Interfacing RLDRAM II with Stratix II, Stratix,& Stratix GX Devices

November 2005, ver. 3.1 Application Note 325

Introduction

Reduced latency DRAM II (RLDRAM II) is a DRAM-based point-to-point memory device designed for communications, imaging, and server systems requiring high density, high memory bandwidth, and low latency. The fast random access speeds in RLDRAM II devices make them a viable alternative to SRAM devices at a lower cost.

There are two types of RLDRAM II devices: common I/O (CIO) and separate I/O (SIO). CIO devices share a single data I/O bus which is similar to the double data rate (DDR) SDRAM interface. SIO devices, with separate data read and write buses, have an interface similar to SRAM.

Compared to DDR SDRAM, RLDRAM II has simpler bank management and lower latency inside the memory. RLDRAM II devices are divided into eight banks instead of the typical four banks in most memory devices, providing a more efficient data flow within the device. RLDRAM II offers up to 2.4 Gigabytes per second (Gbps) aggregate bandwidth.

Stratix® II devices in the -3 speed grade support RLDRAM II at up to 300 MHz and 600 Megabits per second (Mbps). Stratix and Stratix GX devices in the -5 speed grade in the flip-chip package (except EP1S60 and EP1S80 devices) support RLDRAM II at up to 200 MHz and 400 Mbps.

Table 1 shows the RLDRAM II support in Stratix II devices.

Table 1. RLDRAM II Maximum Clock Rate Support in Stratix II Devices *Notes (1), (2), (3)*

Speed Grade	DLL-Based Implementation	PLL-Based Implementation
-3	300 MHz	200 MHz
-4	250 MHz *(4)*	175 MHz *(5)*
-5	200 MHz	175 MHz *(5)*

Notes to Table 1:

(1) This analysis is based on the EP2S60F1020. Ensure you perform a timing analysis for your chosen FPGA.

(2) These numbers are from Quartus® II software, version 5.1. Altera recommends using the latest version of the Quartus II software for your design.

(3) These numbers apply to both commercial and industrial devices.

(4) Although there are no 250-MHz RLDRAM II devices, you can underclock 300-MHz RLDRAM II devices at 250 MHz.

(5) This is the lowest frequency a RLDRAM II device can operate.

Table 2 shows the Stratix EP1S10 through EP1S40 devices, and Stratix GX EP1SGX10 through EP1SGX40 devices.

Table 2. RLDRAM II Maximum Clock Frequency Support in EP1S10 through EP1S40 & EP1SGX10 through EP1SGX40 Devices *Note (1)*

Speed Grade	DLL-Based Implementation *(2)*
-5	200 MHz *(3)*
-6	N/A *(4)*
-7	N/A *(4)*

Notes to Table 2:
(1) Stratix EP1S60 and EP1S80 do not support RLDRAM II.
(2) Stratix devices do not offer PLL-based implementation RLDRAM II interfaces.
(3) This frequency is only supported in flip-chip packages and is based on read capture and write timing analysis. You have to perform your own resynchronization timing analysis.
(4) RLDRAM II memory is not supported in this speed grade.

This application note describes an example RLDRAM II interface with Stratix II, Stratix, and Stratix GX devices. It discusses the electrical and timing analysis for the interface and also describes the Altera Stratix series memory board and lists the general board guidelines when interfacing RLDRAM II memory with a Stratix-series device. In general, this application note is applicable for both RLDRAM II CIO and SIO devices.

This application note focuses on the timing analysis that you need to create a successful interface. In order to do so, you need to use the Altera recommended data path, available from the RLDRAM II Controller MegaCore® function from the Quartus II MegaCore Intellectual Property (IP) Library CD. After compiling the controller and ensuring that the design meets the targeted core fmax, you need to perform the following timing analysis: read capture, write capture, command and address, bus turnaround, and round trip delay (for RLDRAM II interfaces in Stratix devices).

Use this application note together with the *External Memory Interfaces* chapter of the *Stratix II Device Handbook* or *Stratix Device Handbook*.

RLDRAM II Overview

RLDRAM II uses a DDR scheme, performing two data transfers per clock cycle. RLDRAM II CIO devices use the bidirectional data pins (DQ) for both read and write data, while RLDRAM II SIO devices use D pins for write data (input to the memory) and Q pins for read data (output from the memory). Both types use two pairs of uni-directional free-running

clocks. The memory uses DK and DK# pins during write operations, and generates QK and QK# pins during read operations. In addition, RLDRAM II uses the system clocks (CK and CK# pins) to sample commands and addresses and generate the QK and QK# read clocks. Address ports are shared for write and read operations.

The RLDRAM II SIO devices are available in ×9 and ×18 data bus width configurations, while the RLDRAM II CIO devices are available in ×9, ×18, and ×36 data bus width configurations. RLDRAM II CIO devices require an extra cycle for bus turnaround time for switching read and write operations.

Write and read operations are burst oriented and all the data bus width configurations of RLDRAM II support burst lengths of two and four. In addition, RLDRAM II devices with data bus width configurations of ×9 and ×18 also support burst length of eight.

The read latency is the time between when the read command is clocked into the memory and the time data is presented at the memory pins. There is a similar latency for write operations called the write latency. The write latency is equal to the read latency plus one clock cycle. The RLDRAM devices have up to three programmable configuration settings that determine the row cycle times, read latency, and write latency of the interface at a given frequency of operation.

RLDRAM II devices use either the 1.5-V HSTL or 1.8-V HSTL I/O standard. Altera recommends the 1.8-V HSTL I/O standard for maximum performance. Each RLDRAM II device is divided into eight banks, where each bank has a fixed number of rows and columns. Only one row per bank is accessed at a time. The memory (instead of the controller) controls the opening and closing of a row, which is similar to an SRAM interface.

RLDRAM II also offers programmable impedance output buffers and on-die termination. The programmable impedance output buffers are for impedance matching and are guaranteed to produce 25- to 60-Ω output impedance. The on-die termination is dynamically switched on during read operations and switched off during write operations. Since Stratix II, Stratix, and Stratix GX devices do not offer a similar dynamic on-die termination, perform an IBIS simulation to observe the effects of this dynamic termination on your system. IBIS simulation can also show the effects of different drive strengths, termination resistors, and capacitive loads on your system.

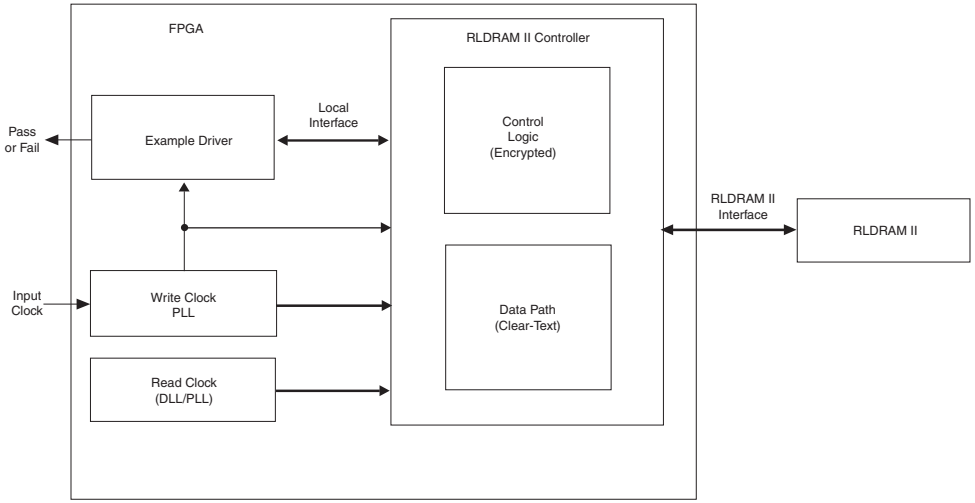
For more information on the RLDRAM II specifications, go to **www.rldram.com**.

Interface Description

This section provides a detailed description of the interface between the FPGA and the RLDRAM II devices. It describes the interface signals and how Altera FPGA pins should be configured to meet the strict RLDRAM II electrical and timing requirements, listing the number of possible interfaces based on the number of DQS and DQ pins available in the FPGA. In addition, this section also describes the architecture of the interface between the FPGA and the RLDRAM II memory.

Understanding how complicated the interface can be, Altera offers a complete solution that will create the memory controller within minutes. The Altera RLDRAM II controller MegaCore function allows you to use the Altera recommended data path whether or not you are using the Altera core. When you use this provided data path, you are ensured a working system as the RLDRAM II IP tool bench constraints your interface pins and data path logic for optimal operation. Figure 1 shows the block diagram for the FPGA to RLDRAM II interface.

Figure 1. Block Diagram for the FPGA to RLDRAM II Interface *Note (1)*



Notes to Figure 1:

(1) You can either use the DLL-based implementation (for top and bottom I/O banks) or the PLL-based implementation (for any I/O bank) for your RLDRAM II interface with Stratix II devices. Only the DLL-based implementation is available in Stratix devices.

Interface Signals

Table 3 shows a summary of the RLDRAM II interface pins and how to connect them to Stratix II, Stratix, and Stratix GX devices.

Table 3. RLDRAM II Interface Pins Summary *Note (1)*

Pins	Description	Stratix II Device Pin Utilization (DLL-Based Implementation)	Stratix II Device Pin Utilization (PLL-Based Implementation)	Stratix or Stratix GX Device Pin Utilization (DLL-Based Implementation)
DQ	RLDRAM II CIO bidirectional read and write data	DQ	User I/O pin *(2)*	DQ
D	RLDRAM II SIO unidirectional write data	User I/O pin	User I/O pin	User I/O pin
Q	RLDRAM II SIO unidirectional read data	DQ	User I/O pin *(2)*	DQ
DK	Unidirectional write clock for the RLDRAM II device to sample data	User I/O pin *(3)*	User I/O pin *(3)*	External clock buffer *(4)*
DK#	Unidirectional write clock for the RLDRAM II device to sample data	User I/O pin *(3)*	User I/O pin *(3)*	External clock buffer *(4)*
QK	Unidirectional read clock from the RLDRAM II device	DQS	PLL dedicated clock input	DQS
QK#	Unidirectional read clock from the RLDRAM II device	Not used *(5)*	Not used *(5)*	Not used *(5)*
CK	System clock for the RLDRAM II device to sample address and command	User I/O pin *(3)*	User I/O pin *(3)*	External clock buffer *(4)*
CK#	System clock for the RLDRAM II device to sample address and command	User I/O pin *(3)*	User I/O pin *(3)*	External clock buffer *(4)*
QVLD *(6)*	Optional read data valid pin	DQVLD	User I/O pin *(2)*	User I/O pin
DM	Write data mask pin	User I/O pin	User I/O pin *(2)*	User I/O pin
All other	Address, control, etc.	User I/O pin	User I/O pin	User I/O pin

Notes to Table 3:

(1) Stratix and Stratix GX devices do not offer PLL-based RLDRAM II interface implementation.
(2) Refer to the Stratix II pin table for the recommended pins for PLL-based (non-DQS) implementation.
(3) Use the same pin to drive CK and DK signals and another pin for CK# and DK# signals. You can use two different pins if Quartus II reported skew between these pins meet the RLDRAM II device's t_{CKDK} specification.
(4) In order to meet the RLDRAM II t_{CKDK} and input slew rate specification, Stratix devices requires external clock buffer for the CK/CK# and DK/DK# signals generation.
(5) Stratix II, Stratix, and Stratix GX devices do not use QK# signals for read operation.
(6) Quartus II labels this pin as DQ pins in the top and bottom I/O banks.

Clock Signals

RLDRAM II devices use CK and CK# signals to clock the command and address bus in single data rate (SDR). There is one pair of CK and CK# pins per RLDRAM II device.

Instead of a strobe, RLDRAM II devices use two sets of free-running differential clocks to accompany the data. The DK and DK# clocks are the differential input data clocks used during writes while the QK or QK# clocks are the output data clocks used during reads. Even though QK and QK# signals are not differential signals according to the RLDRAM II data sheets, Micron treats these signals as such for their testing and characterization. Each pair of DK and DK# or QK and QK# clocks are associated with either 9 or 18 data bits.

The exact clock-data relationships are as follows:

- For ×36 data bus width configuration, there are 18 data bits associated with each pair of write and read clocks. So, there are two pairs of DK and DK# pins and two pairs of QK or QK# pins.
- For ×18 data bus width configuration, there are 18 data bits per one pair of write clocks and nine data bits per one pair of read clocks. So, there is one pair of DK and DK# pins, but there are two pairs of QK and QK# pins.
- For ×9 data bus width configuration, there are nine data bits associated with each pair of write and read clocks. So, there is one pair of DK and DK# pins and one pair of QK and QK# pins each.

QK pins are connected to the DQS pins in Stratix II, Stratix, and Stratix GX devices. Even though the QK signal is an input-only signal to the Stratix II, Stratix, and Stratix GX device, you need to configure it as bidirectional and tie the OE pin to ground to achieve the input-only operation in the Quartus II software. The QK# pins are not used to capture data in Stratix II, Stratix, and Stratix GX devices and can be left unconnected.

There are t_{CKDK} timing requirements for skew between CK and DK or CK# and DK#. For optimal performance, use the same pair of user I/O pins to generate CK and DK or CK# and DK# (using double data registers) in Stratix II devices to minimize skew between these signals. When interfacing RLDRAM II SIO devices with Stratix II devices, you should place the CK and CK# and DK and DK# signals in the same I/O bank as the D signals to minimize skew between these signals. You can use two different pins for CK and DK (and similarly CK# and DK#) if the Quartus II reported skew (the difference between the t_{CO} times) meets the memory's t_{CKDK} requirement. Use an external clock buffer to generate CK,

CK#, DK, and DK# when interfacing RLDRAM II devices with Stratix or Stratix GX devices to meet the required RLDRAM II device input slew rate specifications.

Make sure that the pin pairs for CK and CK# (and DK and DK#) have matched t_{CO}. You can use any unused DQS and DQS# pins for interfaces on top and bottom I/O banks or use LVDS pin-pair for interfaces on the side I/O banks.

Due to the loads on these I/O pins, the maximum frequency you can achieve depends on the number of RLDRAM II devices you are connecting to the Stratix II, Stratix, or Stratix GX device. Perform SPICE or IBIS simulations to analyze the loading effects of the pin-pair on multiple RLDRAM II devices.

Data, DM & QVLD Signals

When interfacing with a CIO device, connect the DQ pins with the Stratix II, Stratix, or Stratix GX device DQ pins. For DLL-based implementation, use the ×8 and ×9 DQS and DQ groups in Stratix II devices and ×16 DQS and DQ groups in Stratix and Stratix GX devices for RLDRAM II devices with data bus width configurations of ×9 and ×18. Use ×16 and ×18 DQS and DQ groups in Stratix II devices and ×32 DQS and DQ groups in Stratix and Stratix GX devices for RLDRAM II devices with data bus width configuration of ×36. This is because there are nine data bits associated with one QK pin in RDLRAM II devices in ×9 and ×18 data bus width configuration, and there are 18 data bits associated with one QK pin in RDLRAM II devices in ×36 data bus width configuration.

The DQS and DQ groups in Stratix II, Stratix, and Stratix GX devices indicate how many DQ pins one DQS pin can drive. For example, a ×8 DQS and DQ group means that the DQS pin can drive up to eight DQ pins in that group.

If you only use eight bits of the RLDRAM II device with data bus width configurations of ×9 and ×18, then you can use the ×8 DQS and DQ groups in the Stratix or Stratix GX devices. Similarly, if you are only using 16 bits of the RLDRAM II devices with data bus width configuration of ×36, you can use the Stratix or Stratix GX device's ×16 DQS and DQ groups.

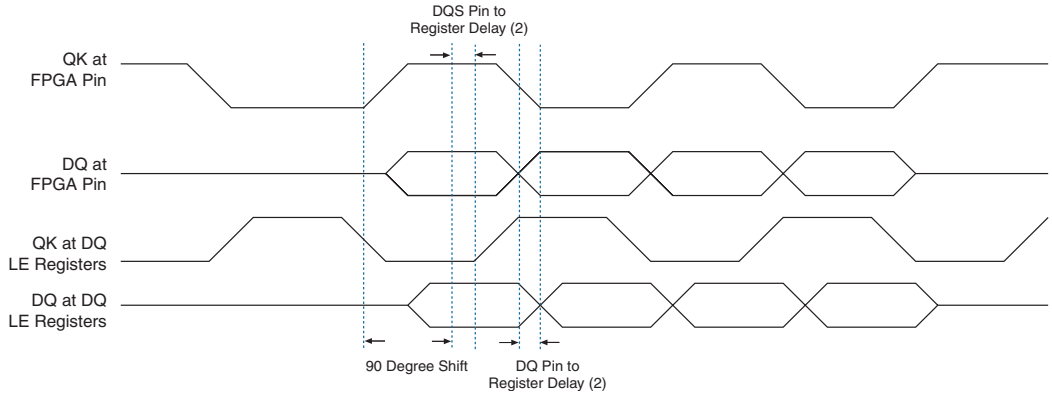
For PLL-based implementation, refer to the Stratix II pin tables for the available DQS/DQ groups in the device.

When interfacing with RLDRAM II SIO devices, the memory device's Q ports must be connected to the DQ pins of the Stratix II device in a similar fashion as when interfacing with RLDRAM II CIO devices. You

can use any of the Stratix II, Stratix, or Stratix GX user I/O pins as outputs to drive the D ports of the RLDRAM II SIO device. Altera recommends the D, CK, CK#, DK, and DK# pins be in the same I/O bank to minimize skew between these signals.

The write data is center-aligned with the DK and DK# clocks while the read data is edge-aligned with the QK or QK# clocks (see Figures 2 and 3). The memory controller shifts the DK or DK# signal to center align the DQ and DK or DK# signal during a write and to shift the QK signal during a read, so that read data (DQ or Q signals) and QK clock is center-aligned at the capture register. Stratix II, Stratix, and Stratix GX devices use dedicated DQS phase-shift circuitry to shift the incoming QK signal during reads and use a PLL to center-align the DK and DK# signals with respect to the DQ signals during writes.

Figure 2. DQ & QK Relationship During a RLDRAM II Read *Note (1)*

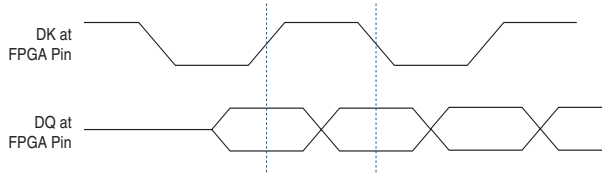


Notes to Figure 2:

(1) This is an example of a 90° shift. The required phase shift for your system should be based on your timing analysis and may not be 90°.

(2) The delay from the QK pin to the register and from the DQ pin to the register can be configured to minimize additional skew between the two signals at the IOE register in Stratix II devices. In Stratix and Stratix GX devices, the delay from the QK pin to the register may not match the delay from the DQ pin to the register.

Figure 3. DQ & DK Relationship During RLDRAM II Write



Tables 4 through 6 shows the number of DQ and DQS groups supported in each Stratix II, Stratix, and Stratix GX device density and package combination. Each column denotes the maximum number of DQ pins that can be driven by one DQS pin per device. These pins are used for DLL-based implementation. Refer to the Stratix II pin tables for DQS/DQ groups used in PLL-based implementation.

Table 4. DQS & DQ Bus Mode Support in Stratix II

Device	Package	Number of ×4 Groups *(1)*	Number of ×8/×9 Groups	Number of ×16/×18 Groups	Number of ×32/×36 Groups
EP2S15	484-pin FineLine BGA	8	4	0	0
	672-pin FineLine BGA	18	8	4	0
EP2S30	484-pin FineLine BGA	8	4	0	0
	672-pin FineLine BGA	18	8	4	0
EP2S60	484-pin FineLine BGA	8	4	0	0
	672-pin FineLine BGA	18	8	4	0
	1,020-pin FineLine BGA	36	18	8*(2)*	4
EP2S90	484-pin Hybrid FineLine BGA	8	4	0	0
	780-pin FineLine BGA	18	8	4	0
	1,020-pin FineLine BGA	36	18	8*(2)*	4
	1,508-pin FineLine BGA	36	18	8*(2)*	4
EP2S130	780-pin FineLine BGA	18	8	4	0
	1,020-pin FineLine BGA	36	18	8*(2)*	4
	1,508-pin FineLine BGA	36	18	8*(2)*	4
EP2S180	1,020-pin FineLine BGA	36	18	8*(2)*	4
	1,508-pin FineLine BGA	36	18	8*(2)*	4

Notes to Table 4:

(1) This mode is not used for RLDRAM II interfaces.

(2) There are two extra ×8 and ×9, DQS and DQ groups in addition to the listed numbers: one in I/O bank 4 and one in I/O bank 7.

Table 5. DQS & DQ Bus Mode Support in Stratix Devices *Note (1)*

Device	Package	Number of ×8 Groups *(2)*	Number of ×16 Groups *(3)*	Number of ×32 Groups *(4)*
EP1S10	672-pin BGA 672-pin FineLine BGA	12 *(5)*	0	0
	484-pin FineLine BGA 780-pin FineLine BGA	16 *(6)*	0	4
EP1S20	484-pin FineLine BGA	18 *(7)*	7 *(8)*	4
	672-pin BGA 672-pin FineLine BGA	16 *(6)*	7 *(8)*	4
	780-pin FineLine BGA	20	7 *(8)*	4
EP1S25	672-pin BGA 672-pin FineLine BGA	16 *(6)*	8	4
	780-pin FineLine BGA 1,020-pin FineLine BGA	20	8	4
EP1S30	956-pin BGA 780-pin FineLine BGA 1,020-pin FineLine BGA	20	8	4
EP1S40	956-pin BGA 1,020-pin FineLine BGA 1,508-pin FineLine BGA	20	8	4
EP1S60 *(9)*	956-pin BGA 1,020-pin FineLine BGA 1,508-pin FineLine BGA	20	8	4
EP1S80 *(9)*	956-pin BGA 1,508-pin FineLine BGA 1,923-pin FineLine BGA	20	8	4

Notes to Table 5:

(1) See the *Using Selectable I/O Standards in Stratix & Stratix GX Devices* chapter of the *Stratix Device Handbook, Volume 2* for V_{REF} guidelines.

(2) This mode can be used for RLDRAM II ×9 and ×18 data bus width configuration if parity is not used (i.e., the design only uses 8 of the 9 data pins).

(3) This mode is for RLDRAM II ×9 and ×18 data bus width configuration. It can also be used for ×36 data bus width configuration if only 16 data pins are used.

(4) This mode is for RLDRAM II ×36 data bus width configuration.

(5) These packages have six groups in I/O banks 3 and 4 and six groups in I/O banks 7 and 8.

(6) These packages have eight groups in I/O banks 3 and 4 and eight groups in I/O banks 7 and 8.

(7) This package has nine groups in I/O banks 3 and 4 and nine groups in I/O banks 7 and 8.

(8) These packages have three groups in I/O banks 3 and 4 and four groups in I/O banks 7 and 8.

(9) These devices do not support RLDRAM II devices.

Table 6. DQS & DQ Bus Mode Support in Stratix GX Devices *Note (1)*

Device	Package	Number of ×8 Groups *(2)*	Number of ×16 Groups *(3)*	Number of ×32 Groups *(4)*
EP1SGX10	672-pin FineLine BGA	12 *(5)*	0	0
EP1SGX25	672-pin FineLine BGA	16 *(6)*	8	4
	1,020-pin FineLine BGA	20	8	4
EP1SGX40	1,020-pin FineLine BGA	20	8	4

Notes to Table 6:

(1) See the *Using Selectable I/O Standards in Stratix & Stratix GX Devices* chapter of the *Stratix Device Handbook, Volume 2* for V_{REF} guidelines.

(2) This mode can be used for RLDRAM II ×9 and ×18 data bus width configuration if parity is not used (i.e., the design only uses 8 of the 9 data pins).

(3) This mode is for RLDRAM II ×9 and ×18 data bus width configuration. It can also be used for ×36 data bus width configuration if only 16 data pins are used.

(4) This mode is for RLDRAM II ×36 data bus width configuration.

(5) These packages have six groups in I/O banks 3 and 4 and six groups in I/O banks 7 and 8.

(6) These packages have eight groups in I/O banks 3 and 4 and eight groups in I/O banks 7 and 8.

The maximum number of RLDRAM II CIO devices that the Stratix II, Stratix, or Stratix GX device can interface with is partially dependent on the number of DQS and DQ groups available in that particular Altera device. When interfacing with the RLDRAM II SIO devices or when having multiple controllers in the FPGA for each RLDRAM II device, you also need to consider the available number of user I/O pins that support 1.8-V HSTL class I or class II, depending on SIO or CIO mode, I/O standard. Tables 7 through 9 show the maximum number of RLDRAM II devices that can be supported in Stratix II, Stratix, and Stratix GX devices using both available DLLs in the FPGA. These tables only consider the number of DQS and DQ groups in that particular device and do not take into account pin availability for the address and command pins, the loading effect on the address and command pins, V_{REF} pad placement guidelines, or simultaneous switching noise.

Table 7. Maximum Number of RLDRAM II Devices That Can Be Supported in Stratix II Devices *Note (1)*

Device	Package	Data Bus Width Configuration		
		×9	×18	×36
EP2S15	484-pin FineLine BGA	4	2	0
	672-pin FineLine BGA	8	4	2
EP2S30	484-pin FineLine BGA	4	2	0
	672-pin FineLine BGA	8	4	2
EP2S60	484-pin FineLine BGA	4	2	0
	672-pin FineLine BGA	8	4	2
	1,020-pin FineLine BGA	18	8*(2)*	4*(2)*
EP2S90	1,020-pin FineLine BGA 1,508-pin FineLine BGA	18	8*(2)*	4*(2)*
EP2S130	1,020-pin FineLine BGA 1,508-pin FineLine BGA	18	8*(2)*	4*(2)*
EP2S180	1,020-pin FineLine BGA 1,508-pin FineLine BGA	18	8*(2)*	4*(2)*

Note to Table 7:

(1) This is for DLL-based implementation. These numbers are preliminary and only take into account the number of DQS and DQ groups available in the device. You still need to make sure that there are enough pins for the address and command signals and that the address and command signals are not degraded from the multiple RLDRAM II loads. You should also consider the V_{REF} pad placement guidelines and simultaneous switching noise. For the maximum number of interfaces using just one DLL, divide the number by two.

(2) You can also have two extra ×9 RLDRAM II interfaces, one in I/O bank 4 and one in I/O bank 7.

Table 8. Maximum Number of RLDRAM II Devices that Can Be Supported in Stratix Devices *Notes (1), (2), (3)*

Device	Package	Data Bus Width Configuration		
		×9	×18	×36
EP1S10	672-pin BGA *(4)* 672-pin FineLine BGA *(4)*	0	0	0
	484-pin FineLine BGA *(4)* 780-pin FineLine BGA *(4)*	0	0	2
EP1S20	484-pin FineLine BGA	7	3	2
	672-pin BGA 672-pin FineLine BGA	7	3	2
	780-pin FineLine BGA	7	3	2
EP1S25	672-pin BGA 672-pin FineLine BGA	8	4	2
	780-pin FineLine BGA 1,020-pin FineLine BGA	8	4	2
EP1S30	956-pin BGA 780-pin FineLine BGA 1,020-pin FineLine BGA	8	4	2
EP1S40	956-pin BGA 1,020-pin FineLine BGA 1,508-pin FineLine BGA	8	4	2

Notes to Table 8:

(1) This is for DLL-based implementation. These numbers only take into account the number of DQS and DQ groups in that particular device and do not consider pin availability for the address and command pins, the loading effect on the address and command pins, V_{REF} pad placement guidelines, or simultaneous switching noise. For the maximum number of interfaces using just one DLL, divide the number by two.

(2) If you are not using the parity bits in the RLDRAM II devices, you can interface with more RLDRAM II devices by using the ×8 mode for RLDRAM II devices in the ×9 and ×18 data bus width configurations and the ×16 mode for RLDRAM II devices in the ×36 data bus width configuration.

(3) EP1S60 and EP1S80 devices do not support RLDRAM II memory.

(4) This device can support RLDRAM II devices in ×9 and ×18 bus width configurations if the interface does not use any parity bits by using the Stratix ×8 DQS and DQ mode.

Table 9. Maximum Number of RLDRAM II Devices that Can Be Supported in Stratix GX Devices
Notes (1), (2)

Device	Package	Data Bus Width Configuration		
		×8	×16	×32
EP1SGX10 *(3)*	672-pin FineLine BGA	0	0	0
EP1SGX25	672-pin FineLine BGA	8	4	2
	1,020-pin FineLine BGA	8	4	2
EP1SGX40	1,020-pin FineLine BGA	8	4	2

Notes to Table 9:

(1) This is for DLL-based implementation. These numbers only take into account the number of DQS and DQ groups in that particular device and do not consider account pin availability for the address and command pins, the loading effect on the address and command pins, V_{REF} pad placement guidelines, or simultaneous switching noise. For the maximum number of interfaces using just one DLL, divide the number by two.

(2) If you are not using the parity bits in the RLDRAM II devices, you can interface with more RLDRAM II devices by using the ×8 mode for RLDRAM II devices in the ×9 and ×18 data bus width configurations and the ×16 mode for RLDRAM II devices in the ×36 data bus width configuration.

(3) This device can support RLDRAM II devices in x9 and x18 bus width configurations if the interface does not use any parity bits by using the Stratix GX ×8 DQS and DQ mode.

Similar to DDR SDRAM or DDR2 SDRAM, the RLDRAM II data mask (DM) pins are only used during a write. The memory controller drives the DM signal low when the write is valid and drives it high to mask the DQ signals. There is one DM pin per RLDRAM II device. When interfacing with RLDRAM II CIO devices, you can use any of the I/O pins in the same bank as the associated DQ pins to generate the DM signals. When interfacing with RLDRAM II SIO devices, connect the DM pins to any of the Stratix II, Stratix, or Stratix GX device user I/O pins in the same bank as the D pins.

The DM timing requirements at the input to the RLDRAM II are identical to those for DQ data. The DDR registers, clocked by the write clock, create the DM signals. This reduces any skew between the DQ and DM signals.

The RLDRAM II device's setup time (t_{DS}) and hold (t_{DH}) time for the write DQ and DM pins are relative to the edges of the DK or DK# clocks. The DK and DK# signals are generated on the positive edge of system clock, so that the positive edge of CK or CK# is aligned with the positive edge of DK or DK# respectively to meet the RLDRAM II t_{CKDK} requirement. The DQ and DM signals are clocked using a shifted clock so that the edges of DK or DK# are center-aligned with respect to the DQ and DM signals when they arrive at the RLDRAM II device.

Perform timing analysis to calculate the optimal phase shift for the data and data mask signals.

The clocks, data, and DM board trace lengths should be tightly matched to minimize the skew in the arrival time of these signals.

RLDRAM II devices also have a QVLD pin indicating valid read data. The QVLD signal is edge-aligned with QK or QK# and is high approximately half a clock cycle before data is output from the memory. Connect the QVLD pin from the RLDRAM II device to the Stratix II device DQVLD pin.

The DQVLD pins in Stratix II devices are treated like DQ pins as signals coming into the DQVLD pins can be captured by the shifted DQS signal.

Stratix devices do not use the RLDRAM II QVLD pins. Resynchronization in this interface uses a feedback clock scheme.

Commands & Addresses

The CK and CK# signals clock the commands and addresses into RLDRAM II devices. These pins operate at single data rate using only one clock edge. RLDRAM II devices have 18 to 21 address pins, depending on the data bus width configuration and burst length. RLDRAM II supports both non-multiplexed and multiplexed addressing. Multiplexed addressing allows you to save a few user I/O pins while non-multiplexed addressing allows you to send the address signal within one clock cycle instead of two clock cycles. CS#, REF#, and WE# pins are input commands to the RLDRAM II device.

The commands and addresses must meet the memory address and command setup (t_{AS}, t_{CS}) and hold (t_{AH}, t_{CH}) time requirements. You can use any I/O pins to generate the commands and addresses for RLDRAM II.

Interface Architecture

Altera provides two different methods to implement the read-side interface to RLDRAM II devices. Stratix II, Stratix, and Stratix GX devices use a read-side DLL to center align the QK read clock with the read data (DQ or Q). In addition to that method, Stratix II devices also offer the PLL-based implementation, using a PLL to center align the QK read clock with read data. Stratix devices do not offer the PLL-based implementation.

The write-side implementation is identical for both PLL- and DLL-based implementations. A write-side PLL outputs two clocks that generate the write data (DQ or D) and center-aligned write clocks (DK and DK#) using the dedicated double data rate input/output (DDIO) circuits. This implementation results in matched propagation delays for clock and data signals from the FPGA to the RLDRAM II, minimizing skew.

For detailed information about the DDR I/O and DLL circuits, refer to the *External Memory Interface* chapter of the *Stratix II Device Handbook* or the *External Memory Interface* chapter of the *Stratix Device Handbook*.

DLL-Based Data-Path Architecture

The RLDRAM II interface implementation in Stratix II devices uses the following:

- A write-side PLL to generate CK and CK# system clocks, DK and DK# write clocks, and clock out address, command, and data signals.
- A read-side DLL-based phase-shift circuitry to register read data from the memory read clocks QK and QK#.

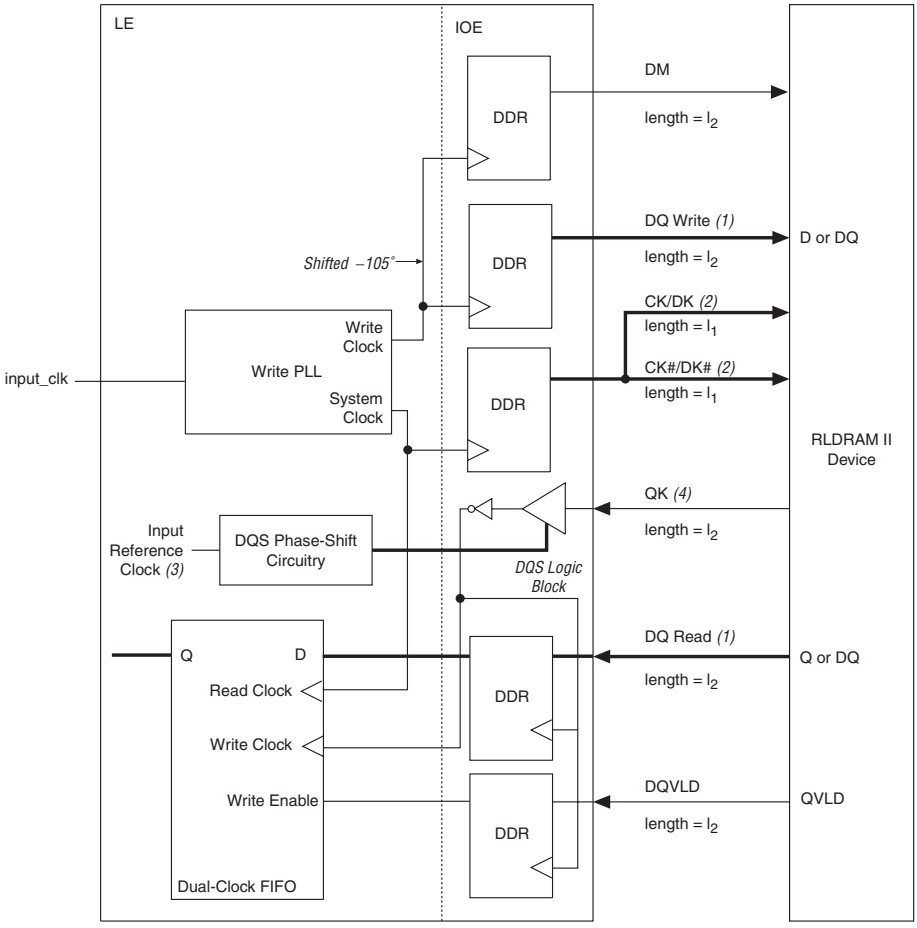
This implementation is also called DQS mode or DLL-based read implementation.

There are separate DQS phase shift circuits available on the top side and on the bottom side of the FPGA. Each DQS phase shift circuit needs an input reference clock. In Stratix and Stratix GX devices, the input reference clock must come from an input clock pin, while in Stratix II devices, the input reference can come from an input clock pin or PLL 5 or PLL 6. The DQS phase shift circuitry shifts the QK signal to center-align the signal with the DQ signal at the IOE register, ensuring the data gets latched at the IOE register. The QK signal is then inverted before going to the DQ IOE registers clock ports as described in the *External Memory Interfaces* chapter of the *Stratix II Device Family Handbook* or the *Stratix Device Family Handbook*.

Figure 4 shows a summary of how the data (D, Q, or DQ), CK, CK#, DK, DK#, QK, and QK# pins are connected in Stratix II devices. The write PLL generates the system clock and write clock. The system clock and write clock have the same frequency as the DQS signal. The write clock is shifted –105° from the system clock, which is the default in the RLDRAM II controller MegaCore function. You should perform your own timing analysis to calculate the optimal write phase shift for your system.

The example in Figure 4 uses the QVLD signal to synchronize the data to the system clock. The board trace length for QVLD should match closely with the data (D, Q, or DQ) and QK signal board trace lengths. The QVLD signal is captured as if it is data and then it is routed into the FPGA cores as a FIFO write enable. The FIFO is used as a buffer where the write clock is connected to the delayed DQS signal and the read clock is connected to the system clock. The overflow and underflow checking for the FIFO should be turned on to avoid reading garbage data from the FIFO.

Figure 4. DLL-Based RLDRAM II Data Path Interface for Stratix II Devices



Notes to Figure 4:

(1) RLDRAM CIO device data is bidirectional on the DQ pins. RLDRAM SIO device data is sent on two separate unidirectional lines (D for write data and Q for read data). For RLDRAM II SIO interfaces, connect Q to the DQ pins in the Stratix II device and use any of the user I/O pins in I/O banks 3, 4, 7, or 8 for the D ports.

(2) If loading may be an issue, you can use two different pins to generate CK and DK (and similarly CK# and DK#). However, please make sure that the skew reported by the Quartus II software meets the RLDRAM II device's t_{CKDK} specification.

(3) The input reference clock can either be from `input_clk`, another clock pin, or a PLL 5 or 6 output.

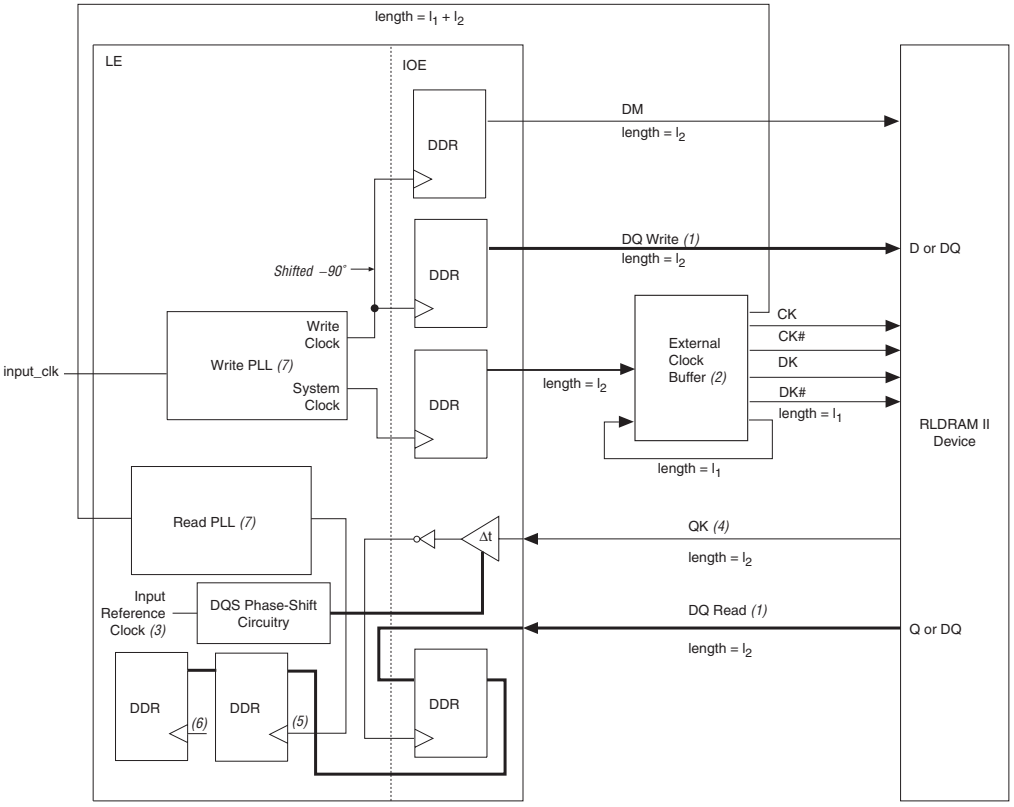
(4) Connect the RLDRAM II device QK pin to the DQS pins in Stratix II devices.

In Stratix II devices, the shifted DQS signal is routed to the FPGA logic array to clock the resynchronization FIFO and any read-side registers using regular routing. When using the resynchronization FIFOs, your memory controller needs to account for any additional latency from routing the DQS signals to the FPGA logic array.

Figure 5 shows the RLDRAM II interface data path for Stratix and Stratix GX devices. Figure 5 also uses a feedback clock and a second PLL to simplify resynchronization. Stratix II, Stratix, and Stratix GX devices use an external clock buffer to generate CK, CK#, DK, and DK# signals to meet the required 2-V/ns slew rate at the RLDRAM II device. The external clock buffer is a zero-delay clock buffer. The clock buffer loop-back trace length (l_1) is equal to the CK, CK#, DK, and DK# trace lengths from the clock buffer to the RLDRAM device such that the effective trace length for the CK, CK#, DK, and DK# signals from the FPGA to the RLDRAM II device is equal to l_2. Currently, the Altera RLDRAM II controller MegaCore function does not support the Stratix and Stratix GX device family.

Micron preliminary guidelines says that the required setup and hold time of the RLDRAM II device may double if the slew rate of the driving device is 1 V/ns.

Figure 5. DLL-Based RLDRAM II Data Path Interface for Stratix & Stratix GX Devices



Notes to Figure 5:

(1) RLDRAM CIO device data is bidirectional on the DQ pins. RLDRAM SIO device data is sent on two separate, unidirectional lines (D for write data and Q for read data). For RLDRAM II SIO interfaces, connect Q to the DQ pins in the Stratix and Stratix GX device and use any of the user I/O pins in I/O Banks 3, 4, 7, or 8 for the D ports.

(2) The external clock buffer is in external feedback mode. Its feedback input is in phase to the signals going into the RLDRAM II device minus any trace length skew, jitter, and offset error.

(3) The input reference clock must come from an input clock pin in Stratix and Stratix GX devices. You must use a PLL output clock to feed the DLL and must be able to only enable the input reference clock to the DLL during refresh and initialization cycle.

(4) Connect QK from the RLDRAM II devices to the DQS pins in Stratix and Stratix GX devices.

(5) The clock to the resynchronization register can be from the system clock, write clock, and extra clock output from the write PLL or from the read PLL.

(6) The clock to this register can either be the system clock or another clock output of the write PLL. If another clock output of the write PLL is needed, another register is needed to transfer the data back to the system clock domain.

(7) The PLL is in normal mode. You should only use even M and N PLL counter values to avoid excessive duty cycle distortion.

PLL-Based Data-Path Architecture

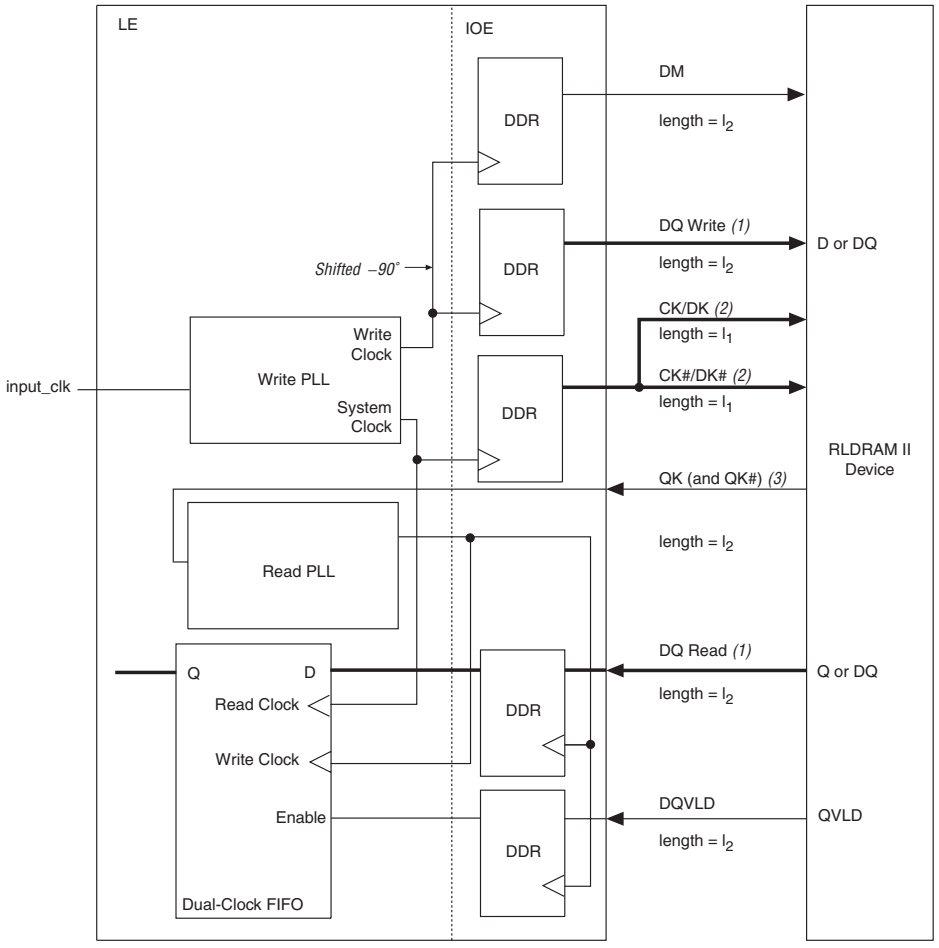
The RLDRAM II interface implementation without using dedicated DQS circuitry uses the following:

- A write-side PLL to generate CK and CK# system clocks DK and DK# write clocks, and clock out address, command, and data signals.
- A read-side PLL-based phase-shift to register read data from the memory.

This implementation is also called PLL-based read implementation (or non-DQS mode). The read PLL is in source synchronous mode in this implementation. You need to ensure that the read PLL is in the same side of the device as the data pins are in because Quartus II associates a particular PLL with a particular I/O bank for source synchronous operation. The clock delay to the worst case I/O registers in this I/O bank are fully compensated and result in closely matched data delays and clock delays from pin to the I/O registers across PVT. When using I/O registers in the non-compensated I/O banks, clock delays and data delays are less closely matched. Use a Fast PLL for implementing the interface in side I/O banks, and use an Enhanced PLL for implementing the interface on top or bottom I/O banks for best clock and data delay matching. You also need to set the input pin delay to register option to 0 in Quartus II.

Figure 6 shows a summary of how Stratix II devices generate the DQ, DQS, CK, and CK# signals. The write PLL generates system clock and write clock. The read clock QK from the RLDRAM II device goes to a PLL input pin which generates the proper phase shift to capture read data.

Figure 6. PLL-Based RLDRAM II Data Path Interface in Stratix II Devices



Notes to Figure 6:

(1) RLDRAM CIO device data is bidirectional on the DQ pins. RLDRAM SIO device data is sent on two separate unidirectional lines (D for write data and Q for read data). For RLDRAM II SIO interfaces, connect Q to the DQ pins in the Stratix II device and use any of the user I/O pins in I/O banks 3, 4, 7, or 8 for the D ports.

(2) If loading may be an issue, you can use two different pins to generate CK and DK (and similarly CK# and DK#). However, please make sure that the skew reported by the Quartus II software meets the RLDRAM II device's t_{CKDK} specification.

(3) Connect the RLDRAM II QK to the input clock pin of the read PLL. You can also connect the differential QK/QK# to the differential input clock pins of the read PLL.

Altera Memory Controller IP

The RLDRAM II Controller MegaCore function allows you to instantiate a simplified interface to the industry-standard RLDRAM II memory. The RLDRAM II Controller initializes the memory devices and manages the read and write operations. The MegaCore function translates read and write requests from the local interface into all the necessary RLDRAM II command signals.

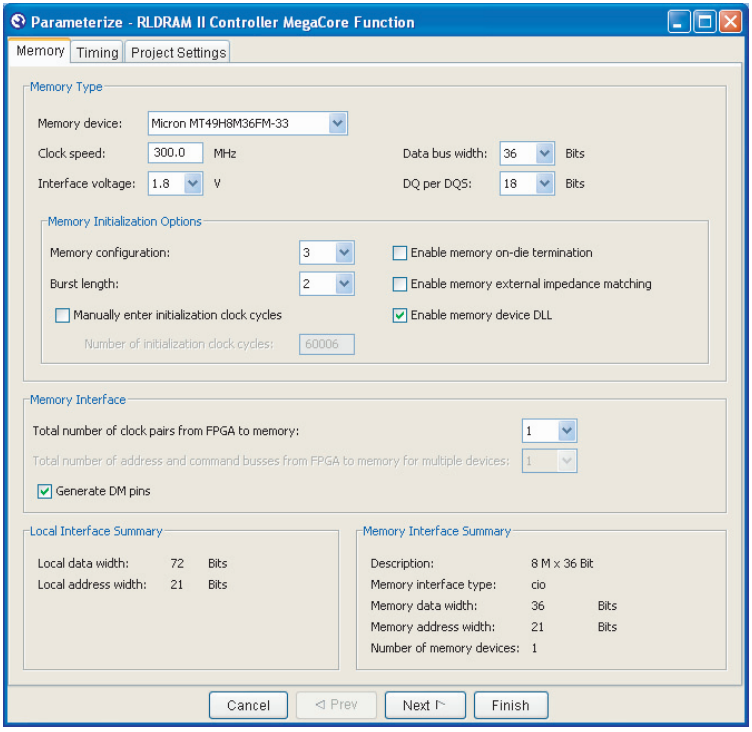
The RLDRAM II Controller contains encrypted control logic as well as an clear-text data path that you can use in your design without a license. Download this MegaCore function whether you plan to use the Altera RLDRAM II controller or not to get the clear-text data path, logic and pin placement constraints.

You can download the RLDRAM II Controller MegaCore function from www.altera.com. Alternatively, the MegaCore function is also available in the IP library CD as part of the Quartus II CD pack.

The MegaCore function is accessible through the RLDRAM II IP Tool Bench. When you parameterize your custom RLDRAM II interface, the RLDRAM II IP Tool Bench automatically decides the best-read phase shift for your frequency of operation and FPGA settings to give you the best margin for your RLDRAM II interface. It then generates an example instance that instantiates a PLL, an example driver, and your RLDRAM II Controller custom variation as shown in Figure 1 on page 4.

After downloading and installing the RLDRAM II Controller MegaCore function, you need to create a project before launching the RLDRAM II IP Tool Bench. Figure 7 shows the Parameterize page of the RLDRAM II IP tool bench. You need to pick the RLDRAM II memory, the frequency of operation, and the configuration that you are implementing. You can also set the location constraints and generate the simulation model from the RLDRAM II IP Tool Bench. The RLDRAM II IP Tool bench generates a fully functional example design that can be simulated, synthesized, and used in hardware. The example driver in the example design issues read and writes to the controller and compares the read data with the previously written data to generate pass-fail and test-complete signals. If you do not want to use the Altera RLDRAM II controller encrypted control logic, you can replace it with your own custom logic. This allows you to use the Altera supported data path with your own logic. For more detailed information on the usage of the RLDRAM II Controller MegaCore function, please download the RLDRAM II Controller MegaCore User Guide at **www.altera.com**.

Figure 7. RLDRAM II Controller MegaCore Function



Interface Timing Analysis

When designing an external memory interface for your FPGA, you have to analyze timing margins for several paths. All memory interfaces require analysis of the read and write capture timing paths. Additionally, some interfaces might require analysis of the resynchronization timing paths and other memory-specific paths (such as postamble timing).

This application note describes Altera's recommended timing methodology using write and read capture timing paths as examples. You should use this methodology for analyzing timing for all applicable timing paths (including address/command and resynchronization for Stratix and Stratix GX interfaces). While these analyses account for all FPGA related timing effects, you should design in adequate margin to account for board level effects.

The following presents an analysis of the read and write capture timing margins for the Micron MT49H8M36FM-33 RLDRAM II interface with the Stratix II EP2S60F1020C3. This is a sample interface that shows you

the proper methodology for this timing analysis to ensure you include the proper timing specifications for your preferred FPGA and memory device.

Methodology

To analyze timing paths you need to consider the data and clock arrival times at the destination register. Figure 8 illustrates a simplified block diagram to analyze timing at any register. The setup time margin is defined as the time between "earliest clock arrival time" and "latest valid data arrival time" at the register ports. Similarly, hold time margin is defined as the time between "earliest invalid data arrival time" and the "latest clock arrival time" at the register ports. These arrival times are calculated based on propagation delay information with respect to a common reference point (such as a memory clock edge or system clock edge). Figure 9 shows how a data valid window is derived based on arrival times.

Figure 8. Simplified Block Diagram for Timing Analysis

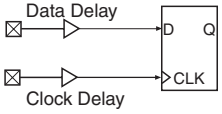
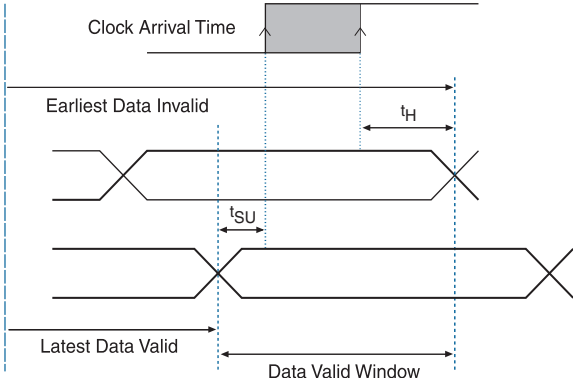


Figure 9. Data Valid Window Timing Waveform



FPGA Timing Information

If your design is required to work under all conditions (between 0° C to 85° C for commercial FPGAs), the timing margins should be evaluated at all process, voltage, and temperature (PVT) conditions. To facilitate this, Altera provides two device timing models in the Quartus II software: slow corner model and fast corner model.

- The slow corner model provides timing delays between two nodes within the FPGA with slow silicon for that speed grade, high temperature, and low voltage. In other words, the model provides the slowest possible delay for that timing path on any device for that particular speed grade.
- The fast corner model provides timing delays between two nodes within the FPGA with fast silicon, low temperature, and high voltage. In other words, the model provides the fastest possible delay for that timing path on any device in that density.

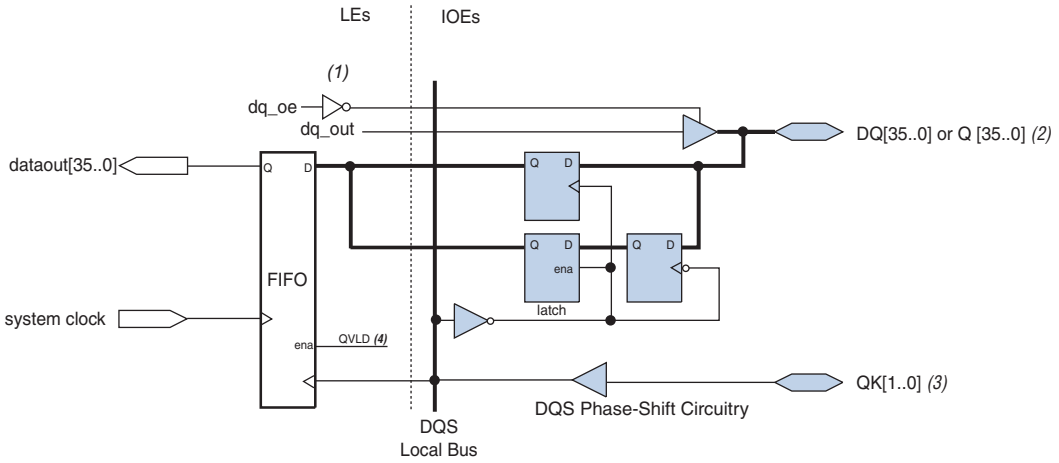
Note that while almost all FPGA timing delays and uncertainties are modeled in the Quartus II software, a handful of factors are not modeled and should be accounted for during margin analysis. Some examples include clock jitter on PLL and DLL outputs. These uncertainties are described in the Stratix II device data sheet. These timing uncertainties or adder terms, when used in conjunction with the Quartus II software reported timing data, provide the most accurate device timing information. The following analysis will detail the use of these timing adder terms.

Read Timing Margins for DLL-Based Implementation

During read operations, the RLDRAM II memory device provides a read clock (QK) that is edge-aligned with the data bus (Q or DQ). The memory controller (FPGA) is required to shift the clock edge to the center of the data valid window and capture the DQ input data. See Figure 2 on page 8 for the relationship between DQ and QK during a read.

Figure 10 shows the DLL-based read data path from the Stratix II device. The QK signal goes to the DQS phase-shift circuitry and gets shifted. The shifted QK signal goes to the DQS bus and gets inverted before it clocks the DQ input registers. The register outputs then go to a dual-clock FIFO with the captured QVLD signal as the write enable.

Figure 10. RLDRAM II DLL-Based Read Data Path in Stratix II Devices



Notes to Figure 10:

(1) The `dq_oe` signals are active low in silicon. However, the Quartus II software implements it as active high and adds the inverter automatically during compilation.

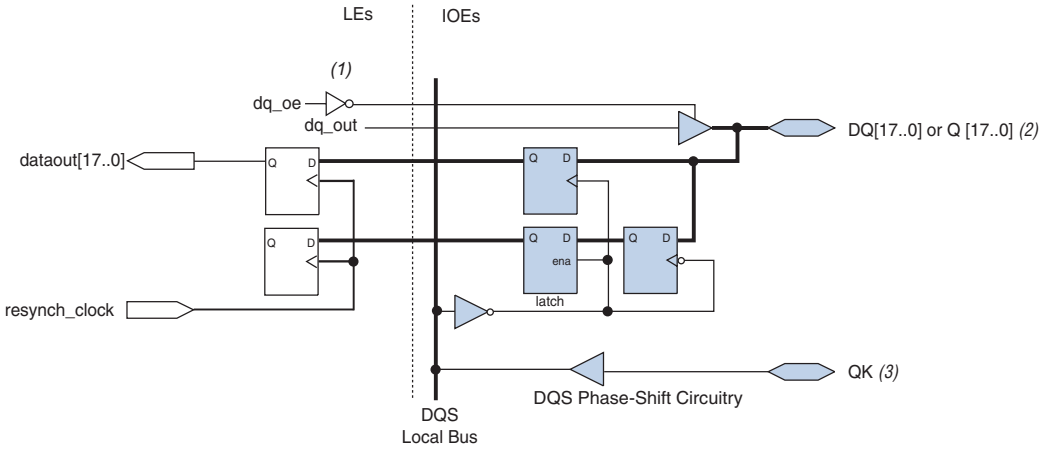
(2) The RLDRAM II device may have the same or separate ports for data read and data. If separate, connect Q of the RLDRAM II device to the DQ pins of the Stratix II, Stratix, or Stratix GX device and connect `dq_oe` to GND. You can use any of the user I/O pins in I/O banks 3, 4, 7, or 8 to connect to the D pins of the RLDRAM II device.

(3) The DQS pins in Stratix II, Stratix, or Stratix GX devices must be connected to the QK signals of the RLDRAM II device.

(4) This QVLD signal has been captured in the IOE by the shifted QK signal.

Figure 11 shows the DLL-based read data path from the Stratix or Stratix GX device. The QK signal goes to the DQS phase-shift circuitry and gets shifted. The shifted QK signal goes to the DQS bus and gets inverted before it clocks the DQ input registers. The register outputs then go to the resynchronization register in the logic array, sampled by the `resynch_clock` signal. You can use the system clock, the write clock, or another clock output of the PLL as the `resynch_clock` signal.

Figure 11. RLDRAM II Read Data Path in Stratix or Stratix GX Devices



Notes to Figure 11:

(1) The `dq_oe` signals are active low in silicon. However, the Quartus II software implements it as active high and adds the inverter automatically during compilation.

(2) The RLDRAM II device may have the same or separate ports for data read and data. If separate, connect Q of the RLDRAM II device to the DQ pins of the Stratix II, Stratix, or Stratix GX device and connect `dq_oe` to GND. You can use any of the user I/O pins in I/O banks 3, 4, 7, or 8 to connect to the D pins of the RLDRAM II device.

(3) The DQS pins in Stratix II, Stratix, or Stratix GX devices must be connected to the QK signals of the RLDRAM II device.

Memory Timing Parameters

You would start the read timing analysis by obtaining the timing relationship between the DQ and DQS outputs from the RLDRAM II memory device. The following describes the timing analysis for the 300-MHz RLDRAM II interface in a Stratix II EP2S60F1020C3 device. For 300-MHz clock speeds or 600-Mbps data rates, the read data clock high or low time is 1349 ps after accounting for duty cycle distortion on the input clock to the memory and the read data clock itself. This is specified as t_{QKH}

in the memory data sheet and is 0.9 × 0.45 x 3333 ps. Apart for t_{QKH}, the memory also specifies $t_{QKQ0/QKQ1}$, which specifies the maximum time from a QK edge to the last Q or DQ valid. For 300 MHz operation, $t_{QKQ0/QKQ1}$ is specified to be ± 250 ps.

With these memory timing parameters, the data valid window at the memory to be equal to $t_{QKH} - 2 \times t_{QKQ0}/_{QKQ1} = 849$ ps. Assuming the board trace length variations amongst all DQ and DQS traces are not more than ± 20ps, the data valid window present at the FPGA input pins is 809 ps.

FPGA Timing Parameters

FPGA timing parameters are obtained from two sources: the Quartus II software timing analyzer and the Stratix II data sheet. While the former provides all clock/data propagation delays, the data sheet specifies all clock uncertainties and skew adder terms.

The timing analysis methodology outlined earlier suggests the use of the earliest and latest arrival times for clock and data. The following details timing analysis for the clock (QK).

The Stratix II features dedicated phase-shift circuitry in the top/bottom IO banks of the device, which will center-align the DQS edge with respect to the DQ input signals. This phase shift circuitry has a coarse and fine delay resolution. The coarse delay feature is self-compensating over PVT and has a resolution of 22.5° to 36° of the reference clock frequency (based on the DLL mode of operation). For 300 MHz, you can select between DLL modes 2 (high) and 3 (very high). DLL mode 2 gives a 30° coarse phase resolution, while mode 3 gives 36° resolution. You can further fine-tune this phase shift with a DLL phase-shift offset implement using uncompensated delay chains. This example analyzes timing with a 72° phase shift on the DQS strobe (DLL mode 3), knowing that the phase shift (and DLL mode) can always be adjusted at the end of this timing analysis for balanced setup and hold margins on the read capture register.

Table 10 shows the default DQS phase shifts in the Altera RLDRAM II Controller MegaCore function.

Table 10. Default Phase Shifts in the Altera RLDRAM II Controller MegaCore Function

RLDRAM II Clock Frequency	Default DQS Phase Shift *Note (1)*
241 – 300 MHz	72°
230 – 240 MHz	90°
175 – 229 MHz	67.5°

Notes to Table 10:
(1) You should calculate your read capture margin and reconfirm that the default DQS phase shift is acceptable for your system.

The DQS phase shift circuitry uses a DLL to provide the self-compensating coarse delay shift. This means you have to account for any jitter and phase shift error on the DQS signal. To achieve 72° phase shift in DLL mode 3, two DQS delay buffer stages are used in the DQS logic block. The data sheet shows t_{DQS_JITTER} is ± 55 ps and t_{DQS_PSERR} is ± 25 ps timing parameters when using two DQS delay buffer stages.

After encountering the phase shift circuitry, the DQS signal travels on a dedicated DQS bus to the DQ capture registers. The fan-out of this bus could range from ×4 to ×36. While the Quartus II software provides clock propagation delays to each of these DQ register clock ports, you still need to account for additional uncertainties with the $t_{DQS\_SKEW\_ADDER}$ adder term listed in the data sheet. For the ×18 mode used by this Micron RLDRAM II device, the skew adder is ± 37.5 ps.

To obtain the Quartus II software timing data for the target device, instantiate and compile the RLDRAM II Controller MegaCore function. If you are using your own controller logic, you should instantiate the clear-text RLDRAM II data path instead to obtain timing delays. For the read interface, the Quartus II software reports individual setup and hold times for each DQ pin. Select the "List Paths" option in the timing report to get the data and clock propagation delays for that DQ pin. Select the worst-case setup and hold DQ registers to extract the minimum and maximum propagation delays.

For example, Figure 12 shows a "List Paths" example on the setup time for DQ[18]. This path shows propagation delays of 1.815 ns on the DQ pin to register path, and 2.407 ns on the DQS clock pin to register path.

Figure 12. List Paths

```
Info: tsu for register "my_core_wrapper:my_core|<..>|dq_captured_falling[10]" (data pin = "rldramii_dq[10]", clock pin = "rldramii_qk[0]") is -0.470 ns
    Info: + Longest pin to register delay is 1.815 ns
        Info: 1: + IC(0.000 ns) + CELL(0.000 ns) = 0.000 ns; Loc. = PIN_A28; Fanout = 2; PIN Node = 'rldramii_dq[10]'
        Info: 2: + IC(0.000 ns) + CELL(1.815 ns) = 1.815 ns; Loc. = IOC_X7_Y52_N1; Fanout = 1; REG Node = 'my_core_wrapper:my_core|<..>|dq_captured_falling[10]'
        Info: Total cell delay = 1.815 ns ( 100.00 % )
    Info: + Micro setup delay of destination is 0.122 ns
    Info: - Shortest clock path from clock "rldramii_qk[0]" to destination register is 2.407 ns
        Info: 1: + IC(0.000 ns) + CELL(0.000 ns) = 0.000 ns; Loc. = PIN_D25; Fanout = 1; CLK Node = 'rldramii_qk[0]'
        Info: 2: + IC(0.000 ns) + CELL(1.704 ns) = 1.704 ns; Loc. = IOC_X11_Y52_N0; Fanout = 352; COMB Node = 'my_core_wrapper:my_core|<..>|rldramii_dqs_group_0|dqs_clk[0]'
        Info: 3: + IC(0.298 ns) + CELL(0.405 ns) = 2.407 ns; Loc. = IOC_X7_Y52_N1; Fanout = 1; REG Node = 'my_core_wrapper:my_core|my_core_auk_rldramii_datapath:rldramii_io|<..>|dq_captured_falling[10]'
        Info: Total cell delay = 2.109 ns ( 87.62 % )
        Info: Total interconnect delay = 0.298 ns ( 12.38 % )
```

Using this approach, minimum and maximum propagation delays on the clock and data path are extracted and presented in Table 11. This timing extraction is done twice, once with each device model (fast model and slow model). Observe that the difference between minimum and maximum delays is very small due to the matched routing paths within the die and package.

Table 11. Minimum and Maximum Propagation Delays for EP2S60F1020C3 Device

FPGA Timing Delays	Fast Model	Slow Model for C3 Speed Grade
Data Delay (minimum)	1.152 ns	1.755 ns
Data Delay (maximum)	1.212 ns	1.815 ns
Clock Delay (minimum)	1.808 ns	2.407 ns
Clock Delay (maximum)	1.826 ns	2.430 ns
Micro Setup	0.068 ns	0.122 ns
Micro Hold	0.037 ns	0.072 ns

Setup & Hold Margins Calculations

You can calculate the read setup and hold time margins of the DQ capture register after obtaining all relevant timing information from the memory, FPGA, and board.

Earliest clock arrival time = Minimum clock delay within FPGA – DQS uncertainties

$$= \text{Clock delay (minimum)} - t_{DQS\_JITTER} - t_{DQS\_PSERR} - t_{DQS\_SKEW\_ADDER}$$

$$= 2407 - 30 - 25 - 37.5$$

$$= 2314.5 \text{ ps (with slow timing model)}$$

Latest data valid time = Memory DQS-to-DQ valid + maximum data delay in FPGA
= t_{QKQ0}/t_{QKQ1} + data delay (maximum)
= 250 + 1815
= 2065 ps (with slow timing model)

Setup time margin = Earliest clock arrival – latest data valid – micro setup – board uncertainty
= $t_{EARLY\_CLOCK} - t_{LATE\_DATA\_VALID} - \mu t_{SU} - t_{EXT}$
= 2314.5 – 2065 – 122 – 20
= 107.5 ps (with slow timing model)

When repeating these calculations with the fast timing model, the derived setup margin is 166 ps.

Latest clock arrival time = Maximum clock delay within FPGA + DQS uncertainties
= Clock delay (maximum) + t_{DQS_JITTER} + t_{DQS_PSERR} + $t_{DQS\_SKEW\_ADDER}$
= 2430 + 30 + 25 + 37.5
= 2522.5 ps

Earliest data invalid time = Memory clock-to-data invalid + minimum data delay in FPGA
= ($t_{QKH} - t_{QKQ0/QKQ1}$) + data delay (minimum)
= (1349 – 250) + 1755
= 2854 ps (with slow timing model)

Hold time margin = Latest clock arrival time – earliest data_Latest clock arrival time invalid time – micro hold – board uncertainty
= $t_{EARLY\_DATA\_INVALID} - t_{LATE\_CLOCK} - \mu t_{H} - t_{EXT}$
= 2854 – 2522.5 – 72 – 20
= 239.5 ps (with slow timing model)

You also need to repeat the hold timing margin calculation with the fast timing model. Using the fast model numbers in Table 11, the margin is 276 ps.

Table 12 shows the read timing margin analysis at 300 MHz for the RLDRAM II interface in Stratix II EP2S60F1020C3 devices, when the board trace variations for the DQ and DQS pins is ± 20 ps (approximately ± 0.12 inches of FR4 trace length variations) using the methodology that was just described. Table 13 shows the RLDRAM II read timing margins

analysis with Stratix or EP1S25F780C5 at 200 MHz. You can perform a similar timing analysis for your interface with another RLDRAM II memory, by replacing the $t_{QKQ0/QKQ1}$ and t_{QKH} values in the table below with those from your memory data sheet and by replacing the FPGA specification for your particular device. This timing analysis applies for both RLDRAM II CIO and SIO interfaces.

Table 12. Read Data Timing Analysis for 300-MHz RLDRAM II Interface in EP2S60F1020C3 with the DQS Phase-Shift Circuitry (Part 1 of 2)

	Parameter	Fast Corner Model	Slow Corner Model	Description
Memory Specifications *(1)*	t_{QKH}	1.349	1.349	Half period as specified by the memory data sheet (including memory clock duty cycle distortion)
	$t_{QKQ0/tQKQ1}$	0.250	0.250	QK edge to output data edge skew
FPGA Specifications *(2)*	$t_{DQS\_PHASE\_JITTER}$	0.030	0.030	This is the peak-to-peak digital-phase jitter on the DQS clock network with the DLL used
	t_{DQS_PSERR}	0.025	0.025	Phase shift error on DQS output delayed by DLL
	$t_{DQS\_SKEW\_ADDER}$	0.038	0.038	Clock delay skew adder for x16/x18 DQS/DQ group
	Minimum Clock Delay (Input) *(3)*, *(4)*	1.808	2.407	Minimum DQS pin to IOE register delay from Quartus II (with 72° DLL-based phase shift)
	Maximum Clock Delay (Input) *(3)*, *(4)*	1.826	2.430	Maximum DQS pin to IOE register delay from Quartus II (with 72° DLL-based phase shift)
	Minimum Data Delay (Input) *(3)*, *(4)*	1.152	1.755	Minimum DQ pin to IOE register delay from Quartus II
	Maximum Data Delay (Input) *(3)*, *(4)*	1.212	1.815	Maximum DQ pin to IOE register delay from Quartus II
	μt_{SU} *(3)*	0.068	0.122	Intrinsic setup time of the IOE register
	μt_{H} *(3)*	0.037	0.072	Intrinsic hold time of the IOE register
Board Specifications	t_{EXT}	0.020	0.020	Board trace variations on the DQ and DQS lines

Table 12. Read Data Timing Analysis for 300-MHz RLDRAM II Interface in EP2S60F1020C3 with the DQS Phase-Shift Circuitry (Part 2 of 2)

	Parameter	Fast Corner Model	Slow Corner Model	Description
Timing Calculations	t_{EARLY_CLOCK}	1.716	2.315	Earliest possible clock edge after DQS phase-shift circuitry and uncertainties (minimum clock delay – t_{DQS_JITTER} – t_{DQS_PSERR} – $t_{DQS\_SKEW\_ADDER}$)
	t_{LATE_CLOCK}	1.919	2.523	Latest possible clock edge after DQS phase-shift circuitry and uncertainties (maximum clock delay + t_{DQS_JITTER} + t_{DQS_PSERR} + $t_{DQS\_SKEW\_ADDER}$)
	$t_{EARLY\_DATA\_INVALID}$	2.251	2.854	Time for earliest data to become invalid for sampling at FPGA flop (t_{QKH} – t_{QKQ} + minimum data delay)
	$t_{LATE\_DATA\_VALID}$	1.462	2.065	Time for latest data to become valid for sampling at FPGA flop (t_{QKQ} + maximum data delay)
Results	Read setup timing margin	0.166	0.108	t_{EARLY_CLOCK} – $t_{LATE\_DATA\_VALID}$ – μt_{SU} – t_{EXT}
	Read hold timing margin	0.276	0.240	$t_{EARLY\_DATA\_INVALID}$ – t_{LATE_CLOCK} – μt_{H} – t_{EXT}
	Total margin	0.441	0.347	Setup margin + hold margin

Notes for Table 12:
(1) The memory numbers used here come from Micron MT49H8M36FM-33.
(2) This analysis is performed with FPGA timing parameters for Stratix II EP2S60F1020C3. You should use this template to analyze timing for your preferred Stratix II density-package combination. For more information on FPGA specifications, refer to the *DC & Switching Characteristics* in *Volume 1* of the *Stratix II Handbook*.
(3) These numbers are from the Quartus II software version 5.1.
(4) Package trace skews are modeled by the Quartus II software.

Table 13. Read Analysis When Using DQS Circuitry in Stratix & Stratix GX -5 Speed Grade Devices (Part 1 of 2)

Parameter	Specification	200 MHz *(1)*	Description
Memory specifications	t_{QKH}	2.025 ns	Minimum output data clock high time as specified by the memory data sheet
	t_{QKQ}	0.350 ns	QK edge to any output data edge as specified by the memory data sheet

Table 13. Read Analysis When Using DQS Circuitry in Stratix & Stratix GX -5 Speed Grade Devices (Part 2 of 2)

Parameter	Specification	200 MHz *(1)*	Description
FPGA specifications	DLL phase shift *(2)*	1.000 ns	Ideal DLL phase shift
	$t_{DLLJITTER}$ *(3)*	0.100 ns	Stratix device DLL jitter
	t_{PSERR}	0.082 ns	DLL phase shift error
	$t_{DQS2IOE\_MIN}$ *(4)*	0.579 ns	Minimum DQS pin to IOE register delay
	$t_{DQS2IOE\_MAX}$ *(4)*	1.021 ns	Maximum DQS pin to IOE register delay
	t_{DQ2IOE_MIN} *(4)*	0.549 ns	Minimum DQ pin to IOE register delay
	t_{DQ2IOE_MAX} *(4)*	0.939 ns	Maximum DQ pin to IOE register delay
	$t_{DQSQINT}$	0.150 ns	DQS–DQ internal skew inside Stratix device
	μt_{SU}	0.280 ns	Intrinsic setup time of the IOE register (rounded up)
	μt_{H}	0.070 ns	Intrinsic hold time of the IOE register (rounded up)
Board specification	t_{EXT}	0.050 ns	Board trace variations for the DQ and DQS lines
Timing calculations	t_{SHIFT_MIN}	0.918 ns	Minimum shift provided by the DQS phase-shift circuitry (DLL Shift – t_{PSERR} – $t_{DLLJITTER}$)
	t_{SHIFT_MAX}	1.082 ns	Maximum shift provided by the DQS phase-shift circuitry (DLL Shift + t_{PSERR} + $t_{DLLJITTER}$)
	t_{DELTA_MIN}	0.976 ns	Minimum difference between the DQS and the DQ signals paths ($t_{DQS2IOE\_MIN}$ + t_{SHIFT_MIN} – t_{DQ2IOE_MIN})
	t_{DELTA_MAX}	1.182 ns	Maximum difference between the DQS and the DQ signals paths ($t_{DQS2IOE\_MAX}$ + t_{SHIFT_MAX} – t_{DQ2IOE_MAX})
Results	Read setup timing margin	0.118 ns	$t_{DELTA\_MIN} - t_{QKQ} - t_{EXT} - t_{DQSQINT} - \mu t_{SU}$
	Read hold timing margin	0.241 ns	$t_{QKH} - t_{QKQ} - \mu t_{H} - t_{EXT} - t_{DQSQINT} - t_{DELTA\_MAX}$

Notes to Table 13:

(1) The memory numbers used here come from Micron MT49H16M18/C-5. Assume the external clock buffer does not add any skew to the DK and D or DQ signals.

(2) This example uses 72º phase shift.

(3) This example uses $t_{DLLJITTER}$ equal to 0, because the DLL is on only during refresh and initialization cycles.

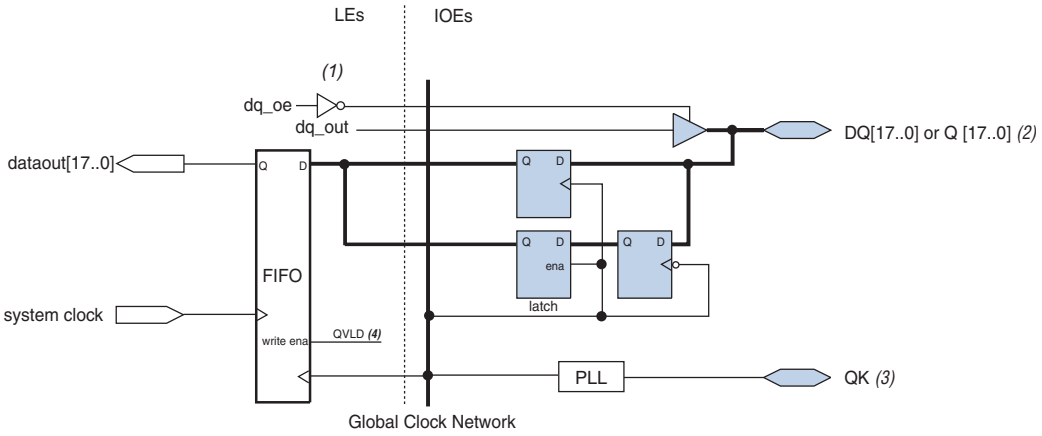
(4) Numbers are taken from the Quartus II software. Use the latest version of the Quartus II software for the most current information.

Read Timing Margins for PLL-Based Implementation

Timing margin analysis for a PLL-based implementation is very similar to the previously described DLL-based implementation. The only differences are the capture clock used and related clock uncertainties. In this mode, the write clock, QK signal is fed into a PLL inside the FPGA. The timing margin analysis presented here uses Stratix II EP2S60F1020C3 devices.

Figure 11 shows the PLL-based read data path from the Stratix II device. The QK signal goes to the PLL and the PLL generates a phase shifted read clock to capture the read data. The outputs of the DQ registers then go to a dual-clock FIFO with the capture QVLD signal as the write enable. Alternatively, you can also take the differential QK and QK# signals and use the PLL differential inputs for the write PLL to take advantage of the differential signaling from the RLDRAM II devices.

Figure 13. PLL-Based Read Data Path *Notes (1), (2), (3), (4)*



Notes to Figure 13:

(1) The `dq_oe` signals are active low in silicon. However, the Quartus II software implements it as active high and adds the inverter automatically during compilation.

(2) The RLDRAM II device may have the same or separate ports for data read and data. If separate, connect Q of the RLDRAM II device to the DQ pins of the Stratix II, Stratix, or Stratix GX device and connect `dq_oe` to GND. You can use any of the user I/O pins in I/O banks 3, 4, 7, or 8 to connect to the D pins of the RLDRAM II device.

(3) The QK (or QK and QK#) signals from the RLDRAM II device must be connected to an input PLL clock pin. Use a Fast PLL for RLDRAM II interfaces on the side I/O banks. Use an enhanced PLL for RLDRAM II interfaces on the top and bottom I/O banks.

(4) This QVLD signal has been captured in the IOE by the shifted QK signal.

Memory Timing Parameters

The timing relationship of data (Q or DQ) with respect to the read clock QK is governed by the t_{QKQ}. For the RLDRAM II memory device under consideration, this timing parameter is ± 500ps.

FPGA Timing Parameters

When the read clock, QK is fed into the PLL for read capture, uncertainties introduced on this clock include jitter and phase shift error. PLL-based read implementation uses a single global clock network to distribute the phase shifted clock signal to DQ capture registers in the IOE. The difference in clock arrival times to these registers (clock skew) is modeled in the Quartus II software, and is reflected in the minimum/maximum propagation delays for the clock. Additionally, the Quartus II software models the package trace delays for every pin in the device. Hence, you do not account for such skews separately in our timing margin analysis. The extracted minimum/maximum clock and data delays account for these uncertainties. Table 14 shows the read timing margin calculation for the PLL-based RLDRAM II interface using an EP2S60F1020C3 device.

Table 14. Read Timing Analysis for 200-MHz DDR2 SDRAM Interface in EP2S60F1020C3 Non-DQS Mode (Part 1 of 3)

Parameter	Specification	Fast Model	Slow Model	Description
Memory Specifications *(1)*	t_{QKH}	2.025	2.025	Half period as specified by the memory data sheet (including memory clock duty cycle distortion)
	t_{QKQ} *(2)*	0.400	0.400	QK edge to output data edge skew

Table 14. Read Timing Analysis for 200-MHz DDR2 SDRAM Interface in EP2S60F1020C3 Non-DQS Mode (Part 2 of 3)

Parameter	Specification	Fast Model	Slow Model	Description
FPGA Specifications	t_{PLL_JITTER} *(2)*	0.125	0.125	Stratix II PLL jitter
	$t_{PLL\_COMP\_ERROR}$ *(3)*	0.100	0.100	PLL Compensation Error (high bandwidth)
	t_{PLL_PSERR} *(3)*	0.030	0.030	PLL Phase Shift Error
	Minimum Clock Delay *(4)*, *(5)*	1.725	2.161	Minimum feedback clock pin to IOE register delay from Quartus II (75° PLL phase shift)
	Maximum Clock Delay *(4)*, *(5)*	1.756	2.213	Maximum feedback clock pin to IOE register delay from Quartus II (75° PLL phase shift)
	Minimum Data Delay *(4)*	0.621	0.974	Minimum DQ pin to IOE register delay from Quartus II
	Maximum Data Delay *(4)*, *(5)*, *(6)*	0.695	1.050	Maximum DQ pin to IOE register delay from Quartus II
	μt_{SU} *(5)*, *(6)*	0.068	0.122	Intrinsic setup time of the IOE register
	μt_{H}	0.037	0.068	Intrinsic hold time of the IOE register
Board Specifications	t_{EXT_SKEW}	0.020	0.020	Board trace variations on the DQ and DQS lines
Timing Calculations	t_{EARLY_CLOCK}	1.470	1.906	Earliest possible clock edge after DQS phase-shift circuitry and uncertainties (minimum clock delay + PLL phase shift – t_{PLL_PSERR} – t_{PLL_JITTER} - $t_{PLL\_COMP\_ERROR}$)
	t_{LATE_CLOCK}	2.011	2.468	Latest possible clock edge after DQS phase-shift circuitry and uncertainties (maximum clock delay + PLL phase shift + t_{PLL_PSERR} + t_{PLL_JITTER} + $t_{PLL\_COMP\_ERROR}$)
	$t_{EARLY\_DATA\_INVALID}$	2.246	2.599	Time for earliest data to become invalid for sampling at FPGA flop (t_{HP} - t_{AC} + minimum data delay)
	$t_{LATE\_DATA\_VALID}$	1.095	1.450	Time for latest data to become valid for sampling at FPGA flop (t_{AC} + maximum data delay)

Table 14. Read Timing Analysis for 200-MHz DDR2 SDRAM Interface in EP2S60F1020C3 Non-DQS Mode (Part 3 of 3)

Parameter	Specification	Fast Model	Slow Model	Description
Results	Read setup timing margin	0.287	0.314	t_{EARLY_CLOCK} - $t_{LATE\_DATA\_VALID}$ - T_{su}(micro) - t_{EXT}
	Read hold timing margin	0.178	0.043	$t_{EARLY\_DATA\_INVALID}$ - t_{LATE_CLOCK} - Th(micro) - t_{EXT}
	Total margin	0.465	0.357	Setup + hold time margin

Notes to Table 14:

(1) The memory numbers used here come from Micron MT49H8M36FM-33.
(2) Only one QK signal is connected to the PLL, while there are 2 in the device. You cannot use the t_{QKQ0}/t_{QKQ1} parameter due to this.
(3) Value may change pending final characterization.
(4) PLL phase shift is adjustable if you need to balance the setup and hold time margin.
(5) These numbers are from the Quartus II software, version 5.1 using the RLDRAM II MegaCore function version 1.0.0.
(6) Package trace length skew is modeled in Quartus II version 5.0 and higher. There is no additional adder required.

If you need more hold time margin, then you need less PLL phase shift. Similarly if you need more setup margin, you need more phase shift. In the example in Table 14, you can balance the setup and hold margin by using a PLL phase shift that is lower than 75°.

Write Data Timing Margins

For write operations, the RLDRAM II memory requires the write clock (DK) to be center-aligned with the data bus (Q or DQ). This is implemented in Stratix II using the PLL phase shift feature. Two output clocks are created from the PLL, with a relative phase offset. The leading clock edge is used to clock out the DQ write data output pins to the memory, while the lagging clock edge is used to generate the DK/DK# write clock and CK/CK# memory output clocks. See Figure 3 on page 8 for the relationship between DQ and DK during a write.

Figure 4 on page 18 shows that the write side uses a PLL to generate the clocks listed in Table 15.

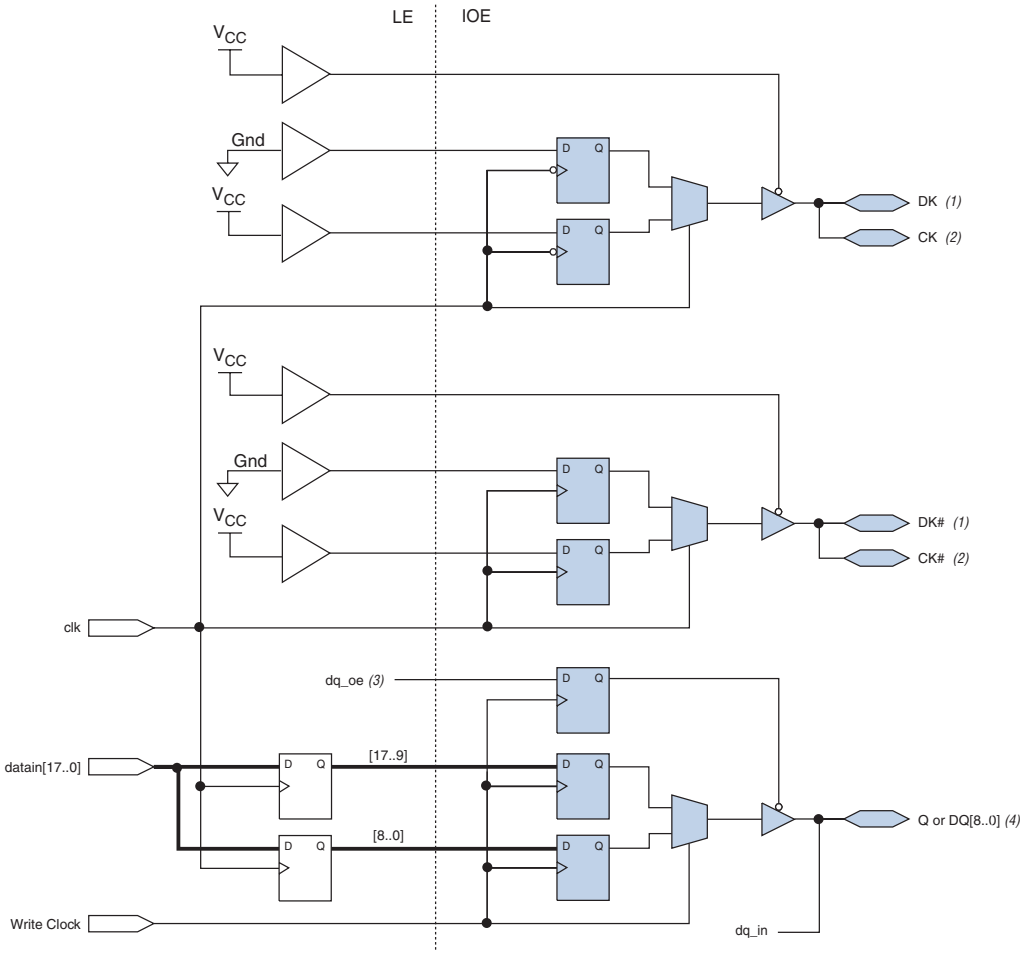
Table 15. Write Side PLL Clocks

Clock	Description
System clock	This is used for the memory controller and to generate the DK/DK# and CK/CK# signals.
Write clock (leading system clock) *(1)*	This is used in the data path to generate the DQ write signals.

Note to Table 15:
(1) Perform write timing calculation to determine the phase shift required for this write clock.

Figure 14 shows the RLDRAM II write data path in Stratix II devices. For best performance, you should use the same pin to generate both CK and DK signals or CK# and DK# signals so that the skew between CK and DK signals or CK# and DK# signals are minimized. You can generate these signals with two different pins after making sure that the Quartus II reported skew (from the t_{CO} times) meets the memory's t_{CKDK} specification. You can use any unused DQS/DQS# or LVDS pin-pair to generate CK/CK# (and DK/DK#) signals.

Figure 14. RLDRAM II Write Data Path in Stratix II Devices



Notes to Figure 14:

(1) RLDRAM II does not use the DQS pin during writes. Instead, DK and DK# are used and generated by user I/O pins via the DDR registers.

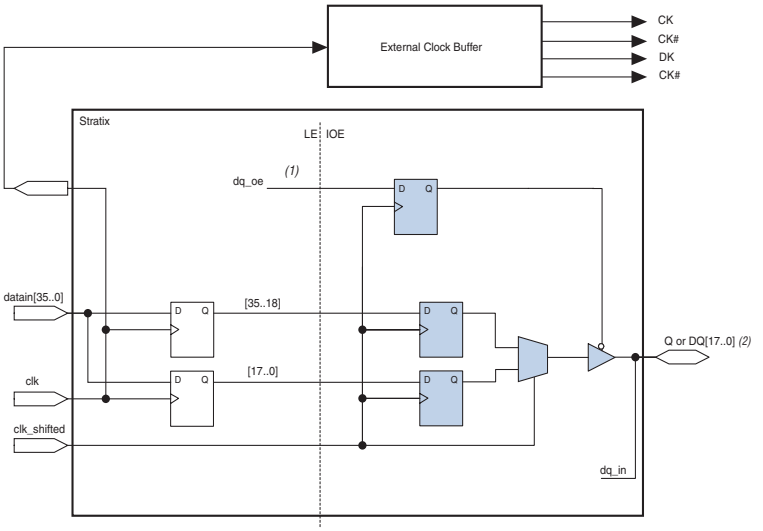
(2) To meet the t_{CKDK} specification from the RLDRAM II device, use the same pin pair to generate both CK and DK or CK# and DK#. You can use two different pins to generate CK and DK (and similarly CK# and DK#) if the t_{CO} difference between these two pins meet the t_{CKDK} specification. Perform IBIS simulation for loading effects on these signals when interfacing with multiple RLDRAM II devices.

(3) The `dq_oe` is active low in silicon. However, the Quartus II software implements it as active high and adds the inverter automatically during compilation.

(4) The RLDRAM II device may have the same or separate ports for data read and data. If separate, connect Q of the RLDRAM II device to the DQ pins of the Stratix II device and connect `dq_oe` to GND. You can use any of the user I/O pins in I/O banks 3, 4, 7, or 8 to connect to the D pins of the RLDRAM II device.

Figure 15 shows the RLDRAM II write data path in Stratix and Stratix GX devices. The CK, CK#, DK, and DK# are generated by an external clock buffer in this interface.

Figure 15. RLDRAM II Write Data Path in Stratix & Stratix GX Devices



Notes to Figure 15:

(1) The `dq_oe` is active low in silicon. However, the Quartus II software implements it as active high and adds the inverter automatically during compilation.

(2) The RLDRAM II device may have the same or separate ports for data read and data. If separate, connect Q of the RLDRAM II device to the DQ pins of the Stratix II device and connect `dq_oe` to GND. You can use any of the user I/O pins in I/O banks 3, 4, 7, or 8 to connect to the D pins of the RLDRAM II device.

Memory Timing Parameters

When writing to a memory, the FPGA needs to ensure that setup and hold times are met. These specifications (t_{DS} and t_{DH}) are obtained from the data sheet (300 ps for the 300 MHz RLDRAM II example). Additionally, the FPGA needs to provide a memory clock (CK/CK#) that meets the clock high/low time specifications. And finally, the skew between the DK write clock and CK output clock cannot exceed limits set by the memory. While the last parameter does not affect timing margins, it needs to be met for successful memory operation.

FPGA Timing Parameters

The timing paths within the FPGA for the DQ and DK outputs to memory are matched by design. Dedicated clock networks drive DDR IO structures to generate DQ and DQS. This results in minimal skew between these outputs. There are three skew parameters to be considered phase shift error, clock skew, and package skew.

The two clock networks used are driven by the same PLL, however with a relative phase-shift. The system (0°) clock is used to generate DK/DK# write clock, while a leading clock is used to generate DQ. You should perform a similar timing analysis like the one described below to calculate the write clock phase shift. Typical PLL uncertainties such as jitter and compensation error affect both clock networks equally, so these timing parameters do not affect write timing margins. However, since the clock generating DQ is phase-shifted, you need to account for the PLL phase-shift uncertainty (t_{PLL_PSERR} = ± 30 ps), listed in the Stratix II device data sheet when calculating DQ arrival times at the memory pins.

The Quartus II software models intra-clock skew, skew between nodes driven by the same dedicated clock network. However, skew between two such clock networks is not modeled and specified in the data sheet as an adder term. You should add this skew component to the propagation delays extracted from the Quartus II software.

For our RLDRAM II interface, the clock skew adder between two clock networks is specified as ± 50 ps ($t_{CLOCK\_SKEW\_ADDER}$). This uncertainty is accounted while calculating DQS arrival times at the memory pins.

The final skew component is package skew. As noted earlier, the Quartus II software (starting from version 5.0) models package trace delay for each pin on the device. Extracted propagation delays reflect any skew between output signals to the memory.

You need to perform this write timing analysis to find the optimum phase shift to generate your data pins. As a rule of thumb, you need a more negative phase shift if you want more setup time margin and vice versa. You need a more positive phase shift if you want more hold time margin.

Setup & Hold Margins Calculations

Table 16 shows the RLDRAM II write timing margin analysis when interfacing with Stratix II devices at 300 MHz. The board trace variations for the DQ and DQS pins is ± 20 ps (approximately ±0.12 inches of FR4 trace length variations). You can perform a similar timing analysis for our interface with another RLDRAM II memory by replacing the t_{DS} and t_{DH} values in Table 16 with those from your memory data sheet. This timing analysis applies for both RLDRAM II CIO and SIO devices.

Table 16. Write Timing Analysis for 300 MHz RLDRAM II Interface in EP2S60F1020C3 (Part 1 of 2)

	Parameter	Fast Corner Model	Slow Corner Model	Description
Memory specifications *(1)*	t_{DS}	0.300	0.300	Memory data setup requirement
	t_{DH}	0.300	0.300	Memory data hold requirement
FPGA specifications*(2)*	t_{HP}	1.485	1.485	Ideal half period minus 5% duty cycle distortion
	t_{PLL_JITTER}	0.000	0.000	Does not affect margin as the same PLL generates both write clocks (0° and –75°)
	t_{PLL_PSERR}	0.030	0.030	PLL Phase Shift Error
	$t_{CLOCK\_SKEW\_ADDER}$	0.050	0.050	Clock skew between two dedicated clock networks feeding IO banks on same side of the FPGA
	Minimum Clock Delay (Output) *(3)*, *(4)*, *(5)*	0.849	1.626	Minimum DQS t_{CO} from Quartus II (0° PLL output clock)
	Maximum Clock Delay (Output) *(3)*, *(4)*, *(5)*	0.849	1.626	Maximum DQS t_{CO} from Quartus II (0° PLL output clock)
	Minimum Data Delay (Output) *(3)*, *(4)*	–0.096	0.526	Minimum DQ t_{CO} from Quartus II (–105° PLL output clock)
	Maximum Data Delay (Output) *(3)*, *(4)*	0.077	1.096	Maximum DQ t_{CO} from Quartus II (–105° PLL output clock)
Board specifications	t_{EXT}	0.020	0.020	Board trace variations on the DQ and DQS lines
Timing calculations	t_{EARLY_CLOCK}	0.799	1.576	Earliest possible clock edge seen by memory device (minimum clock delay – t_{PLL_JITTER} – $t_{CLOCK\_SKEW\_ADDER}$)
	t_{LATE_CLOCK}	0.849	1.676	Latest possible clock edge seen by memory device (maximum clock delay + t_{PLL_JITTER} + $t_{CLOCK\_SKEW\_ADDER}$)
	$t_{EARLY\_DATA\_INVALID}$	1.539	2.161	Time for earliest data to become invalid for sampling at the memory input pins (t_{HP} + minimum data delay – t_{PLL_PSERR})

Table 16. Write Timing Analysis for 300 MHz RLDRAM II Interface in EP2S60F1020C3 (Part 2 of 2)

	Parameter	Fast Corner Model	Slow Corner Model	Description
Results	$t_{LATE\_DATA\_VALID}$	0.107	1.126	Time for latest data to become valid for sampling at the memory input pins (maximum data delay + t_{PLL_PSERR})
	Write setup timing margin	0.372	0.130	$t_{EARLY\_CLOCK} - t_{LATE\_DATA\_VALID} - t_{DS} - t_{EXT}$
	Write hold timing margin	0.320	0.165	$t_{EARLY\_DATA\_INVALID} - t_{LATE\_CLOCK} - t_{DH} - t_{EXT}$
	Total margin	0.692	0.295	Setup margin + hold margin

Notes for Table 16:

(1) The memory numbers used here come from Micron MT49H8M36FM-33.

(2) This analysis is performed with FPGA timing parameters for an EP2S60F1020C3. You should use this template to analyze timing for your preferred Stratix II density-package combination. For more information on FPGA specifications, refer to the *DC & Switching Characteristics* in *Volume 1* of the *Stratix II Handbook*.

(3) These numbers are from the Quartus II software version 5.1 using the Altera RLDRAM II MegaCore function version 1.0.0.

(4) Package trace skews are modeled by the Quartus II software.

(5) For CK/CK# signals, choose pins with matched t_{CO}. You can use any unused DQS/DQS# or LVDS pin-pairs.

Table 17 shows the RLDRAM II example write timing margin analysis for a Stratix EP1S25 device.

Table 17. Write Timing Analysis for a Stratix EP1S25 -5 Speed Grade Device

Parameter	Specification	200 MHz	Description
Memory specifications	t_{CK}	5 ns	Clock period
	$t_{DS} = t_{DH}$ *(1)*	0.4 ns	DQ and DM setup and hold time from the memory data sheet
FPGA specifications	t_{IOSKEW}	0.160 ns	Absolute value of the difference in clock-to-out times (t_{CO}) between any two output registers on the top/bottom of the device fed by a common clock source
	$t_{CLKSKEW}$	0.150 ns	Skew between two PLL outputs
	t_{DCD}	0.250 ns	Duty cycle distortion (5% of clock period)
Board specification	t_{EXT}	0.05 ns	Board trace variations for the DQ and DQS lines (166 ps per inch for an FR4 trace)
Timing calculation	t_{SHIFT_MIN}	1.1 ns	Minimum shift from the PLL ($0.25 \times t_{CK}$ (90° phase shift) – $t_{CLKSKEW}$)
	t_{SHIFT_MAX}	1.4 ns	Maximum shift from the PLL ($0.25 \times t_{CK}$ (90° phase shift) + $t_{CLKSKEW}$)
Results	Write setup timing margin	0.240 ns	$t_{SHIFT\_MIN} - t_{DCD} - t_{IOSKEW} - t_{EXT} - t_{DS}$
	Write hold timing margin	0.240 ns	$0.5 \times t_{CK} - t_{SHIFT\_MAX} - t_{CDCD} - t_{IOSKEW} - t_{EXT} - t_{DH}$

Note for Table 17:
(1) The memory numbers used here come from Micron MT49H16M18/C-5.

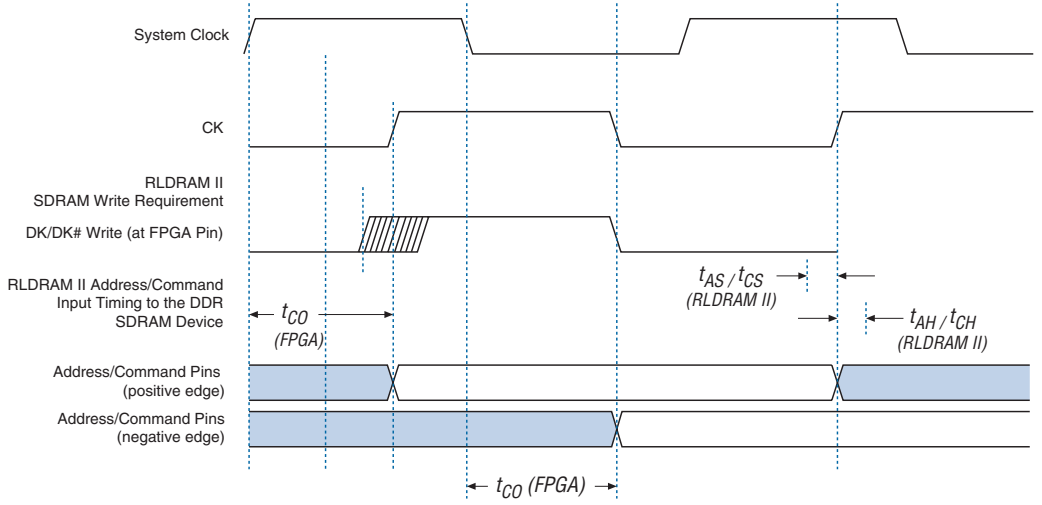
Command & Address Timing

Command and address signals are generated from the system clock (or another clock) in single data rate. The command and address signals must meet the setup and hold time requirement with respect to the rising edge of the CK signal at the RLDRAM II device. The FPGA also generates the CK signal from the system clock either directly or via an external clock buffer. Depending on the location of the registers for the commands and addresses, you may need to use a different system clock edge or add a phase shift on the system clock to make sure that these signals meet the setup and hold time requirement at the RLDRAM II device. This section outlines the commands and addresses timing considerations.

For example, if you place the command and address registers clocked by the system clock in the IOE, the command and address signals at the FPGA change at the same time as the CK signal since they are both generated from the system clock either directly or via an external clock buffer. The delays from the IOE registers to the pins are then similar (exactly the same if there is no clock skew). The Altera RLDRAM II controller MegaCore constrains the command and address registers in the IOE.

In this example, the command and address signals are edge-aligned with the CK signal (if there are no variations in the package or board trace length of the system for the different signals). This means that the address and command signals cannot meet the setup time requirement of the RLDRAM II device. In order to meet the setup and hold time requirement, you have to use the negative edge of the system clocks to generate the command and address signals if you want to place the command and address registers in the IOE. Figure 16 shows the command and address timing and how the system clock edge affects how the signals meet the RLDRAM II t_{AS}/t_{CS} and t_{AH}/t_{CH} requirements.

Figure 16. Command & Address Timing Using the IOE Registers *Note (1)*



Note to Figure 16:
(1) In this waveform, there is no clock skew or any variations in package and board trace length.

You can perform timing analysis for command and address signals similar to the write data timing analysis to find the optimal phase shift to generate the command and address signals. The only difference between

the write data timing analysis and the command and address timing analysis is that the command and address timing signals are single data rate whereas the data signals are double data rate.

The default clock for the command and address signals in the RLDRAM II MegaCore function is the falling edge of the system clock. Quartus II reports the t_{CO} does not automatically verify source-synchronous outputs will meet the setup/hold requirements at the destination, so you will need to subtract ½ clock cycle from the t_{CO} reported. Table 18 shows the t_{CO} reported by Quartus II and the actual t_{CO} if you relate this t_{CO} with the CK/CK# t_{CO}.

Table 18. Reported and Adjusted Command/Address t_{co} Note (1)

	Fast Timing Model	Slow Timing Model	Units
Quartus II reported minimum command/address t_{CO}	0.971	1.720	ps
Quartus II reported maximum command/address t_{CO}	1.044	1.792	ps
Adjusted minimum command/address t_{CO}	-0.696	0.053	ps
Adjusted maximum command/address t_{CO}	-0.623	0.126	ps

Note to Table 18:

(1) Quartus II does not automatically verify source-synchronous output will meet the setup/hold requirements at the destination. Since this is a system timing analysis, you need to find the actual t_{CO} compared to the t_{CO} for the CK/CK# signals. The adjusted t_{CO} numbers show the ½ clock cycle reported from the Quartus II reported t_{CO} to account for the falling edge signal.

Table 19 shows an example of the command and address timing analysis for an EP2S60F1020C3 interfacing with 300-MHz RLDRAM II device.

Table 19. Command/Address Timing Analysis for 300-MHz RLDRAM II Interface in EP2S60F1020C3 (Part 1 of 3)

	Parameter	Fast Corner Model	Slow Corner Model	Description
Memory Specifications *(1)*	t_{DS}	0.500	0.500	Memory Command/Address Setup Requirement
	t_{DH}	0.500	0.500	Memory Command/Address Hold Requirement

Table 19. Command/Address Timing Analysis for 300-MHz RLDRAM II Interface in EP2S60F1020C3 (Part 2 of 3)

	Parameter	Fast Corner Model	Slow Corner Model	Description
FPGA Specifications *(2)*	t_{CK}	2.997	2.997	Clock period - 10% duty cycle distortion
	t_{PLL_JITTER}	0.000	0.000	Does not affect margin as the same clock is used to generate CK and command/address signals
	t_{PLL_PSERR}	0.030	0.030	PLL Phase Shift Error (On –90° clock output)
	$t_{CLOCK\_SKEW\_ADDER}$ *(3)*	0.000	0.000	Same clock is used to generate CK and command/address signals, so the skew should be accounted for in Quartus II
	Minimum Clock Delay (Output) *(4)*, *(5)*	0.849	1.626	Minimum CK t_{CO} from Quartus II
	Maximum Clock Delay (Output) *(4)*, *(5)*	0.849	1.626	Maximum CK t_{CO} from Quartus II
	Minimum Command/Address Delay (Output) *(5)*, *(6)*	-0.696	0.053	Minimum Command/Address t_{CO}
Board Specifications	Maximum Command/Address Delay (Output) *(5)*, *(6)*	-0.623	0.126	Maximum Command/Address t_{CO}
Timing Calculations	t_{EXT}	0.020	0.020	Board trace variations on the DQ and command/address lines
	t_{EARLY_CLOCK}	0.849	1.626	Earliest possible clock edge seen by memory device (minimum clock delay – t_{PLL_JITTER} – $t_{CLOCK\_SKEW\_ADDER}$)
	t_{LATE_CLOCK}	0.849	1.626	Latest possible clock edge seen by memory device (maximum clock delay + t_{PLL_JITTER} + $t_{CLOCK\_SKEW\_ADDER}$)
	$t_{EARLY\_CMD\_ADD\_INVALID}$	2.272	3.021	Time for earliest command/address to become invalid for sampling at the memory input pins (t_{HP} + Minimum Data Delay – t_{PLL_PSERR})
Results	$t_{LATE\_CMD\_ADD\_VALID}$	-0.593	0.156	Time for latest command/address to become valid for sampling at the memory input pins (Maximum Data Delay + t_{PLL_PSERR})
	Command/address setup timing margin	0.922	0.951	t_{EARLY_CLOCK} – $t_{LATE\_CMD\_ADD\_VALID}$ – t_{DS} – t_{EXT}
	Command/address hold timing margin	0.903	0.875	$t_{EARLY\_CMD\_ADD\_INVALID}$ – t_{LATE_CLOCK} – t_{DH} – t_{EXT}

Table 19. Command/Address Timing Analysis for 300-MHz RLDRAM II Interface in EP2S60F1020C3 (Part 3 of 3)

	Parameter	Fast Corner Model	Slow Corner Model	Description
	Total margin	1.824	1.825	Setup margin + Hold margin

Notes to Table 19:

(1) The memory numbers used here come from Micron MT49H8M36FM-33.

(2) This analysis is performed with FPGA timing parameters for an EP2S60F1020C3. You should use this template to analyze timing for your preferred Stratix II density-package combination. Refer to the *DC & Switching Characteristics* chapter in *Volume 1* of the *Stratix II Handbook*.

(3) If you are using a dedicated clock to generate command and address signals, then this parameter is not 0. Refer to the *DC & Switching Characteristics* chapter in *Volume 1* of the *Stratix II Handbook* for the most up-to-date clock skew adder specification.

(4) Command and Address signals are generated on the falling edge of the system clock in this example. This value is adjusted from Quartus II reported t_{CO} as Quartus II reports the t_{CO} based on the rising edge of the input clock regardless of how the signal is generated.

(5) These numbers are from the Quartus II software, version 5.1 using the Altera RLDRAM II MegaCore function version 1.0.0.

(6) Package trace skews are modeled by the Quartus II software.

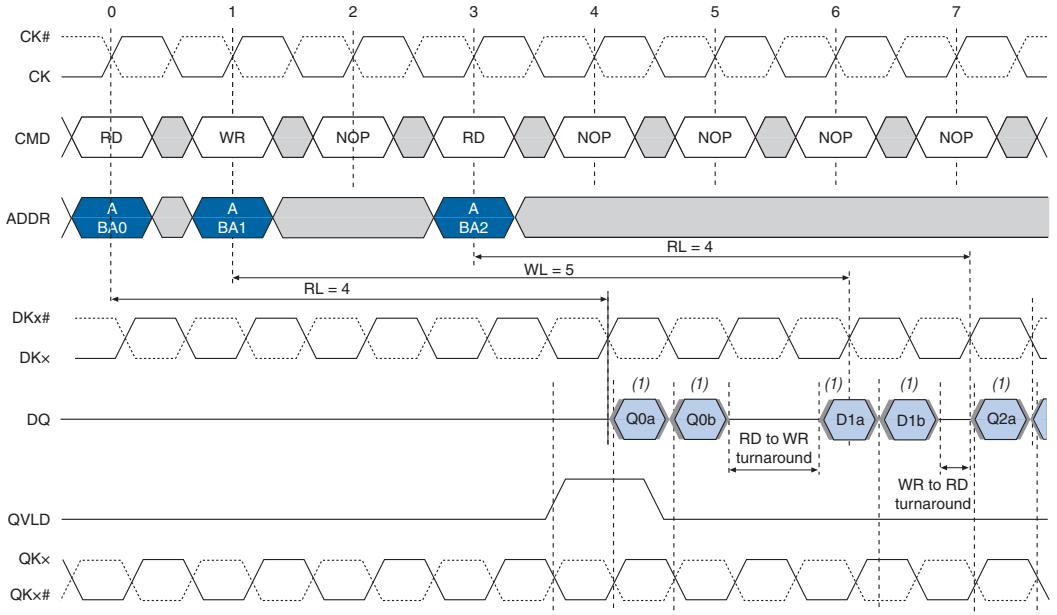
Bus Turnaround Timing Analysis

When using Stratix II, Stratix, or Stratix GX devices to interface with RLDRAM II CIO devices, you need to insert a no-operation command when switching from read to write or from write to read since the Stratix II device cannot turn the bus around fast enough.

Figure 17 shows a timing waveform of an example of read and write operations for RLDRAM II CIO devices when burst length equals two. The figure shows a read, followed by a write and then another read. You need to calculate whether you need to insert a no-operation command in between a read to write or a write to read transition.

For RLDRAM II CIO devices operating in burst length of four, you always need to insert a no-operation per memory specification when switching from read to write and write to read for uninterrupted read and write operations. You still need to insert one extra no-operation command if the bus turnaround of the FPGA is going to cause contention.

Figure 17. Bus Turnaround in RLDRAM II Interfaces When Burst Length Equals Two



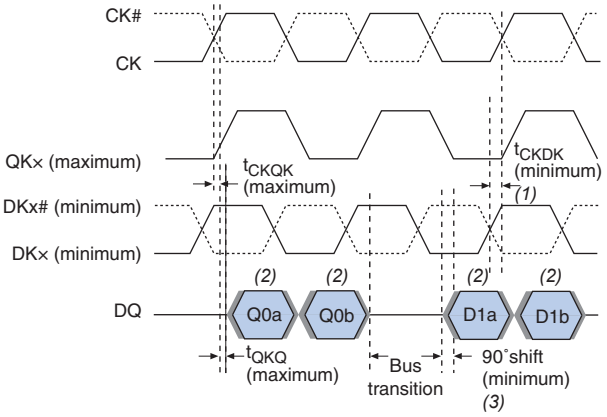
Note to Figure 17:
(1) The gray section shows when data is changing. The blue section shows data valid.

As shown in Figure 17, the read-to-write transition (RD to WR turnaround) is about 0.75 clock cycle maximum and the write-to-read transition (WR to RD turnaround) is about 0.25 clock cycle maximum. You need to make sure that the FPGA can turn off and turn on the bus quickly enough to avoid contention. The following sections describe the calculation in detail for burst-of-two operations.

Read-to-Write Transition

In order to calculate the fastest (worst) bus turnaround time during a read-to-write transition, use the maximum RLDRAM II timing delays and the minimum FPGA timing delays. Figure 18 shows part of the timing waveform shown in Figure 17, focusing on the read-to-write transition part.

Figure 18. Worst Case Read-to-Write Transition Timing Waveform



Notes to Figure 18:

(1) Even though CK and DK signals are derived the same way, there may be skew either from the board or from the external clock buffer if you are using Stratix or Stratix GX devices.

(2) The gray section shows when data is changing. The blue section shows that data is valid.

(3) The minimum 90° phase shift accounts for PLL output skew, I/O output skew, offset error from the external clock buffer, and board trace length mismatch.

As shown in Figure 18, when you use the maximum RLDRAM II timing delays, the read data is shifted to the right. Figure 18 also shows that when you use the minimum FPGA timing delays, the write data is shifted to the left. As a result, the bus transition time shrinks from the optimal 0.75 clock cycle maximum transition width. To determine whether a no-operation command needs to be inserted when switching from a read to a write operation, you need to calculate the actual bus transition width of the interface.

For example, consider a 200 MHz RLDRAM II interface in a Stratix device using a burst length of two with the following RLDRAM II specifications:

- t_{CKQK}: 0.5 ns
- t_{QKQ}: 0.4 ns
- Maximum data width: $(1.1 \times 0.55 \times t_{CK}) + (2 \times t_{QKQ})$
- Longest board trace length skew: + 20 ps

Using CK as the reference clock, the read data is invalid after a time period as calculated below:

Read data invalid period=
t_{CKQK} + t_{QKQ} + burst length × maximum data width =
0.5 ns + 0.4 ns + 2 × 3.825 ns + 0.020 ns = 8.57 ns

To determine how long after the first CK edge the invalid time occurs, divide the time period from the previous equation by the time it takes to send the data burst.

Invalid time = Read data invalid period / (burst length / 2 × t_{CK})
8.57 ns/ 5 ns per clock cycle = 1.714 clock cycles

This means that read data can be valid 0.714 clock cycles later than when it is supposed to end.

On the write side of the transition, the DK signal can arrive earlier than CK by $t_{CLKBUFVAR}$ (variations from the external clock buffer) which consists of:

- $t_{CLKBUFSKEW}$ (the skew between two outputs from the external clock buffer)
- $t_{CLKBUFJITTER}$ (jitter from the external clock buffer)
- $t_{CLKBUFDCD}$ (duty cycle distortion from the external clock buffer).

If your design interfaces Stratix II devices with RLDRAM II devices, it will not have a $t_{CLKBUFVAR}$ parameter. Instead, you should account for the Stratix II PLL jitter and duty cycle distortion.

The 90º phase-shifted clock from the FPGA that generates the write data can also be skewed from its ideal phase by $t_{PLLSKEW}$ (the skew between two outputs of the PLL in the FPGA).

IDT clock buffer IDT5T2110 has the following specifications (from its data sheet):

- $t_{CLKBUFSKEW}$: ± 0.1 ns
- $t_{CLKBUFJITTER}$: ± 0.125 ns
- $t_{CLKBUFDCD}$: ± 0.1 ns

In addition, read data can also be skewed by the I/O output skew specification from the FPGA (t_{IOSKEW}), and the board trace length variations (t_{EXT}).

Stratix EP1S25 characterization data shows the following:

- $t_{PLLSKEW}$: ± 0.150 ns
- t_{IOSKEW}: ± 0.160 ns

Use the following equation to determine the earliest the data can arrive at the DQ pins to drive the bus from its ideal time:

$$t_{CLKBUFSKEW} + t_{CLKBUFJITTER} + t_{CLKBUFDCD} + t_{PLLSKEW} + t_{IOSKEW} = 0.1 \text{ ns} + 0.125 \text{ ns} + 0.1 \text{ ns} + 0.150 \text{ ns} + 0.160 \text{ ns} = 0.635 \text{ ns}$$

This means that the data can come earlier by 0.127 clock cycle (0.635 ns/ 5 ns clock period)

Based on this example, you need an extra no-operation command because you have a maximum of 0.75 clock cycle turnaround time and the variations require 0.841 clock cycles (0.714 + 0.127 clock cycles). Therefore, you need to add a no-operation command when switching from a read to a write.

You can perform a similar calculation for Stratix II FPGAs in C3 speed grade interfacing with 300-MHz RLDRAM II devices using the following parameters:

- $t_{CLOCK\_SKEW\_ADDER}$ (clock skew between two dedicated clock networks feeding I/O banks on the same side of the FPGA) = ± 50 ps
- t_{PLL_PSERR} (PLL phase-shift error) = ± 30 ps
- Quartus reported I/O skew (by getting the t_{CO} differences for the OE signals for the DQ pins): 73 ps
- Quartus reported clock skew (if you are using two different pins to generate CK and DK). In this example, the design has 0 ps skew.
- t_{DCD} (duty cycle distortion from the PLL) = 167 ps (5% of 300 MHz)

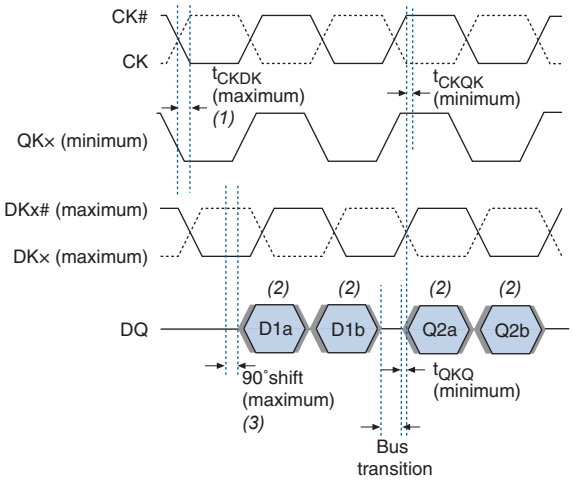
The earliest data arrival is the summation of all the parameters above. In this example, the total is 320 ps. This means data can come earlier by 0.096 clock cycle. The read data invalid for 300-MHz RLDRAM II is 5.6 ns (1.682 clock cycles). With this example, you need an extra no-operation command in 300-MHz RLDRAM II interfaces since data can come as early as 0.778 clock cycles (0.682 + 0.0946 clock cycles), which exceeds the 0.75 clock cycle turn-around time.

PLL jitter in the Stratix II timing calculation is considered zero since data and clock are generated the same way and so they will jitter together.

Write-to-Read Transition

Similarly, in order to calculate the fastest (worst) bus turnaround time during a write-to-read transition, you need to use the maximum FPGA timing delays and the minimum RLDRAM II timing delays. Figure 19 shows part of the timing waveform shown in Figure 17, focusing on the write-to-read transition part.

Figure 19. Worst Case Write-to-Read Transition Timing Waveform



Notes to Figure 19:

(1) Even though CK and DK are derived the same way, there may be skew either from the board or from the external clock buffer.

(2) The gray section shows when data is changing. The blue section shows that data is valid.

(3) The maximum 90º phase shift accounts for PLL output skew, I/O output skew, offset error from the external clock buffer, and board trace length mismatch.

As shown in Figure 19, when you use the maximum FPGA timing delays, the write data is shifted to the right. Figure 18 also shows that when you use the minimum RLDRAM II timing delays, the read data is shifted to the left. As a result, the bus transition period shrinks from the optimal 0.25 clock cycle maximum transition width. To determine whether a no-operation command needs to be inserted when switching from a read to a write operation, you need to calculate the bus transition width of the interface.

Using the same RLDRAM II and FPGA specification as in the "Read-to-Write Transition" section as an example, you can calculate whether you need an extra no-operation when switching from a write to a read.

Using CK as the reference clock, the write data is invalid after a time period for Stratix devices interfacing with 200-MHz RLDRAM II devices, as calculated below:

$t_{CLKBUFSKEW} + t_{CLKBUFJITTER} + t_{CLKBUFDCD} + t_{IOSKEW} + t_{PLLSKEW} + t_{EXT}$ + burst length × maximum data width = 0.1 ns + 0.125 ns + 0.1 ns + 0.150 ns + 0.160 ns + 0.02 ns + (2 × 1.1 x 0.55 × 5 ns)= 6.705 ns

To determine how long after the first CK edge the invalid time occurs, divide the time period from the previous equation by the clock period.

6.735 ns/5 ns per clock cycle = 1.347 clock cycles

This means that read data can be valid 0.347 clock cycles later than when it is supposed to end. This already exceeds the ideal 0.25 clock cycles bus transition period so you need to add one no-operation command every time you switch from a write to a read.

Similarly, the calculation for Stratix II devices interfacing with 300-MHz RLDRAM II devices would be as follows:

$t_{CLOCK\_SKEW\_ADDER} + t_{PLL\_PSERR}$ + Quartus reported I/O skew + Quartus reported clock skew + $t_{EXT} + t_{DCD}$ = 0.050 ns + 0.030 ns + 0.073 ns + 0.00 ns + 0.020 ns + 0.167 + (2 × 1.1 × 0.55 × 3.33 ns) = 4.369 ns

This means data can be valid 0.312 clock cycles later (4.369/4.369 – 1 clock cycle) than when it is supposed to end. You also need an extra no-operation command here as it exceeds the 0.25 clock cycle bus transition period.

The same parameters mentioned in the "Read-to-Write Transition" on page 51 section for Stratix II are also applicable here.

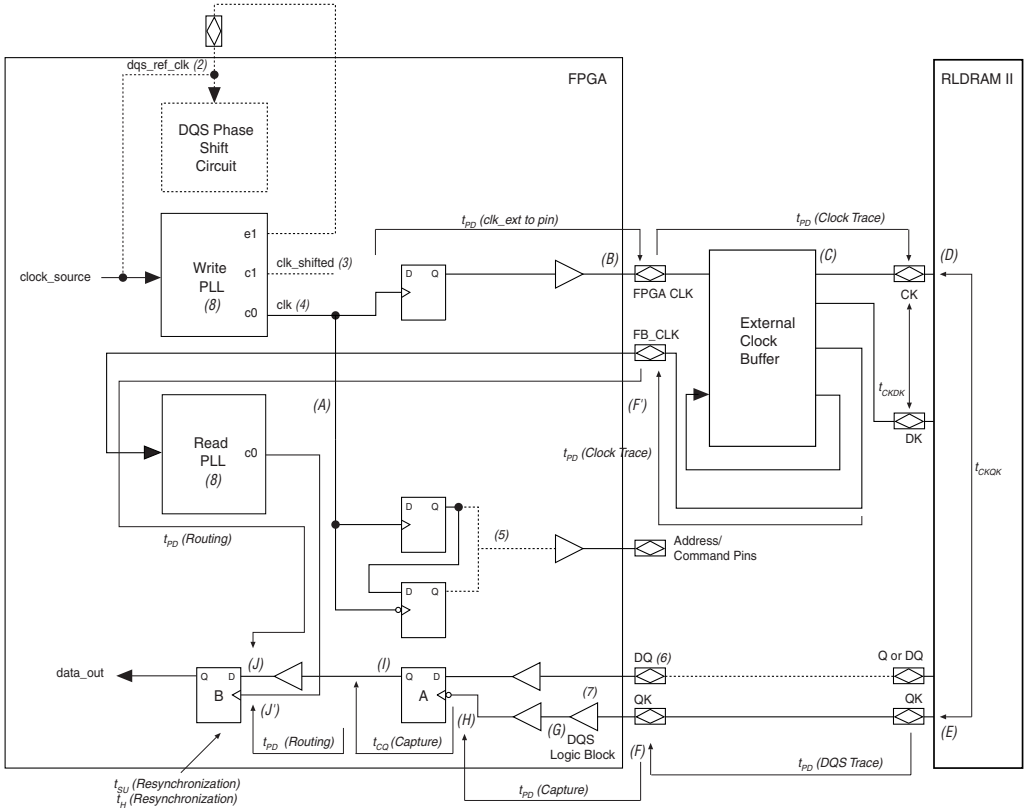
Round Trip Delay Calculation

Read data is captured into the DDR registers using the QK signal as a clock. Therefore, data must be transferred from the QK clock domain to the system clock domain (resynchronization), to present data from the input registers synchronously at the local-side interface. To determine the point at which the data can be reliably resynchronized, calculate the minimum and maximum round trip delay. You can then determine what resynchronization logic to use for your system.

This timing analysis applies for RLDRAM II interface with Stratix and Stratix GX devices. In the RLDRAM II interface with Stratix II devices, QVLD signal is used with a dual-clock FIFO for resynchronization and its timing is analyzed by the Quartus II software.

Figure 20 shows the round trip delay for Stratix and Stratix GX devices. Figure 20 shows the path from the FPGA clock to the RLDRAM II and back to the Stratix or Stratix GX device (input to register B). This analysis is required to reliably transfer data from register A (in the IOE) to register B (in the LE). You can also use the shifted DQS signals for resynchronization, but this method is not discussed in this application note.

Figure 20. RLDRAM II Round Trip Delay Illustration in Stratix & Stratix GX Devices *Note (1)*



Notes to Figure 20:

(1) The nodes for the round trip delay analysis are marked with letters (A) through (I).
(2) The `dqs_ref_clk` input for Stratix and Stratix GX devices must come from an input clock pin.
(3) The `clk_shifted` signal is shown for completeness, but it is not needed in the timing analysis for round-trip delay or address/command timing.
(4) `clk` is the system clock.
(5) You can clock the address/command register with either a rising edge or falling edge of the `clk` signal.
(6) You can either connect the RLDRAM II CIO DQ pins to the Stratix or Stratix GX DQ pins or connect the RLDRAM II SIO Q pins to the Stratix or Stratix GX DQ pins.
(7) The DQS phase-shift reference circuit controls the phase shift on the DQS signal dynamically. The control path is not shown and its operation is user transparent.
(8) This PLL is in normal mode.

Register A in Figure 20 represents the DDR capture logic. The Q output from register A represents the point at which the read data has been converted from DDR to single data rate (SDR). At the output of register A, the data is already at single data rate, but is still in the QK clock domain. DQ_H (DQ data during QK high) is sampled on the positive edge of the phase-shifted QK pulse, but re-sampled on the negative edge of the phase-shifted QK pulse, to align it with DQ_L (DQ data during DQS low).

Once sampled by the negative edge of the phase-shifted QK signal, DQ_L and DQ_H are available for resynchronization. To sample the Q output of register A into register B, you need the time relationship between register B's clock input and the D input. This time relationship depends on the phase relationship between QK and clock and involves the following steps:

1. Calculate the system's round-trip delay (described below).
2. Select a resynchronization phase of the system clock or other available clock that reliably samples the Q output of register A, based on the calculated safe resynchronization window.
3. Apply the correct clock edge for the resynchronization logic in the memory controller.

The feedback clock and the read PLL shown in Figure 20 improve the resynchronization process. This feedback clock is from the external clock buffer that is also used to provide CK, CK#, DK, and DK# signals to the RLDRAM II devices.

The read PLL needs to be in normal mode such that its output is in phase with the input to the PLL (if there is no phase-shifting). The PLL input is skewed by the RLDRAM II ± t_{CKQK} value plus any board trace skew between QK or CK and the `FB_CLK` traces. The PLL can then be used to compensate for the delay between IOE register to the LE register and be used to synchronize the data from the QK clock domain to the feedback clock domain.

The feedback clock lags the system clock by the board trace length for the CK signal plus the board trace length for the QK signal delay. You can calculate whether register outputs clocked by the feedback clock need another resynchronization stage before getting to the system clock domain.

To determine the data timing at the D input of register B relative to the clock, you have to know when the following occur:

- CK input clock arrives at the RLDRAM II memory
- FB_CLK signal arrives at the clock input of the read PLL
- Data arrives at the Q output of register A
- Data arrives at the D input of register B
- Clock arrives at the clock input of register B

There are two round trip delays to consider: one for the CK signal going to the RLDRAM II devices and the QK signal coming back to the Stratix II devices and another one for the FB_CLK signal.

You need the maximum and minimum values (taking into account PVT variations) to calculate the following delays:

- Clock-to-out delays for the CK signal from the FPGA
- CK board trace lengths
- The external clock buffer jitter and compensation error from the external clock buffer
- QK board trace lengths
- Register to register delays between the registers in the feedback clock domain and the registers in the system clock domain

For the FB_CLK signal round trip delay, you need the maximum and minimum values (taking into account PVT variations) to calculate the following delays:

- Clock-to-out delays for the CK signal from the FPGA
- Board trace lengths from the CK pin to the FB_CLK pin
- The external clock buffer jitter and compensation error from the external clock buffer
- Delays from the FB_CLK input to input register B

To determine the point at which the data can be reliably resynchronized, calculate the minimum and maximum round-trip delay, and then determine what resynchronization logic to use for your system. Remember to take into account PVT variations.

Delay (A) to (B) is the clock-to-out time to generate the clock signals to the RLDRAM II device.

Delay (B) to (C) is the trace delay for the external clock buffer.

Delay (C) to (D) is the trace delay from the external clock buffer to the RLDRAM II device. Since the clock is an external feedback clock buffer, the signal going into the RLDRAM II device should be aligned with the feedback clock going into the clock buffer. However, you should account

for any trace length mismatch, the clock buffer jitter, and compensation in the round trip delay calculation. The jitter and compensation error are shown in Figure 20.

Delay (D) to (E) is the relationship between the clock and the QK clock timing during reads. This is t_{CKQK} in RLDRAM II specifications, nominally 0, but typically varies by ±0.3 ns for 300 MHz RLDRAM II devices. The QK output strobe is only guaranteed to be within $\pm t_{CKQK}$ of the clock input. So use t_{CKQK} (maximum) for calculating the maximum round trip delay; t_{CKQK} (minimum) for calculating the minimum delay.

Delay (E) to (F) is the trace delay for QK, which typically matches the trace delay for the Q or DQ signals in the same byte group. To calculate the maximum round trip delay, use the byte group with the longest trace lengths; for the minimum use the shortest. Trace lengths between different byte groups do not have to be tightly matched, but a difference between the longest and shortest decreases the safe resynchronization window within which the data can be reliably resynchronized. PLL jitter and clock duty cycle also affect the round trip delay. Add each of these delays to the maximum value and subtract from the minimum. PLL jitter and clock duty cycle are not shown in Figure 20.

Delay (F) to (G) is the 90° phase shift delay including the DLL jitter and DLL phase shift error.

Delay (G) to (H) is the delay from the QK pin to the IOE register.

Delay (H) to (I) is the micro clock-to-out time for the IOE registers

Delay (I) to (J) is the delay from the IOE registers to the LE resynchronization registers. To calculate the maximum round trip delay, use the longest delay for the whole interface and use the shortest delay for the whole interface for the minimum round trip delay.

Delay (B) to (F′) is the delay that mimics the CK trace length to the RLDRAM II device and QK trace length back to the FPGA.

Delay (F′) to (J′) is the delay from the `FB_CLK` pin to the clock port of the LE resynchronization registers. This delay includes the PLL compensation delay.

Board Design Guidelines

This section provides general guidelines for board design when using the Stratix II, Stratix, and Stratix GX devices to interface with RLDRAM II devices. It also provides information about decoupling capacitance. The following general guidelines apply when designing with Stratix II, Stratix, and Stratix GX devices and RLDRAM II devices.

- Keep the memory component and the Stratix II, Stratix, and Stratix GX devices close together. The routing length between Stratix II, Stratix, and Stratix GX devices and the memory component should be within 4.5 inches. The total distance between the Stratix II device to the V_{TT} termination resistor must not exceed six inches when routed as fly-by.
- Pull-up resistors R_T to V_{TT} (0.90 V) are required for data, data strobe, data mask, address, and control signals and should be located after the end of the memory structure in a fly-by termination scheme. Routing length to the pull-ups is less critical, but most designs require 0.5 to 1 inch to route. Figure 21 shows this termination scheme.

These termination instructions are guidelines only. The best way to predict that the termination arrangement meets your requirements is to simulate your design, including the PCB and device packages.

Figure 21. Termination Scheme for RLDRAM II CIO Devices

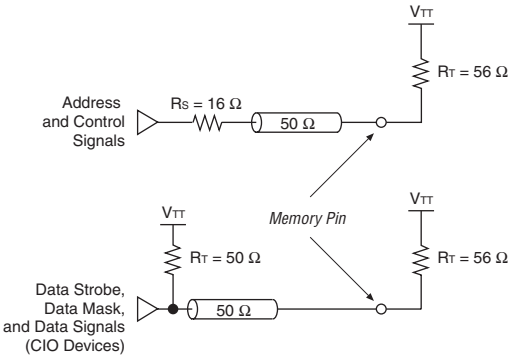
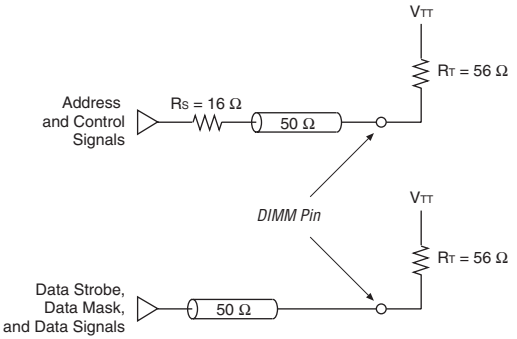


Figure 22. Termination Scheme for RLDRAM II SIO Devices



- Match routing for data byte-groups as closely as possible on the PCB. For example, you should match the timing skews for data groups (including data strobe and data mask signals) as closely as possible. These should be 17 ps (0.1 inch). Altera also recommends matching the timing skews of different data byte-groups. These should also be 17 ps (0.1 inch) to 85 ps (0.5 inch). To match the routing, take the longest trace and match the rest of the signals with the longest trace. Also, you should account for vias, which have electrical length, in all trace balancing configurations. Proper routing topology is best achieved when all point-to-point connections match not only in physical length but also in electrical length.
- Keep each clock trace segment length as short as possible. The total length of a clock trace (all segments), including passive components, should be less than three inches.

- Series resistors in a clock network must be as close to the source as possible.
- Route clocks with differential pairs next to each other throughout the trace length. The trace width should be five mils, and spacing between the positive and negative traces should be five mils. Spacing between these traces and other signals should be a minimum of 30 mils, and they should be matched in length.
- Avoid routing signals across split planes. Altera recommends controlling returns at high frequencies. Also, avoid routing memory signals any closer than 0.025 inches from PCI or system clocks. Avoid routing memory signals close to system reset signals to reduce crosstalk.

- When using resistor networks, Altera recommends confining the address and control signals to separate physical packages from data signals. To eliminate crosstalk within R-pack resistors, the address, control, and data lines should not share R-pack series resistors. Use series and pull-up resistors with 1 to 2% network tolerances.
- The distance between the RLDRAM II device pin and the termination resistor pack (to 0.90 V) less than 1.25 inches.
- The lengths that all signals must travel should be within ±0.250 inches. The spacing between signal traces must be 5 mils for parallel traces less than 0.5 inches long, 10 mils for parallel runs between 0.5 and 1.0 inches, and 15 mils for parallel runs between 1.0 and 6.0 inches.
- Maintain 25 mils of space between all RLDRAM II signal traces.
- The spacing between RLDRAM II address lines must be 10 mils for parallel traces less than 0.5 inches long, 15 mils for parallel runs between 0.5 and 1.0 inches, and 20 mils for parallel runs between 1.0 and 6.0 inches.

Decoupling Capacitance

Traditional methods for providing decoupling involve placing capacitors in locations that are convenient based on the routing of the board, and applying some predetermined ratio of capacitors to driver pins. However, the higher switching speeds of DDR make typical ratios less useful. Perform careful planning and analysis to ensure that sufficient decoupling is provided. The amount of capacitance on a board is usually not the critical limiting factor in designing a decoupling system. Typically, the amount of inductance in the capacitor leads and the vias attaching the capacitors to the power and ground planes creates limitations. Altera recommends using 0.1-µF capacitors in an 0603-sized package to provide sufficient capacitance without adding too much inductance. Make V_{TT} voltage decoupling on the motherboard close to the parallel pull-up resistors. Connect the decoupling capacitors between V_{TT} and ground. The Stratix II, Stratix, and Stratix GX memory interface board has a 0.1-µF capacitor for every other V_{TT} pin. The Stratix and Stratix GX memory interface board also has 0.1- and 0.01-µF capacitors for every VDD and VDDQ pin.

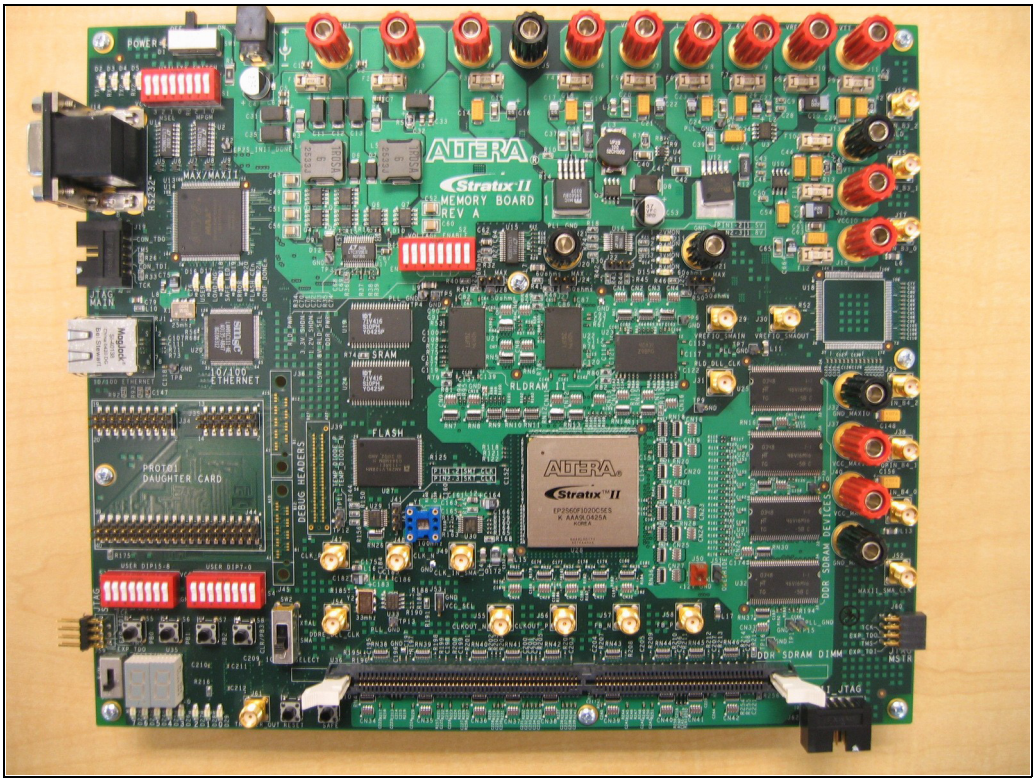
Stratix-Series Memory Board I

Altera produces the Stratix II memory board I and Stratix memory board I to demonstrate DDR SDRAM and RLDRAM II interfaces with the Stratix-series device family. The Stratix II memory board I includes a Stratix II EP2S60F1020C4 device interfacing with the following external memory devices:

- Four DDR SDRAM ×16 devices connected to the Stratix II side I/O banks of banks 1 and 2. The boards will use one of the following third-party memory devices: Micron MT46V16M16TG-5B, Infineon HYB25D25616OBT-5A, or Samsung K4H561638F-TCCC.
- One DDR SDRAM module connected to the Stratix II I/O banks 7 and 8. The boards will use one of the following third-party memory devices: Micron MT9VDDT3272AG-40B, Infineon HYS72D32300GU-5-B, or Samsung M381L3223ETM-CCC.
- One RLDRAM-II SIO ×18 device connected to the Stratix II I/O bank 3. The boards will use the Micron MT49H16M18CFM-2.5 third-party memory device.
- Two RLDRAM-II CIO ×18 devices connected to the Stratix II I/O bank 4 that support 400 MHz double data rate (DDR). The boards will use one of the following third-party memory devices: Micron MT49H16M18FM-2.5 or Infineon HYB18RL28818AC-2.5.

Figure 23 shows the Stratix II memory board I. The Stratix memory board I has the same components interfacing with a Stratix EP1S40F1020C5 device.

Figure 23. Stratix II Memory Board I



The Stratix-series memory board I is powered by a a single DC input with on-board regulators generating the other required lower voltages. In addition to the on-board regulators, fuse-isolated banana jacks will be provided for all unique voltages for characterization purposes. The incoming DC voltage will be regulated down to 3.3 V and 1.2 V using a dual switching power supply in order to efficiently support the fairly large DC drop from the input to the output. All other board voltages are generated from this 3.3-V rail. There are fuse sockets to isolate planes from regulators to allow bench supplies to power these sections using banana jacks. The following regulators are available on the Stratix memory board:

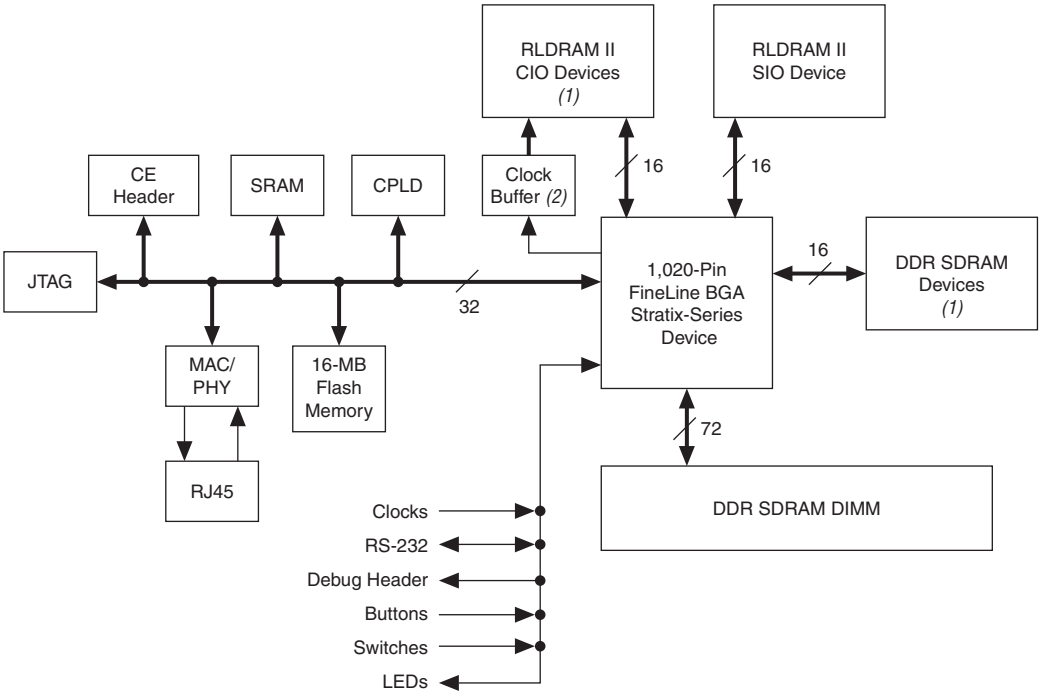
- Linear Technology LTC2901—Programmable supply monitor for monitoring board voltages
- National Semiconductor LP2996MR—DDR SDRAM and RLDRAM II termination regulator for generating the termination voltage (V_{TT}) and reference voltage (V_{REF})
- Linear Technology LTC1778EGN—Synchronous DC/DC controller to generate 3.3-V outputs.
- Micrel Semiconductor MIC29502BU—High-current low-dropout regulator for generating the power for the memory devices and the Stratix PLL.

The following regulators are available on the Stratix II memory board:

- Linear Technology LTC3728—Dual-output regulator for generating the Stratix II device's V_{CCINT} and 3.3-V outputs.
- Micrel Semiconductor MIC29502BU—High-current low-dropout regulator for generating the power for the memory devices and the Stratix PLL.
- National Semiconductor LP2996MR—DDR SDRAM and RLDRAM II termination regulator for generating the termination voltage (V_{TT}) and reference voltage (V_{REF}).
- Micrel Semiconductor MIC94300—Low-voltage low-dropout regulator for generating the Stratix II PLL power.
- Linear Technology LTC1872B—Step-up DC/DC controller for generating power for the fan circuit.

Figure 24 shows the Stratix-series memory board I block diagram.

Figure 24. Stratix Series Memory Board I Block Diagram



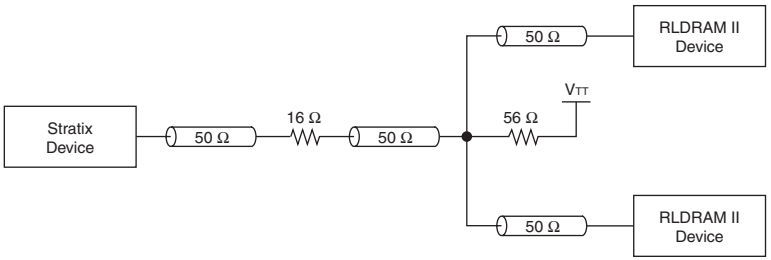
Notes to Figure 24:

(1) The Stratix series memory board I has multiple RLDRAM II CIO and DDR SDRAM devices.

(2) The clock buffer is only required for Stratix and Stratix GX devices.

Figure 25 shows the address termination scheme between Stratix-series and RLDRAM CIO devices for the shared address and control lines.

Figure 25. Address Termination Scheme between Stratix II & RLDRAM CIO Devices in Stratix-Series Memory Board I



Conclusion

RLDRAM II devices bridge the performance gap between DDR SDRAM and SRAM devices. With DRAM memory densities and SRAM-like low latency, RLDRAM II is ideal for communications, imaging, and server applications. The versatile memory interface in Stratix, Stratix GX and Stratix II FPGA devices enables designers to quickly and easily interface to RLDRAM II and take advantage of these enhanced features.

References

MT49H16M18, 288 CIO Reduced Latency RLDRAM II advance data sheet, Micron Technology, Inc.

MT49H16M18C, 288 SIO Reduced Latency RLDRAM II advance data sheet, Micron Technology, Inc.

Revision History

The information contained in version 3x.x of *AN 325: Interfacing RLDRAM II with Stratix II, Stratix & Stratix GX Devices* supersedes information published in previous versions.

The following changes were made to *AN 325: Interfacing RLDRAM II with Stratix II, Stratix & Stratix GX Devices* version **3.x**:

- Introduction - updated
- Interface Description
  - Block Diagram - updated
  - Interface Signals - updated text and table
  - Interface Architecture - updated
  - DLL-Based Data-Path Architecture - updated
  - PLL-Based Data-Path Architecture - added section and figure
  - Altera Memory Controller IP - updated and added figure

- Read Timing Margins for DLL-Based Implementation- added figure
- Memory Timing Parameters - updated
- FPGA Timing Parameters - updated, added table and figure
- Setup & Hold Margins Calculations - updated text and table

- Read Timing Margins for PLL-Based Implementation - added section
- Write Data Timing Margins - updated
  - FPGA Timing Parameters - updated
  - Setup & Hold Margins Calculations - updated table
  - Command and Address Timing - updated and added table
  - Bus Turnaround Timing Analysis/Read-to-Write - updated
  - Bus Turnaround Timing Analysis/Write-to-Read - updated

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
www.altera.com
Applications Hotline:
(800) 800-EPLD
Literature Services:
literature@altera.com