Author: Olivier Despaux

XAPP262 (v2.2) April 24, 2002

Summary

Quad Data Rate (QDR™) Synchronous Static RAM (SRAM) is one of the highest bandwidth solutions available for networking and telecommunications applications. This low-cost, high-performance solution is ideal for applications requiring memory buffering, traffic management, look-up tables, or link lists. This application note describes an implementation of a QDR SRAM controller, enabling up to 400 Mb/s (DDR400) simultaneous read and write speed in a Virtex™-II -5 speed grade device using a source-synchronous solution.

Introduction

Micron Technology, Cypress, Hitachi, NEC, Samsung, and IDT jointly created the QDR specification in response to an overwhelming demand for higher bandwidth SRAMs. The main feature of the QDR SRAM is that the data inputs and outputs are separate but operating simultaneously. In this application note, Xilinx proposes an interface to this SRAM family.

Because each data bus operates on two words of data per clock cycle, the data rate for each bus doubles the standard data rate. With both buses operating in parallel, the device operates on four bus-widths of data per clock cycle, hence, the Quad Data Rate name. The minimum set is data operating on two words, i.e., two times the device bus width.

QDR SRAM Review

Basics

This section provides a general overview of QDR SRAM technology. It also includes design recommendations. Designers comfortable interfacing this type of memory with FPGAs can go directly to the Controller Design section. QDR SRAMs are designed for the networking market. Table 1 summarizes QDR SRAM specifications.

Table 1: **QDR Specification Summary**

Parameter	Description
Burst Mode (DDR Mode)	2-word burst devices 4-word burst devices
I/O Terminations	HSTL class I, 1.5V or 1.8V
Data Buses	Separate and non-concurrent Write (D) and Read (Q) buses
Device Density	9 Mb, 18 Mb, 36Mb
Internal Pipeline	Two stage pipeline (low initial latency)
Frequency 2-word Burst	100 - 167 MHz (Double Address Rate (DAR) address bus)
Frequency 4-word Burst	100 - 200 MHz (Single Address Rate (SAR) address bus)
Core Voltage	2.5V
Data Clocks C, $\overline{C}$	Output data from the SRAM are synchronized with respect to C and $\overline{C}$ for source-synchronous systems.
QDR SRAM Suppliers	Cypress, Hitachi, IDT, Micron, NEC, Samsung

QDR SRAMs were specifically created for applications having a nearly equal ratio of read and write cycles occurring at almost the same time. Conventional DDR SRAMs are most efficient in applications employing data streaming, or where the read/write ratio is greater than three. The choice of devices with either a 2-word burst or a 4-word burst depends on the address rate and the data write placement.

Address Rate

The 2-word burst QDR SRAM can sustain indefinitely both a 2-word read and a 2-word write each clock cycle. Internally, the first half-clock cycle is used to execute the read function, and the second half-clock cycle is used to execute the write function. The address bus is shared for the read and write data ports, so a double-address rate (DAR) operation is necessary. The rising edge of a master-clock signal "K" is used to register the read address. The falling edge of this clock signal "K" is used to register the write address.

This application note revision is written for 2-word burst devices able to handle a 167 MHz clock in Virtex-II FPGAs (-5 speed grade). The density of the memory device used is nine.

QDR SRAMs have both read and write data buses when operating in DDR mode. On both buses, the clock is center aligned with the data. This physical placement is advantageous to capture data at the memory controller input.

Write Data Placement

Although a minor consideration in terms of system performance, it is necessary to determine the placement of write data in the design. The address rate comes in an SAR in 4-word burst devices. Four-word burst devices read the write addresses on the rising edge of K. Two-word burst devices read the write addresses on the falling edge of K. The data to store is read one clock period later than the write command in 4-word burst QDR SRAMs. Conversely, 2-word burst devices do not have latency (Figure 1).
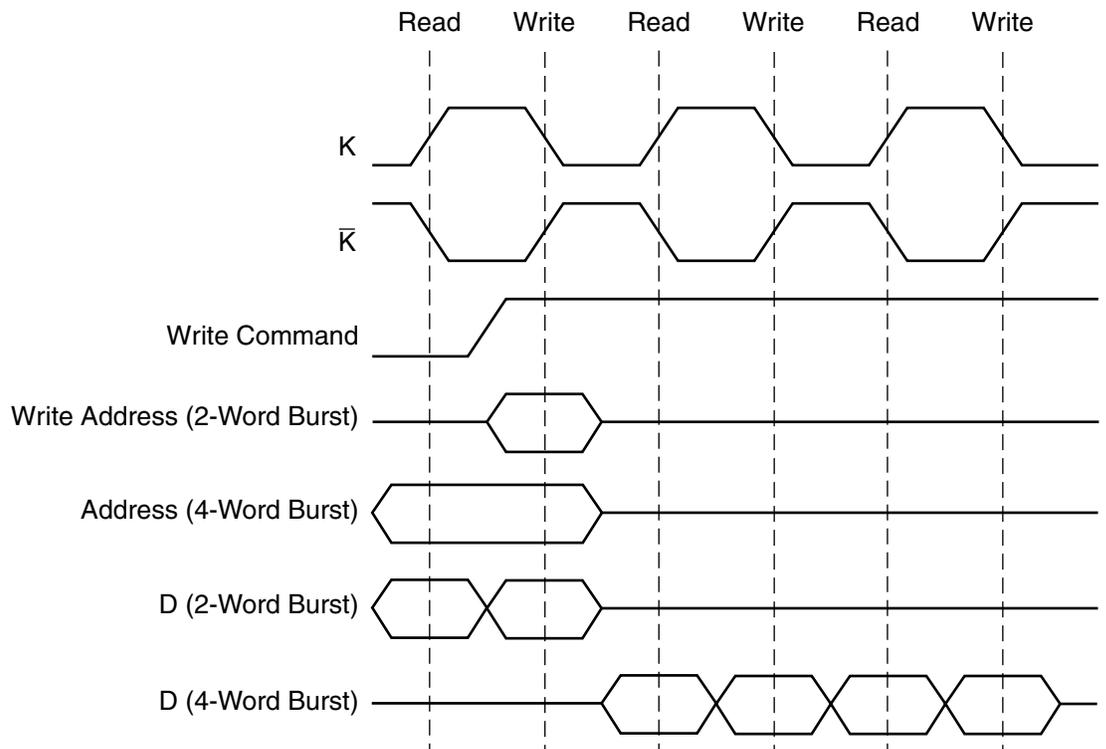
x262_01_032502

Figure 1: **QDR 2-Word versus 4-Word Burst Timing Diagram**

Internal Architecture

The key objective of QDR architecture is to clearly distinguish read and write ports. The QDR architecture is designed to offer the best performance on alternate read and write cycles. Note that if an address read and write occurs in the same clock period, then the data obtained at the memory read port output will be the data written during this same clock period. See Figure 2.
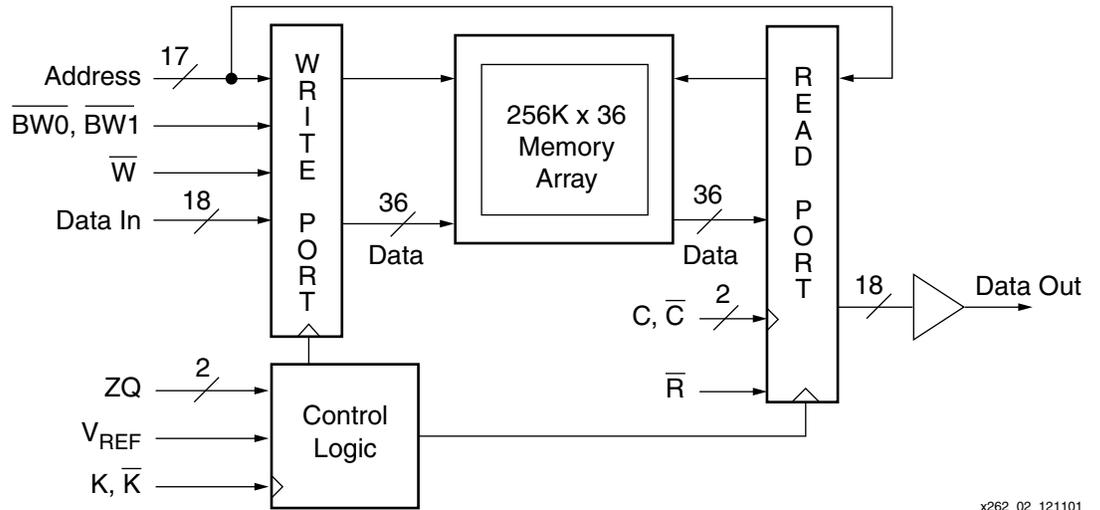

Figure 2: **QDR SRAM Internal Architecture Overview**

Read And Write Operations

One of the best advantages of SRAM devices over SDRAM devices is the ease of interfacing them in terms of control signals. For example, there is no need for sending a refresh command to the memory device.

In 2-word burst devices, the read address location is considered on the rising edge of K clock and the write address location on the falling edge of K clock. The address bus is read as a DDR bus. The 4-word burst devices have a Single Data Rate (SDR) on the address bus. This type of access explains the performance difference between 2-word burst and 4-word burst devices.

Data must be center aligned in regard to K clock when sending the data. On the receive side, data are sent with a guaranteed interval delay in respect to C clock.

Figure 3 shows read and write timing diagrams. Table 2 shows the QDR timing parameters.
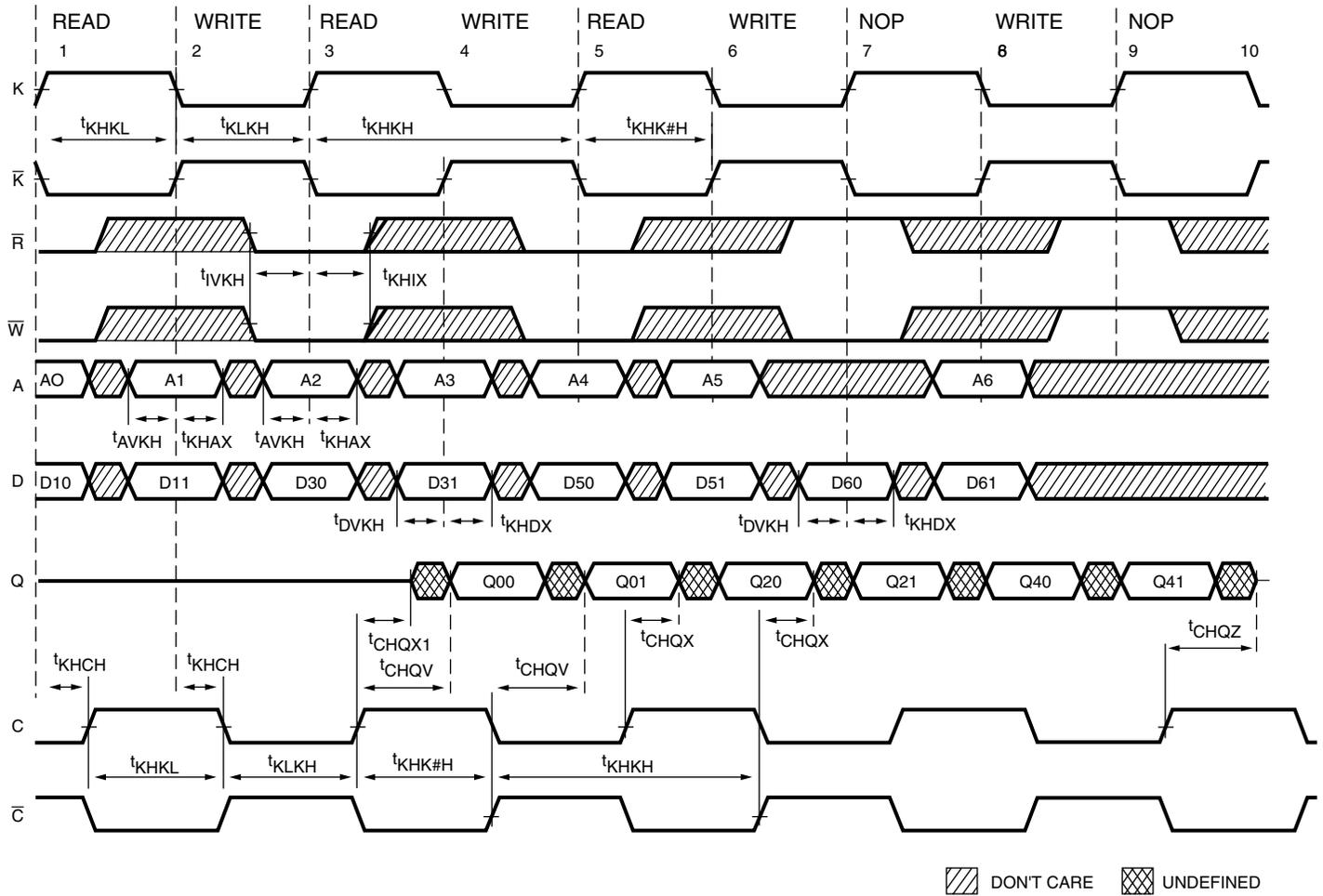
Figure 3: **Read and Write Timing Diagrams**

Table 2: **QDR Timing Parameters**

Parameter	Symbol
Clock	
Clock Cycle Time (K, $\overline{K}$, C, $\overline{C}$)	t_{KHKH}
Clock High time (K, $\overline{K}$, C, $\overline{C}$)	t_{KHKL}
Clock Low time (K, $\overline{K}$, C, $\overline{C}$)	t_{KLKH}
Clock to clock (K$\uparrow \rightarrow$ K$\uparrow$, C$\uparrow \rightarrow$ C$\uparrow$)	t_{KHKH}
Clock to data clock (K$\uparrow \rightarrow$ C$\uparrow$, K$\uparrow \rightarrow$ C$\uparrow$)	t_{KHCH}
Output Times	
C, $\overline{C}$ High to Output Valid	t_{CHQV}
C, $\overline{C}$ High to Output Hold	t_{CHQX}
C High to Output High-Z	t_{CHQZ}
C High to Output Low-Z	t_{CHQX1}
Setup Times	
Address valid to K rising edge	t_{AVKH}

Table 2: **QDR Timing Parameters** *(Continued)*

Parameter	Symbol
Control inputs valid to K rising edge	t_{IVKH}
Data-in valid to K, $\overline{K}$ rising edge	t_{DVKH}
Hold Times	
K rising edge to address hold	t_{KHAX}
K rising edge to control inputs hold	t_{KHIX}
K, $\overline{K}$ rising edge to data-in hold	t_{KHDX}

Special Features

The QDR SRAM incorporates a serial boundary scan test access port (TAP). This feature includes a TAP controller, an instruction register, a boundary scan register, a bypass register and an ID register. The JEDEC Test Access Port (JTAG) is operating at 2.5V. In order to disable this feature, the TCK pin must be tied Low (V_{SS}).

For more information about JTAG functionality and capabilities for QDR SRAMs, refer to the memory vendor's data sheet.

Device Pins

Table 3 is a brief summary of the pins of QDR devices. Please refer to the memory vendor data sheet for more details.

Table 3: **QDR SRAM Pin Summary**

Symbol	Number of Pins (x18)	Type	Name
SA	18	Input	Synchronous Address Inputs
$\overline{R}$	01	Input	Synchronous Read
$\overline{W}$	01	Input	Synchronous Write
$\overline{BW0}$	01	Input	Synchronous Byte Writes D0.D7
$\overline{BW1}$	01	Input	Synchronous Byte Writes D8.D15
K and $\overline{K}$ pair	02	Input	Input Clock
C and $\overline{C}$ pair	02	Input	Output Clock
TMS	01	Input	JTAG Test Input 1
TDI	01	Input	JTAG Test Input 2
TCK	01	Input	JTAG Function Enable
V_{REF}	02	Input	HSTL Input Reference Voltage
ZQ	01	Input	Output Impedance Matching Input
D	18	Input	Synchronous Data Inputs
DNU	02	Output	Do Not Use
Q	18	Output	Synchronous Data Output
TDO	01	Output	IEEE 1149.1 Test Output
V_{DD}	10	Supply	Nominal Power Supply (2.5V)[1]

Table 3: **QDR SRAM Pin Summary** *(Continued)*

Symbol	Number of Pins (x18)	Type	Name
V_{DDQ}	16	Supply	Isolated Output Buffer Supply (1.8V)[1]
V_{SS}	25	Supply	Ground Power Supply (0V)[1]
NC[1]	41	NC[1]	No Connection

Notes:

1. To improve noise immunity and power dissipation, no connect (NC) pins should be connected to ground (GND) on the printed circuit board (PCB). A clean board design is necessary. Use proper capacitors for power supply decoupling, and nominal voltages for supply, voltage references, and logic levels. Xilinx recommends using 1.8V for the HSTL logic levels for better switching characteristics using Virtex-II Platform FPGAs.

QDR SRAM Controller Review

Controller Specification and Overview

This section presents an implementation of a QDR SRAM controller with an Virtex-II FPGA. The controller has a user interface and a QDR SRAM interface. The design may need to be modified depending on the requirements or performance expectations of the designer.

The specifications of this design are the following:

- The 2-word burst design targets a 167-MHz clock in a Virtex-II -5 speed grade device.
- The design is source synchronous and requires the use of K clocks on the transmitter side and C clock on the receiver side.
- The controller has been developed using Micron models and has been successfully tested with Cypress HDL models for 2-word burst devices.
- The controller has an asynchronous Reset capability.
- The design requires two DCMs.
- The estimated dissipated power for one 9 Mb x 18 device in HSTL Class 1, 1.8V is 540 mW.

Table 4 presents a snapshot of this implementation's controller performances, synthesis, and place and route results:

Table 4: **Implementation Performance**

Virtex-II Device XC2V1000 - FG896	Speed Grade		
	-4	**-5**	**-6**
Clock minimum functioning frequency (MHz)	150	167	167
Timing budget margin at this frequency (ps)[1]	400	580	580
Number of Slices	150	150	150
Number of BUFGs	3	3	3
Number of Digital Clock Managers (DCMs)	2	2	2
Number of I/Os	63	63	63

Notes:

1. Using the source-synchronous timing numbers.

Figure 4 is a high-level block diagram of the QDR SRAM controller. The entity *QDR_ctrl* is the top-level QDR controller block containing the address, data send, and receive modules. *USER_gui* passes the signals to the main controller either directly or using a pipeline stage as needed by the designer.
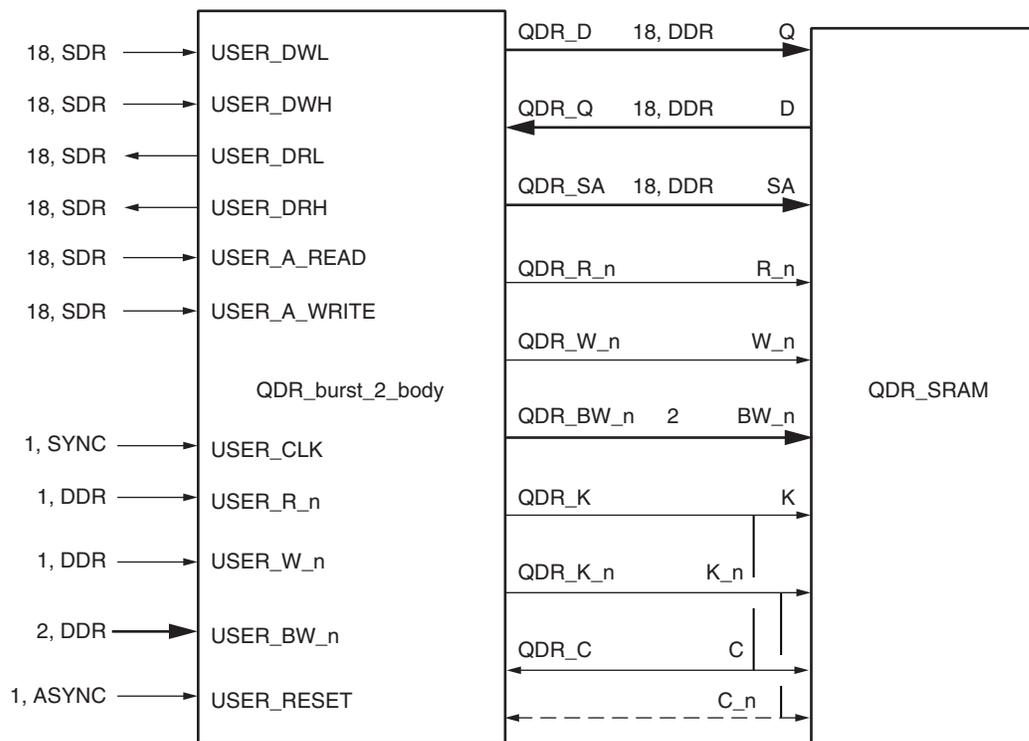
Figure 4: **Top-Level Architecture Block Diagram**

DDR I/Os

Implementation Details

Virtex-II SelectI/O™ inputs and outputs hardware support for DDR operations on both the transmit and receive side. Figure 5 shows an implementation example of the architecture. All inputs and outputs to the memory interface can be registered within the IOB to minimize clock-to-out delays.
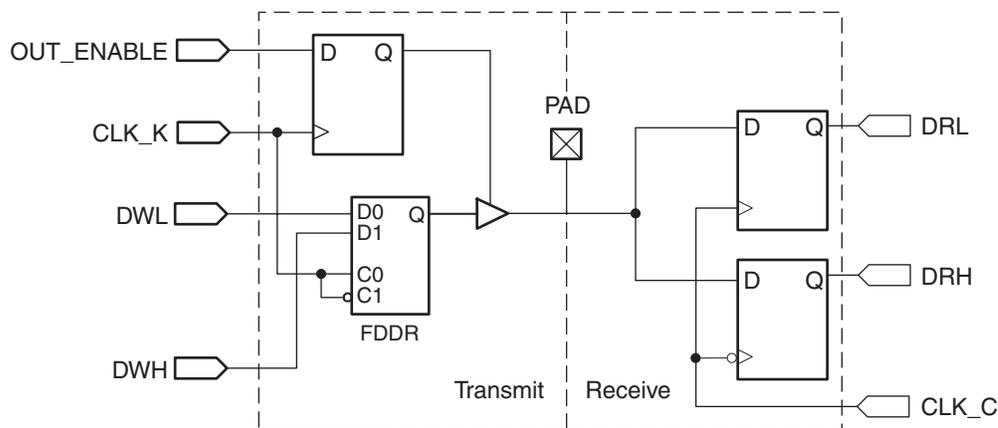
Figure 5: **I/O Implementation Example for Send and Receive Operations**

Clocking scheme

As in most synchronous designs, all control and data signals are sent from the FPGA to the memory device. Paying particular attention to the clock forwarding scheme avoids having to take into account unnecessary scrutiny of the clock-to-out timing parameters in the FPGA. Skew between clock, data, and address signals is negligible if the clock is forwarded to the memory device using the Dual Data Rate flip-flops (FDDR components) inside the FPGA I/Os. Figure 6 shows the actual clock path.
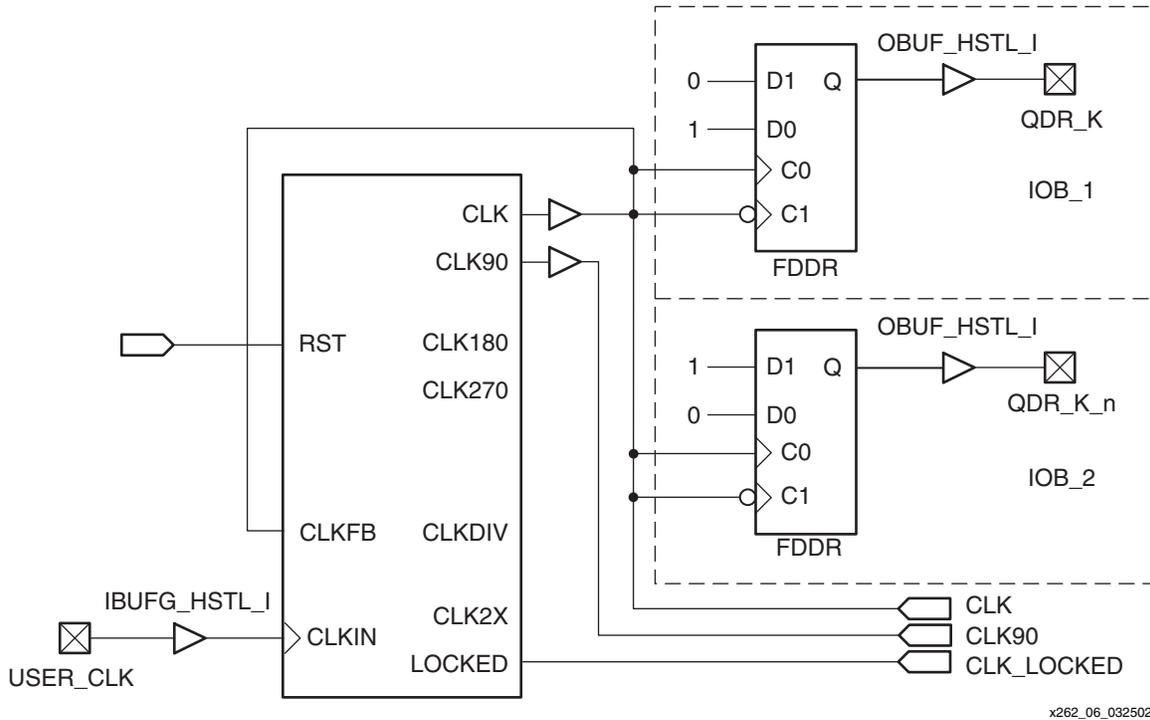

Figure 6: **Clock Forwarding Scheme**

The timing parameters, including the clock-tree skew involved in the timing budget analysis, are figured in the timing section. Duty cycle distortion depends on the clock frequency of the interface. Simulations show the duty-cycle distortion slightly exceeds the tolerance range of 5% at a starting frequency of 167 MHz and especially at higher frequencies. Note: A worst case of duty cycle distortion of 6% to 8% can be expected at 200 MHz. IBIS simulations give more accurate values. To limit board skew, the trace drawing and length must match (as much as possible) for all signals of the same length.

Using Several Devices on the Same Bus

Careful PCB layout is necessary to reduce the skew between clocks and data buses. For example, the trace length between the write data bus and the K clock should be the same. Figure 7 is an example of two devices connected.
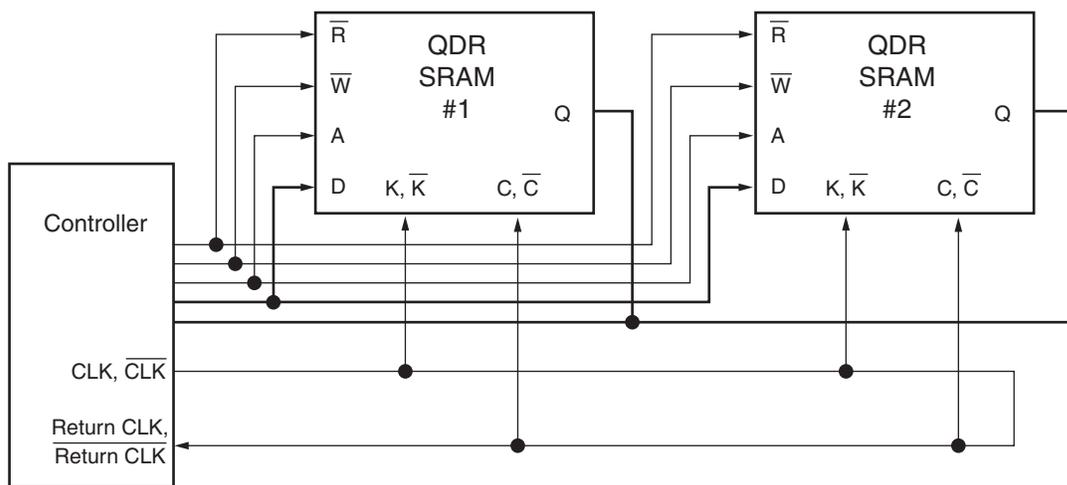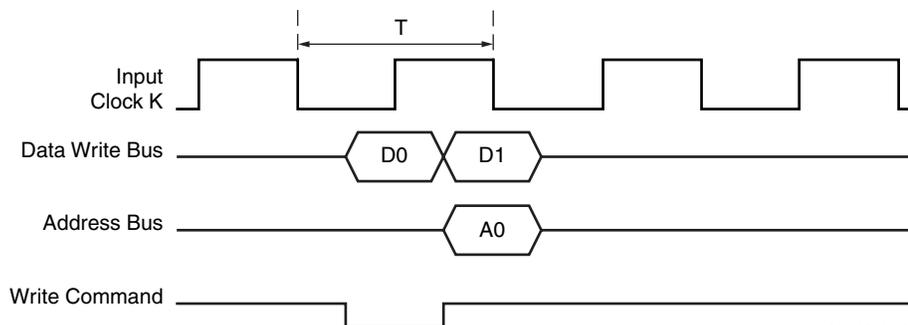

x262_07_032802

Figure 7: **Example of Connection Between one FPGA and two QDR SRAMs.**

Data path

Transmit Side

At the memory device I/Os, data should be center aligned with the clock. Figure 8 presents the basic waveforms used to write data to the memory.

x262_08_042202

Figure 8: **Write Waveforms**

When accessing a memory device, the setup and hold time needs to be considered. Figure 9 presents an example of data path implementation for a write operation.
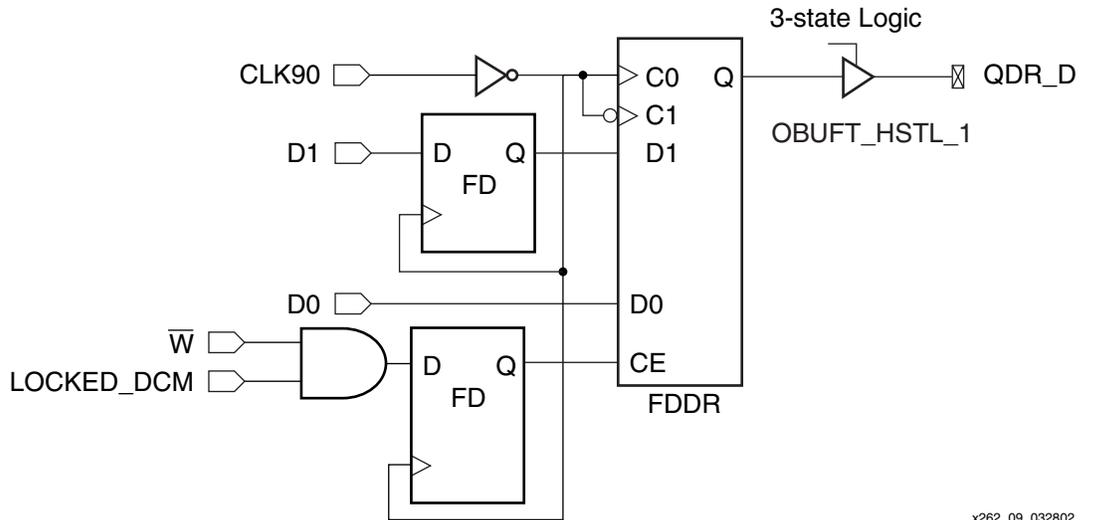
Figure 9: **Write Data Path**

Receive Side

Since the system is source synchronous, the data recapture stage uses the deskew clock coming from the memory device. The DCM is configured to produce a phase offset of the input clock to include capture data at the same time. A detailed analysis of the phase offset value is presented in the timing section. Figure 10 shows the basic waveforms used to read data to the memory.
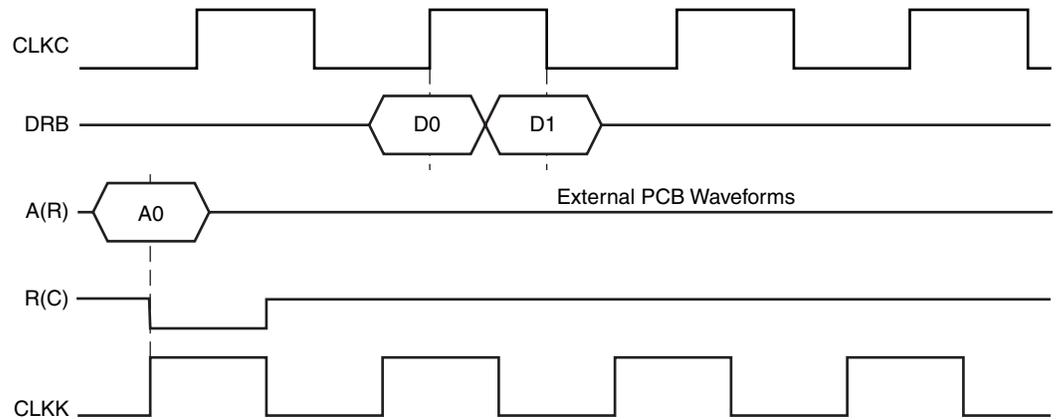
Figure 10: **Read Waveforms**

Figure 11 presents an example of read data path and Figure 12 presents the clock resources used to achieve the data capture.
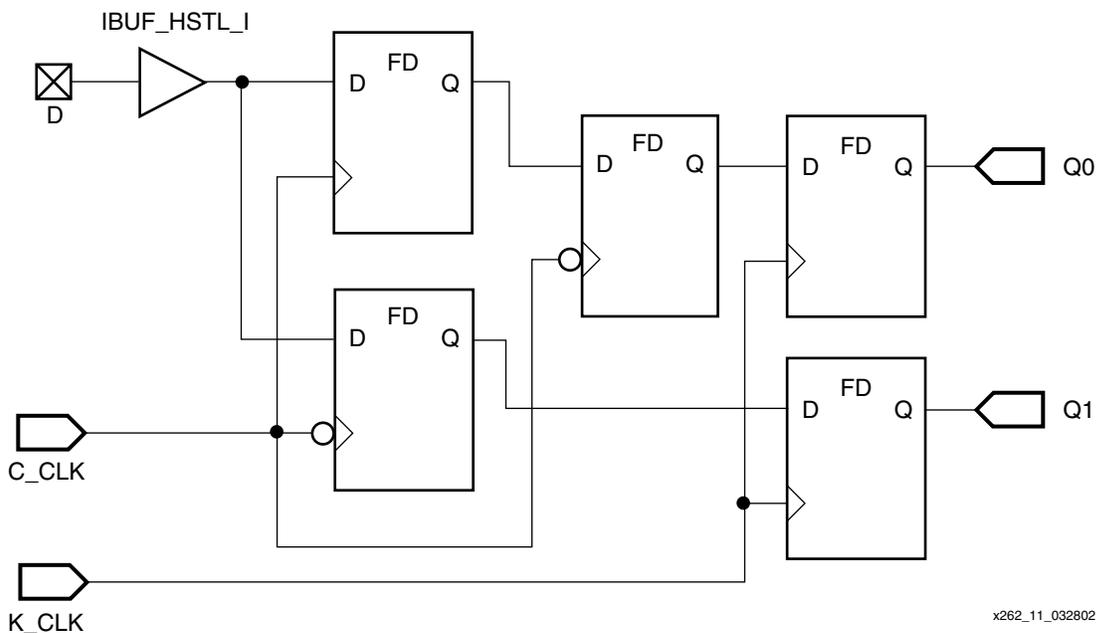
Figure 11: **Read Data Path**

The phase shift settings are not shown in Figure 12. The defined phase shift values must either be specified in the User Constraints file (as in the present reference design) or in the source files using the appropriate syntax for the synthesis tool.
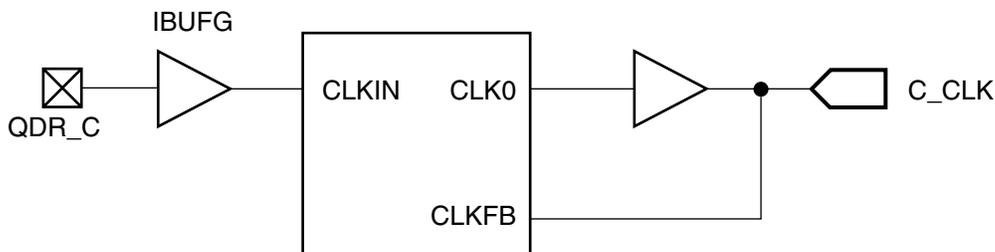
Figure 12: **Data Capture Clock Generation**

Figure 13 shows the different locations of the read waveforms in Figure 14. The clock domain change has not been detailed on Figure 13. Figure 14 also includes the clock domain change from the deskew clock to the design clock. The closer the characteristics of the length traces are between all traces, the better skew and jitter analysis.
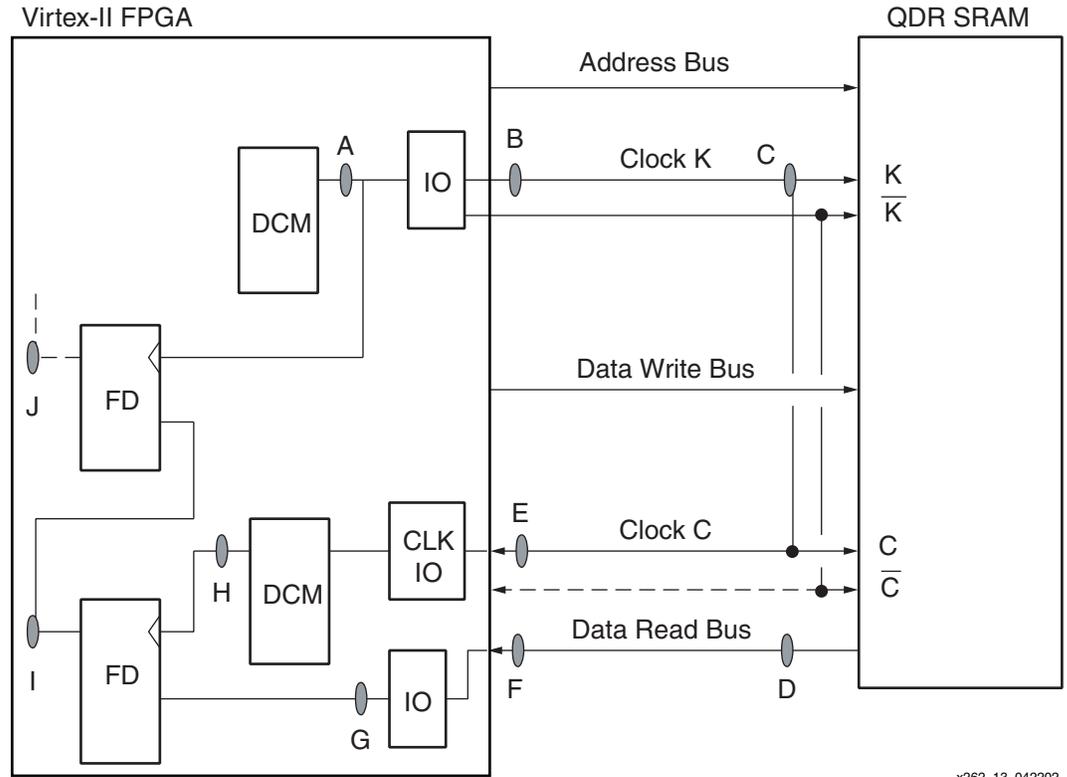
Figure 13: **System Diagram**
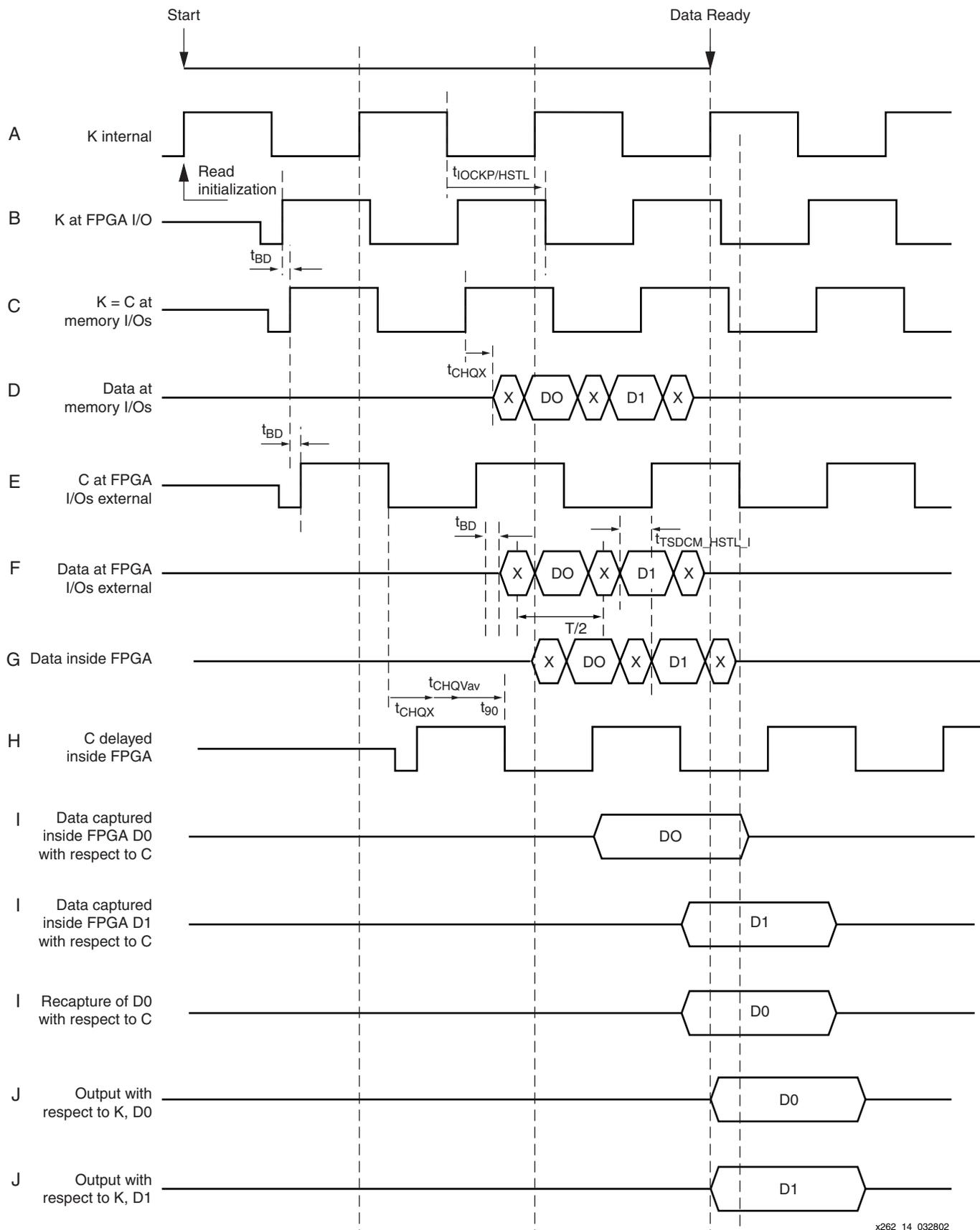
Figure 14: **Data Capture and Clock Domain Change Waveforms**

The parameter $t_{PSDCM\_HSTL\_I}$ represents the setup time, pin-to-pin number of Virtex-II devices in HSTL-I. The clock inside the FPGA is ahead with respect to the same clock outside the device. The waveforms are drawn using typical and average values; jitter or skew characteristics are not shown. The skew information is available on either the memory vendor's data sheet, in the Virtex-II data sheet, and in the **Timing Budget** section.

Address Path

The address path module depends essentially of the burst mode of the memory device. In this design, it does work in 2-word burst and works in DDR. Figure 15 presents a possible implementation.

x262_15_032802
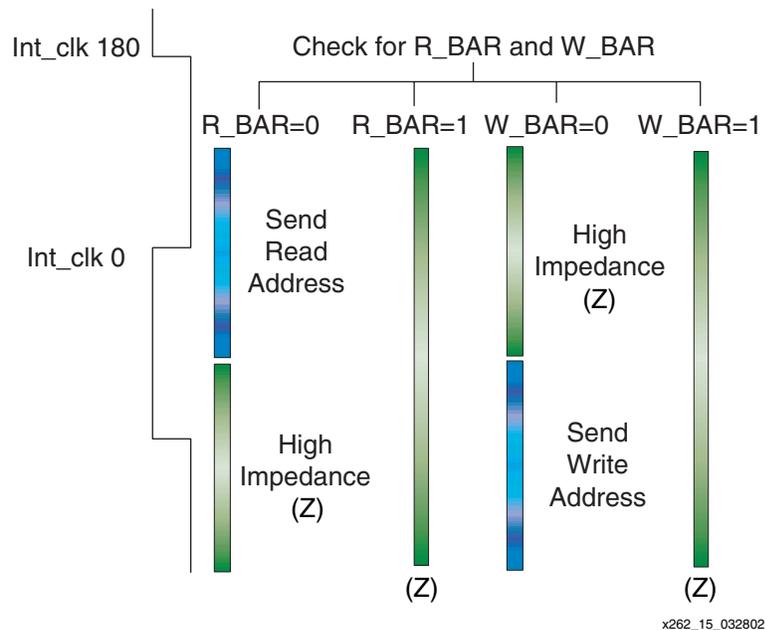
Figure 15: **2-Word Burst Address Path**

The reference design was implemented using a 9 Mb device, 512k x18. The bus width can be increased by replacing the bus width value as needed. Although the controller looks like a Single Data Rate (SDR) device from the user point of view, the external data exchange rate is DDR with an 18-bit bus width between the controller and the memory. In SDR mode, output buses at the user interface are 36-bits wide for both read and write. The number of traces on the PCB is reduced by two compared with using separate SDR bus I/O devices.

Timing Considerations

Clocking Scheme and Clock Forwarding

By forwarding the clock to the memory, the designer can disregard the FPGAs clock-to-out timing parameter. This considerably improves the speed capability of the controller. Using a source-synchronous design on the receive side enlarges the data valid window. Figure 16 shows the clocking scheme.
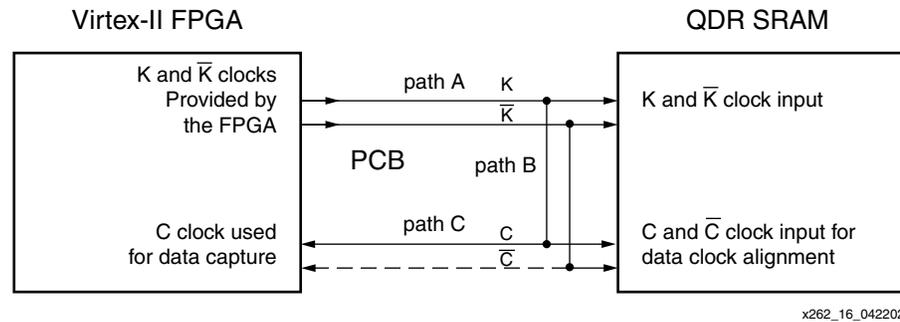
Figure 16: **Clocking Scheme**

- Path A: The clock signals are sent perfectly aligned along with the data by using the clock-forwarding scheme.
- Path B: This trace connects path A and path C directly. It should be as short as possible.
- Path C: This trace is connected on both sides.
    - On the memory side, it is connected to the C and $\overline{C}$ inputs. In this case, the memory device will send in the data with respect to this clock.
    - On the FPGA input side, the "C" deskew clock signal feeds a DCM for the clocking operation on the receiver.
    - Memory vendors recommend making route C and $\overline{C}$ inputs close together on the PCB to avoid noise issues. If $\overline{C}$ needs to be routed to the FPGA, make sure the tool routes it to a general I/O user pin for the correct termination.

Transmit and Receive Data

As the memory vendor guarantees a specific relationship between "C" and the data placement on the Read bus. This deskew clock signal is used for capturing data inside the FPGA. A clock phase shift is applied to center the deskew clock and the valid data window.

Initialization

The QDR SRAM memory behaves like other SRAMs. Special devices have an Echo Clock feature. Those devices have an on-chip embedded Delay-Locked Loop (DLL) and a delay at power up is necessary for the clock circuits to lock and for all levels to stabilize. The reference design uses a conventional QDR SRAM device without Echo Clocks. All clock signals are used to achieve the source-synchronous design. On the FPGA side, after the power-up process, it is necessary to wait for the reset and for the DCMs to lock. For both the memory device and Virtex-II device, this process should take less than one second.

Timing Budget

Transmit Side

To write to the memory, the data is sent on the Write bus in DDR mode. The data is center aligned with the K signal. The K signal is forwarded to the memory. The clock tree and package skew parameters of the FPGA will slightly reduce the width of the valid data window. Table 5 presents the different timing analysis parameters.

Table 5: **QDR x36 Transmit Side Timing Budget at a 166.67 MHz Clock Frequency**

FF896 Package Description		Worst Case Value (ps)	Typical Value (ps)
Channel-to-channel skew			
Clock tree	T_{CKSKEW}	100[1]	25[2]
Package	$T_{PKGSKEW}$	112	112
Duty cycle distortion	T_{DCD_CLK0}	140	140
Jitter		200	200
Clkout_Phase_Offset [3]		140	140
Timing budget total		692	617

Notes:
1. Clock tree skew for the entire device.
2. Clock tree skew for the specific components chosen for this particular interface, using placement constraints.
3. Only two outputs to the DCM are used for this design. In the worst case, this configuration can generate 140 ps of phase offset to the timing budget.

Table 6 is the typical timing margin budget. As shown, there is a comfortable margin in the transmit side of the FPGA.

Table 6: **Timing Margin Budget for Data Transmit**

	200 MHz QDR SRAM	167 MHz QDR SRAM
Valid data window	$2.50 - 0.62 = 1.88$ ns	$3.00 - 0.62 = 2.38$ ns
Timing budget margin	$t_{T/2} - t_{DVKH\_200} - t_{KHDX\_200} - 0.62 = 0.68$ ns	$t_{T/2} - t_{DVKH\_167} - t_{KHDX\_167} - 0.62 = 1.08$ ns

Notes:

$t_{T/2}$ = half clock period,
t_{DVKH_200} = 0.6 ns (Samsung), t_{DVKH_167} = 0.7 ns (Micron),
t_{KHDX_200} = 0.6 ns (Samsung), t_{KHDX_167} = 0.7 ns (Micron)

Receiver Side

There are two options when determining the receiver side timing budget.

- Use timing numbers listed in the Virtex-II or Virtex-II Pro data sheet. These numbers are very conservative, and make a simplistic timing budget. However, a small timing margin budget in particular conditions will not meet the timing.

- Use the timing numbers listed in the source-synchronous data sheet. These numbers are optimized and very accurate. The calculations become more difficult to evaluate the timing budget margin when more details are included. By using the source-synchronous data sheet numbers, timing is met in all cases.

Table 7 is a summary of the pin-to-pin setup and hold time parameters for conventional and source-synchronous data sheet. Continue to consult the source-synchronous data sheet for updates as the data in this table is subject to change.

Table 7: **Source-Synchronous Summary**

Virtex-II Speed Grade	-6		-5		-4		Units
	Setup	Hold	Setup	Hold	Setup	Hold	
Synchronous (HSTL)	1.98	−1.18	2.02	−1.22	2.32	−1.56	ns
Source-Synchronous	TBD	TBD	TBD	TBD	TBD	TBD	ns
Sample window Synchronous	700.00		700.00		1080.00		ps
Sample window Source-Synchronous	480[1]		500[1]		550[1]		ps

Notes:
1.   Advance numbers, please consult latest source-synchronous data sheet for current update

a.   Using Conventional Data Sheet Numbers

The Virtex-II data sheet shows a sample window of 700 ps. This parameter indicates the total sampling error of the Virtex-II DDR input registers across voltage, temperature, technology migration, package skew, or clock tree skew.

Internal clock phase shift analysis:

-   QDR SRAM memory vendors certify (worst case) the availability of data t_{CHQV} after a C edge clock.

-   The data bus is considered busy for a delay of at least t_{CHQX} on the same edge.

-   From the previous two points, data must start to be valid on the data bus some time between t_{CHQV} and t_{CHQX}. The average value is $t_{CHQVavg} = (t_{CHQV} - t_{CHQX})/2$.

-   To align C clock and data at the FPGA inputs, C is delayed by $t_{CHQX} + t_{CHQVavg}$.

-   The data is captured in the middle of the valid data window. A 90° phase shift t_{90PS} on the C clock sets the data in the middle of the theoretical valid data window. The delay is now $t_{CHQX} + t_{CHQVavg} + t_{90PS}$.

-   Center the data capture instant right in the middle of the sample window. To capture data inside the FPGA, C is delayed by:

$$t_{SAMPLE} = \left\| |t_{SETUP}| - |t_{HOLD}| \right\|$$

The total delay for the C phase shift inside the FPGA is:

$$t_{PHASESHIFT} = t_{CHQX} + t_{CHQVav} + t_{90PS} + t_{PSDCM_{HSTL}}$$

The added delay necessary on the receive side is shown in Table 8.

Table 8: **Phase Shift Value for C Clock**

Device	XC2V40 to XC2V8000		
Speed Grade	-6	-5	-4
100 MHz	6.58	6.62	6.92
133 MHz	5.96	6.00	6.3
167 MHz	5.33	5.37	5.67
200 MHz	4.83	4.87	NC[1]

Notes:
1.   Due to the I/O switching capabilities of these high-speed devices, do not use Virtex-II -4 devices for 200 MHz clock interfaces

b. Using the source-synchronous timing numbers:

With a source-synchronous data sheet, the width of the sample window is reduced by shifting the setup and hold time values. The Virtex-II source-synchronous data sheet shows a sample window of t_{SAMP} = 500 ps. This parameter indicates the total sampling error of the Virtex-II DDR input registers across voltage, temperature, or technology migration. The test measurement methodology uses the DCM to capture the DDR input registers' edges of operation.

These measurements include:

- CLK0 and CLK180 DCM jitter
- Worst-case duty-cycle distortion
- DCM accuracy (phase offset)
- DCM phase shift resolution

The measurements do not include package or clock tree skew. Table 9 presents an analysis using two different packages for an interface for up to 36 data bus width devices. The speed grade of the FPGA has an insignificant impact on these numbers.

Table 9: **QDR x36 Receive Side Timing Budget at a 166.67 MHz Clock Frequency**

FF896 Package Description		Worst Case Value (ps)	Typical Value (ps)
Sample window	T_{SAMP}	500	500
Channel-to-channel skew			
Clock tree	T_{CKSKEW}	100[1]	25[2]
Package	$T_{PKGSKEW}$	112	112
Timing budget total		712	637

Notes:
1. Clock tree skew for the entire device.
2. Clock tree skew for the specific components chosen for this particular interface, using placement constraints.

The memory vendor specifies the valid data window.

$t_{VDW167} = t_{T/2} - t_{CHQZ\_167} + t_{CHQX\_167}.$

From a timing point of view, the results on the transmitter side of the interface should never be a problem (Table 10).

Table 10: **Typical Timing Margin Budget for Data Receive**

	200 MHz QDR SRAM	167 MHz QDR SRAM
Valid data window	2.50 − 0.64 = 1.86 ns	3.00 − 0.64 = 2.36 ns
Timing margin budget	$t_{T/2} - t_{CHQZ\_200} + t_{CHQX\_200} - 0.64 = 0.66$ ns	$t_{T/2} - t_{CHQZ\_167} + t_{CHQX\_167} - 0.64 = 1.06$ ns

Notes:
$t_{T/2}$ = half clock period,
t_{CHQZ_200} = 2.2 ns (Samsung), t_{CHQZ_167} = 2.5 ns (Micron),
t_{CHQX_200} = 1.0 ns (Samsung), t_{CHQX_167} = 1.2 ns (Micron)

It is still possible to use this interface with a 200 MHz clock for 4-word burst high-speed devices. In this case, using a Virtex-II -5 speed grade device or faster will ensure sufficient HSTL I/Os switching performance.

Valid data window for data sent by the memory device:

$t_{VDW200} = t_{T/2} - t_{CHQZ\_200} + t_{CHQX\_200} = 1.30$ ns.

There is still a positive margin of 0.66 ns.

Since the valid data window is enlarged in QDR SRAM devices, there is even more latitude when interfacing to the next generation of memory devices.

1. System level timing analysis

   - The system level timing analysis completes the data capture analysis while including the parameters involved in the system design. These parameters depend essentially on the PCB design, the board level skew and jitter. The analysis must ensure the PCB timing requirements are smaller than the available timing margin.

2. Internal clock phase shift analysis:

   The method applied in the previous section requires known values for setup and hold time for Virtex-II source-synchronous designs. If the values are not available, then a bench calibration is needed to set the value of the clock phase shift on the receive side. The designer can set the starting point of the phase shift using the values in Table 8, page 17.

Clock Domain Change

The first step in the receive side is capturing the data. Data capture is done with respect to C clock and the controller has the internal K clock as a reference. It is necessary to realign the received data to K clock.

The current controller uses D flip-flops to make the clock domain change. This is one solution among possible other alternatives: FIFO lists using D flip-flops, SRL16, or block RAMs. Although appropriate for particular applications, these alternate solutions are not developed in this reference design. They require an additional study of the speed characteristics of internal components used during implementation.

The trace board delay timing, t_{BD} parameter is decisive in this analysis. The relationship between K and internal C clock signals is $t_{CDC} = t_{IOCKP\ /HSTL} + t_{PHASESHIFT} + 2 \times t_{BD}$, modulo the clock period. The board delay for all transmit and receive signals is described as $2 \times t_{BD}$.

For example, in the current design, t_{CDC} is roughly equal to $(1.6 + (2 \times t_{BD}))$ ns for a period of 6 ns:

$$t_{CDC} = (2.99 + 0.21) + (1.20 + 0.68 + 1.50 + 1.00) + 2 \times t_{BD}$$

Data is caught on the 90° phase shift with respect to K for D0, and 270° phase shift for D1. Both data registers are aligned in SDR mode (acting like a pipeline) due to the second level of D flip-flops. Since this value will vary, it becomes necessary to rearrange the clocks of this stage to complete the clock domain change properly. As the designer has a precise idea of the value of $t_{PHASESHIFT}$, the unknown parameter is t_{BD}. Bench calibration is one way to determine this value.

Xilinx recommends using the CLKFX output of the DCM if the margin to achieve D0 and D1 alignment is too tight for reducing the clock frequency. This method is useful while using block RAM when the controller is running at a high frequency.

Timing Analysis Summary

The worst case numbers, used in this example, can be substantially different than the best case. The designer should readjust these values and the clocking scheme (especially in the **Clock Domain Change**). Two levels of performance can be achieved using the general type of design developed in this application note:

- Conventional synchronous design is the easiest type of design and it allows the controller to run at 133 MHz. Only one DCM is necessary on the transmit side. It can be used on the receiver side to capture data.

- Source-synchronous design is developed in this application note. It allows a 166 MHz interface for a 2-word burst memory device and a 200 MHz interface for a 4-word burst device. Using a high-speed grade Virtex-II device, it can be done with a comfortable margin.

In any case, a detailed timing analysis is the first step to designing a high-performance memory interface. The PCB characteristics, including new speed files, must be considered to ensure timing budget margin availability.

Reference Design Notes

The reference design is available in VHDL and Verilog at:
ftp://ftp.xilinx.com/pub/applications/xapp/xapp262.zip.

Reference Design Description

- The top-level architecture is described in QDR_burst_2_body.
- The address path is described in Address_burst_2.
- C_Generator is the entity that provide the clock for capturing the data on the receive side.
- Clk_generator is the entity that implements the clock generation for the memory device and the internal logic.
- The data path on the receive side and the clock domain change takes place in Read_burst_2.
- The data path on the transmit side is implemented in Write_burst_2 module.
- A testbench file for the reference design is provided in the package and the name of the architecture is QDR_Ctrl_TB.
- Another test bench is provided to check the direct data placement and exchange with the provided HDL model for QDR SRAMs.
- The Micron Technology MT54V512H18A HDL model.
- The package also contains scripts for simulation, and a .ucf constraints file.

Reference Design Notes and Advise

- The functional simulations use ModelTech ModelSim 5.6. The synthesis uses Synplicity Synplify 7.0 and mapping, place and route with Xilinx ISE 4.2i.
- To reduce the skew on the I/O flip-flops clock tree, Xilinx recommends setting the buses on adjacent pads in the same I/O bank. In this design, all signals are on Bank 6 and Bank 7. If the size of the bus increases, it may be necessary to follow the SSO guideline and insert other signal in the bus implementation.

  Also, the system I/O rules must be followed to achieve mapping. You will note that in our case, the input for deskew clock is set on bank 5.
- The current design should be able to run at a much higher speed than the FPGAs I/Os can handle (332 MHz with a -2 level of effort and few timing constraints). Nevertheless, depending on the design changes or on the resources used for a FIFO buffer, the speed may decrease significantly. Set the appropriate timing constraints to the critical path in the interface design.
- IBIS or H-Spice simulations are strongly recommended to highlight potential signal integrity problems and check the Virtex-II I/Os switching speed.

Abbreviations

Table 11: **Glossary of Abbreviations**

Abbreviation	Description
BST	Boundary Scan Test (IEEE 1149.9)
CLB	Configurable Logic Block
DAR	Double Address Rate
DCM	Digital Clock Manager
DDR	Double Data Rate
DLL	Delay-Locked Loop
FIFO	First In, First Out
GND	Ground
HSTL	High-Speed Transceiver Logic
JEDEC	Joint Electron Device Engineering Council
JTAG	Join Test Action Group
PCB	Printed Circuit Board
PLL	Phase-Locked Loop
QDR SRAM	Quad Data Rate Synchronous Static Random Access Memory
SAR	Single Data Rate
SSO	Simultaneous Switching Outputs
TAP	Test Access Port

References

The following documents are recommended:

1. Micron Technology Inc., QDR SRAM Technical Guide, Technical Note TN-54-01, Rev. 4/01, 2001, **http://www.micron.com**

2. Micron Technology Inc., 9Mb QDR SRAM 2-Word Burst, MT54V512H18A, Revision 3/00, 2000, **http://www.micron.com**

3. Cypress Semiconductor Corporation, 9-Mb Pipelined SRAM with QDR Architecture CY7C1302V25, March 28, 2000, **http://www.cypress.com**

4. Xilinx Inc., Virtex-II Platform FPGA Handbook, Revision 1.3, December 2001, **www.xilinx.com**.

5. Xilinx Inc., Virtex-II Platform FPGA Datasheet, DS031-1 (rev 1.7), DS031-2 (Rev 1.9), DS031-3 (Rev 2.0), DS031-4 (Rev 1.6), November 2001, **www.xilinx.com**.

6. Xilinx Inc., Using the Virtex SelectI/O™ Feature, Application Note XAPP133, Rev 2.5, September 7, 2000, **http://www.xilinx.com/xapp/xapp133.pdf**.

Also more detailed information about QDR SRAM can be found on the Internet web site of the QDR consortium, at **http://www.samsungelectronics.com**. Those memory devices are proposed by:

- Cypress Semiconductor Corporation at **http://www.cypress.com**
- HITACHI, Ltd. at **http://semiconductor.hitachi.com/memory.html**
- Integrated Device Technology, Inc. at **http://www.idt.com**
- Micron Technology, Inc at **http://www.micron.com/**

- NEC Corporation at **http://www.ic.nec.co.jp/memory/index_e.html**
- SAMSUNG Electronics Company, Inc. at **http://www.samsungelectronics.com**

Conclusion

The application note guidelines help to achieve the best performance when interfacing QDR devices with Virtex-II FPGAs. Virtex-II devices support up to DDR400 QDR interfaces as shown in the timing analysis section. The current reference design covers building a Virtex-II based high-performance controller. It also gives a starting point for building a QDR II SRAM interface. The reference design package provides design resources for DDR I/Os, DCM, System I/O interfaces to meet performance expectations.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
01/15/01	1.0	Initial Xilinx release.
02/27/02	2.0	Overall revisions to code and document.
03/28/02	2.1	Changes in many figures and tables.
04/24/02	2.2	Changes to figures and tables, added Verilog support.