


Introduction

DDR3 SDRAM is the latest generation of DDR SDRAM technology, with improvements that include lower power consumption, higher data bandwidth, enhanced signal quality with multiple on-die termination (ODT) selection and output driver impedance control. DDR3 SDRAM brings higher memory performance to a broad range of applications, such as PCs, embedded processor systems, image processing, storage, communications, and networking.

Although DDR2 SDRAM is currently the more popular SDRAM, to save system power and increase system performance you should consider using DDR3 SDRAM. DDR3 SDRAM offers lower power by using 1.5 V for the supply and I/O voltage compared to the 1.8-V supply and I/O voltage used by DDR2 SDRAM. DDR3 SDRAM also has better maximum throughput compared to DDR2 SDRAM by increasing the data rate per pin and the number of banks (8 banks are standard).

 The Altera® ALTMEMPHY megafunction and DDR3 SDRAM high-performance controller only support local interfaces running at half the rate of the memory interface.

Altera Stratix® III and Stratix IV devices support DDR3 SDRAM interfaces with dedicated DQS, write-, and read-leveling circuitry.


Table 1 displays the maximum clock frequency for DDR3 SDRAM in Stratix III devices.

Table 1. DDR3 SDRAM Maximum Clock Frequency Supported in Stratix III Devices (Note 1), (2)

Speed Grade	f _{MAX} (MHz)
-2	533 (3)
-3 and I3	400
-4, 4L, and I4L at 1.1 V	333 (4), (5)
-4, 4L, and I4L at 0.9 V	Not supported

Notes to Table 1:

- (1) Numbers are preliminary until characterization is final. The supported operating frequencies are memory interface maximums for the device family. Your design's actual achievable performance is based on design and system specific factors and static timing analysis of the completed design.
- (2) Applies to modules and components using fly-by termination with leveling scheme.
- (3) Timing can close at the target speed in the Quartus II software version 8.0. In Quartus II software version 8.1, you can use these designs for prototyping, but you should not go to production until Altera releases final DDR3 models in Quartus II software version 9.0..
- (4) Performance is based on 1.1-V core voltage. At 1.1-V core voltage, the -4L speed grade devices have the same performance as the -4 speed grade devices.
- (5) The Quartus II software version 8.1 does not support DDR3 SDRAM below 360 MHz. The DLL mode is not supported below 360 MHz. The Quartus II software incorrectly shows that the design meets I/O timing.

 For more information on DDR3 SDRAM Maximum Clock Frequency Supported in Stratix IV Devices, refer to the [External Memory Interfaces](#) chapter in the *Stratix IV Device Handbook*.

This application note describes the FPGA design flow to implement external memory interfaces using Stratix III and Stratix IV devices, and provides design guidelines.

DDR3 SDRAMs are available as components and modules, such as DIMMs, SODIMMs, and RDIMMs. This application note describes implementing DDR3 SDRAM with Stratix III, Stratix IV, HardCopy® III, or HardCopy IV devices, including information on electrical and timing analysis, and the generation of a complete board-level system that you may use to demonstrate and validate the interface.

Stratix III and Stratix IV devices feature a similar input/output element (IOE) structure, so they effectively have the same external memory interface capabilities. HardCopy III and HardCopy IV devices may also be considered to have identical capabilities to their companion devices.



Throughout this document, statements made for Stratix III devices apply equally for Stratix IV, HardCopy III, and HardCopy IV devices, unless otherwise mentioned.

Background

This section gives background information on the following topics:

- [DDR3 SDRAM Overview](#)
- [IOE Dedicated DDR3 SDRAM Features](#)
- [DDR3 SDRAM Interface Termination and Topology](#)
- [ALTMEMPHY Megafunction Overview](#)

DDR3 SDRAM Overview

DDR3 SDRAM is internally configured as an eight-bank DRAM. DDR3 SDRAM uses an $8n$ prefetch architecture to achieve high-speed operation. The $8n$ prefetch architecture is combined with an interface that transfers two data words per clock cycle at the I/O pins. A single read or write operation for DDR3 SDRAM consists of a single $8n$ -bit wide, four-clock data transfer at the internal DRAM core and two corresponding n -bit wide, one-half clock cycle data transfers at the I/O pins. Read and write operations to the DDR3 SDRAM are burst oriented. Operation begins with the registration of an active command, which is then followed by a read or write command. The address bits registered coincident with the active command select the bank and row to be activated (BA0 to BA2 select the bank; A0 to A15 select the row). The address bits registered coincident with the read or write command select the starting column location for the burst operation, determine if the auto precharge command is to be issued (via A10), and select burst chop (BC) of 4 or burst length (BL) of 8 mode at runtime (via A12), if enabled in the mode register. Before normal operation, the DDR3 SDRAM must be powered up and initialized in a predefined manner.

Differential strobes DQS and $DQSn$ are mandated for DDR3 SDRAM and are associated with a group of data pins, DQ , for read and write operations. DQS , $DQSn$, and DQ ports are bidirectional. Address ports are shared for read and write operations.

Write and read operations are sent in bursts, DDR3 SDRAM supports BC of 4 and BL of 8. DDR3 SDRAM supports CAS latencies of 5 to 10.

DDR3 SDRAM devices use the SSTL-15 1.5-V I/O standard and can hold between 512 MB and 8 GB of data. The 1.5 V operational voltage reduces power consumption by 17% compared to DDR2 SDRAM.

All DDR3 SDRAM devices have eight internal banks. With more banks available, the page-to-hit ratio is twice that of DDR SDRAM. DDR3 SDRAM also allows bank interleaving, which represents a significant advantage for applications accessing random data. Bank interleaving can be extremely effective for concurrent operations and can hide the timing overhead that is otherwise required for opening and closing individual banks.

DDR3 SDRAM also supports calibrated parallel ODT via an external resistor RZQ signal termination options of $RZQ/2$, $RZQ/4$ or $RZQ/6 \Omega$ on all DQ , DM , and DQS and $DQSn$ signals.

DDR3 SDRAM typically supports controlled output driver impedance options of $RZQ/6$ or $RZQ/7$.

DDR3 SDRAM has a maximum frequency of 800 MHz or 1600 Mbps per DQ pin.

DDR3 SDRAM minimum operating frequency is 300 MHz.

Table 2 compares DDR, DDR2, and DDR3 SDRAM features.

Table 2. DDR and DDR2 SDRAM Features

Feature	DDR SDRAM	DDR2 SDRAM	DDR3 SDRAM	DDR3 SDRAM Advantage
Voltage	2.5 V	1.8 V	1.5 V	Reduces memory system power demand by 17%.
Density	64 MB to 1GB	256 MB to 4 GB	512 MB to 8 GB	High-density components simplify memory subsystem.
Internal banks	4	4 and 8	8	Page-to-hit ratio increased.
Prefetch	2	4	8	Lower memory core speed results in higher operating frequency and lower power operation.
Speed	100 to 200 MHz	200 to 533 MHz	300 to 800 MHz	Higher data rate.
Read latency	2, 2.5, 3 clocks	3, 4, 5 clocks	5, 6, 7, 8, 9, 10, and 11	Eliminating half clock setting allows 8n prefetch architecture.
Additive latency (1)	—	0, 1, 2, 3, 4	0, CL1, or CL2	Improves command efficiency.
Write latency	One clock	Read latency – 1	5, 6, 7, or 8	Improves command efficiency.
Termination	PCB, discrete to V_{TT}	Discrete to V_{TT} or ODT	Discrete to V_{TT} or ODT parallel termination. Controlled impedance output.	Improves signaling, eases PCB layout, reduces system cost.
Data strobes	Single-ended	Differential or single-ended	Differential mandated.	Improves timing margin.
Clock, address, and command (CAC) layout	Balanced tree	Balanced tree	Series or daisy chained	The DDR3 SDRAM read and write leveling feature allows for a much simplified PCB and DIMM layout. You can still optionally use the balanced tree topology.

Note to Table 2:

(1) The Altera DDR and DDR2 SDRAM high-performance controllers do not support additive latency.

IOE Dedicated DDR3 SDRAM Features

Stratix III devices have enhanced upon the IOE DDR capabilities of previous generations of devices by including the following functionality availability directly in the IOE.

- DDR registers
- Alignment and synchronization registers (including I/O clock divider)
- Half data-rate registers
- DQS phase-shift circuitry (up to four DLLs each with two-phase offsets)
- DQS postamble circuitry

- Differential DQS mode
- Read and write leveling circuitry
- Dynamic on-chip termination (OCT) control

To use these features you should use the Altera DDR3 SDRAM high-performance controller (a complete solution) or the Altera ALTMEMPHY megafunction (for a fully configured PHY that requires an additional custom or third-party memory controller).

Alternatively, you may access these IOE features directly via the following low-level megafunctions:

- ALTDQ_DQS megafunction—allows you to parameterize the following features:
 - DDR
 - alignment and synchronization
 - half data rate
 - DQS mode
- ALTDLL megafunction—allows you to parameterize the DQS phase-shift circuitry
- ALTOCT megafunction—allows you to parameterize the IOE OCT features
- ALTPLL megafunction—allows you to parameterize the device PLL
- ALTIOBUF megafunction—allows you to parameterize the device IO

Device Pin Utilization

Table 3 shows the DDR3 SDRAM interface pins and how to connect them to Stratix III pins.

Table 3. Stratix III DDR3 SDRAM Interface Pin Utilization (Part 1 of 2)

Pin	Pin Planner Symbol	Stratix III Pin
mem_dq	Q	DQ.
mem_dm	Q	DQ within the respective DQ group. Each DQ group has a common background color for all of the associated DQ and DM pins.
Differential mem_dqs or mem_dqsn	S or Sbar	DQS or DQSn.
mem_clk[0] or mem_clk_n[0]	—	Any unused DQ or DQS pins with DIFFIO_RX capability. (1)
mem_clk[n:1] or mem_clk_n[n:1]	—	Any unused DQ or DQS pins with DIFFOUT capability (where n is greater than or equal to 1). ALTMEMPHY-based solutions support only single rank designs: <ul style="list-style-type: none"> ■ Only the first rank is initialized (other ranks may have a different CAC mapping in DDR3 SDRAM). ■ Only the first rank is calibrated to, so exact timing requirements for other ranks may be different.

Table 3. Stratix III DDR3 SDRAM Interface Pin Utilization (Part 2 of 2)

Pin	Pin Planner Symbol	Stratix III Pin
Address and command	—	Any user I/O pin. To minimize skew, you should place address and command pins in the same bank or side of the device as the following pins: <ul style="list-style-type: none"> ■ mem_clk* pins ■ mem_dq, mem_dqs, mem_dm pins
Clock source	—	Dedicated PLL clock input pin with direct (not using a global clock net) connection to the PLL and optional DLL required by the interface.
Reset	—	Dedicated clock input pin (high fan-out signal).

Note to Table 3:

(1) ALTMEMPHY mimic path requirement only.

DQ and DQS Group Interface Width

For maximum performance and best skew across the interface, you should select a device where each required memory interface can completely reside within a single bank, or at least one side of the device.

Maximum interface width varies from device to device depending on the number of I/Os and DQS and DQ groups available. The smallest 480-pin device sizes can typically support a 128-MB 16-bit wide complete interface in both the top and bottom banks and a 32-bit wide complete interface in side banks. The largest 1760-pin devices can support a 72-bit wide DQ interface in each left and right banks. Achievable interface width depends on the number of address and command pins that the design requires. To ensure adequate PLL, clock and device routing resources are available, you should always test fit any IP in the Quartus II software before PCB sign-off.

Table 4 shows the number of DDR3 SDRAM suitable DQS and DQ groups available in Stratix III devices per side.

Table 4. Number of DQS and DQ Groups in Stratix III Devices per Side

Package	Side	×4	×8/×9
484-pin BGA	Top and bottom	5	2
	Left and right	12	4
780-pin BGA	Top and bottom	17	8
	Left and right	14	6
1152-pin BGA	Top and bottom	26	12
	Left and right	26	12
1517-pin BGA	Top and bottom	38	18
	Left and right	34	16
1760-pin BGA	Top and bottom	44	22
	Left and right	40	18

Note to Table 4:

- (1) Numbers are preliminary.
- (2) Some DQS or DQ pins are dual purpose and can also be required as RUP, RDN, or configuration pins. A DQS or DQ group is lost if you use these pins for configuration or as RUP or RDN pins for calibrated OCT. Ensure that the DQS and DQ groups are not also required for configuration or calibrated OCT.

 For more information, refer to the *External Memory Interfaces* chapter of the *Stratix III Device Handbook* and the *External Memory Interfaces* chapter of the *Stratix IV Device Handbook*.

DDR3 SDRAM Interface Pin Description

This section describes the DDR3 SDRAM interface pin description.

Clock Signals

DDR3 SDRAM devices use CK and CK# signals to clock the address and command signals into the memory. Furthermore, the memory uses these clock signals to generate the DQS signal during a read through the DLL inside the memory. The DDR3 SDRAM data sheet specifies the following timings:

- t_{DQSCK} is the skew between the CK or CK# signals and the DDR3 SDRAM-generated DQS signal
- t_{DQH} is the DQS falling edge from CK rising edge hold time
- t_{DQS} is the DQS falling edge from CK rising edge setup time
- t_{DQSS} is the positive DQS latching edge to CK rising edge

DDR3 SDRAM can use a daisy-chained CAC topology, so the memory clock arrives at each chip at a different time. To compensate for this flight time skew between devices across a typical DIMM, write leveling must be employed.

Strobes, Data, DM, and Optional ECC Signals

The DQS is bidirectional. Differential DQS strobe operation enables improved system timing due to reduced crosstalk and less simultaneous switching noise on the strobe output drivers. The DQ pins are also bidirectional. Regardless of interface width, DDR3 SDRAM interfaces can operate in either $\times 4$ or $\times 8$ mode DQS groups, which is dependent on your chosen memory device or DIMM. The $\times 4$ and $\times 8$ configurations use one pair of bidirectional data strobe signals, DQS and DQS#, to capture input data. However, two pairs of data strobes, UDQS and UDQS# (upper byte), and LDQS and LDQS# (lower byte), are required by the $\times 16$ configuration devices. A group of DQ pins must remain associated with its respective DQS and DQS# pins.

The DQ signals are edge-aligned with the DQS signal during a read from the memory and are center-aligned with the DQS signal during a write to the memory. The PHY shifts the DQ signals by -90° during a write operation to center align the DQ and DQS signals. The memory controller delays the DQS signal during a read, so that the DQ and DQS signals are center aligned at the capture register. Stratix III devices use a phase-locked loop (PLL) to center-align the DQS signal with respect to the DQ signals during writes and use DLL-controlled DQS phase-shift circuitry to shift the incoming DQS signal during reads.

Address and Command Signals

Address and command signals in DDR3 SDRAM devices are clocked into the memory device using the CK or $CK\#$ signal. These pins operate at single data rate (SDR) using only one clock edge. The number of address pins depends on the DDR3 SDRAM device capacity. The address pins are multiplexed, so two clock cycles are required to send the row, column, and bank address. The CS , RAS , CAS , WE , CKE , and ODT pins are DDR3 SDRAM command and control pins.

The DDR3 SDRAM address and command inputs do not have a symmetrical setup and hold time requirement with respect to the DDR3 SDRAM clocks, CK , and $CK\#$.

For SDRAM high-performance controllers in Stratix III devices, the address and command clock is always one of the PLL dedicated clock outputs whose phase can be adjusted to meet the setup and hold requirements of the memory clock. The address and command clock is also typically half-rate, although a full-rate implementation can also be created. The command and address pins use the DDIO output circuitry to launch commands from either the rising or falling edges of the clock. The chip select (CS_N) clock enable (CKE) and ODT pins are only enabled for one memory clock cycle and can be launched from either the rising or falling edge of the address and command clock signal. The address and other command pins are enabled for two memory clock cycles and can also be launched from either the rising or falling edge of the address and command clock signal.



In ALTMEMPHY-based solutions, the address and command clock ac_clk_1x is always half rate. However, because of the output enable assertion, CS_N , CKE and ODT behave like full-rate signals even in a half-rate PHY.

PLL and DLL Features and Availability

Stratix III devices are well equipped to address the clocking requirements of external DDR3 SDRAM interfaces. Stratix III PLLs have an increased number of outputs and global clock routing resources when compared to earlier device generations. Stratix III top and bottom PLLs feature 10 output (C) counters, also left and right PLLs feature 7 output (C) counters. This increased number of PLL outputs allows for the use of dedicated clock phases.

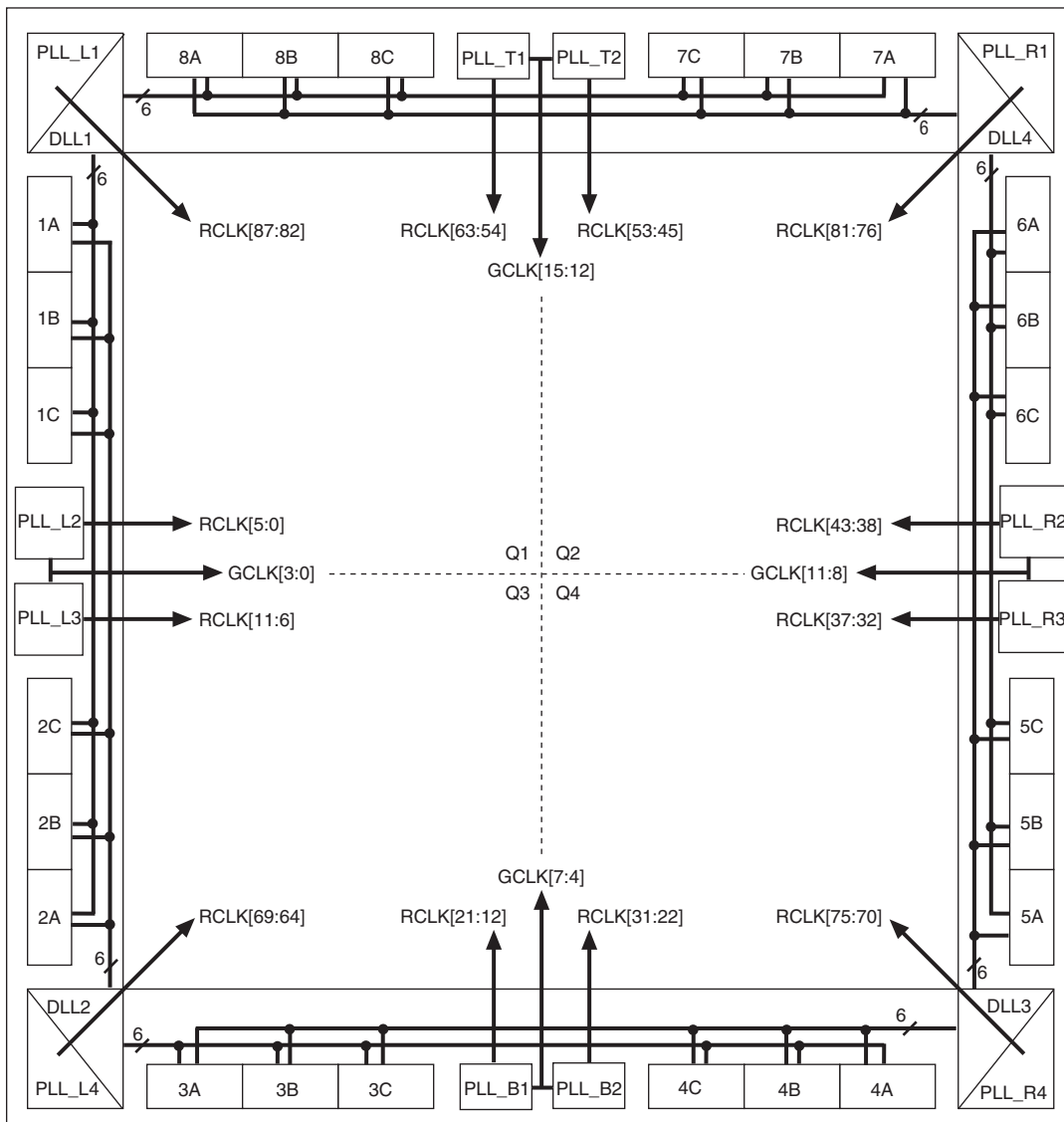
In general, each Stratix III PLL has access to 4 global clocks (GCLK) and 6 regional clocks (RCLK) (left and right) or 10 RCLK (top and bottom).

Stratix III devices also feature four DLLs (one located in each corner of the device). The FPGA can support a maximum of four unique frequencies, with each DLL running at one frequency. Each DLL can also support two different phase offsets, which allow a single Stratix III device to support eight different DLL phase shift settings. Additionally, each DLL can access the two sides adjacent to its location. Thus each I/O bank is accessible by two different DLLs, giving more flexibility when creating multiple frequency and phase-shift memory interfaces.



For more information, refer to the *Clock Networks and PLLs in Stratix III Devices* chapter and the *External Memory Interfaces* chapter in the *Stratix III Device Handbook*.

Figure 1 shows PLL and DLL locations in Stratix III devices with global and regional clock resources.

Figure 1. PLL and DLL Locations and Resources in Stratix III Devices

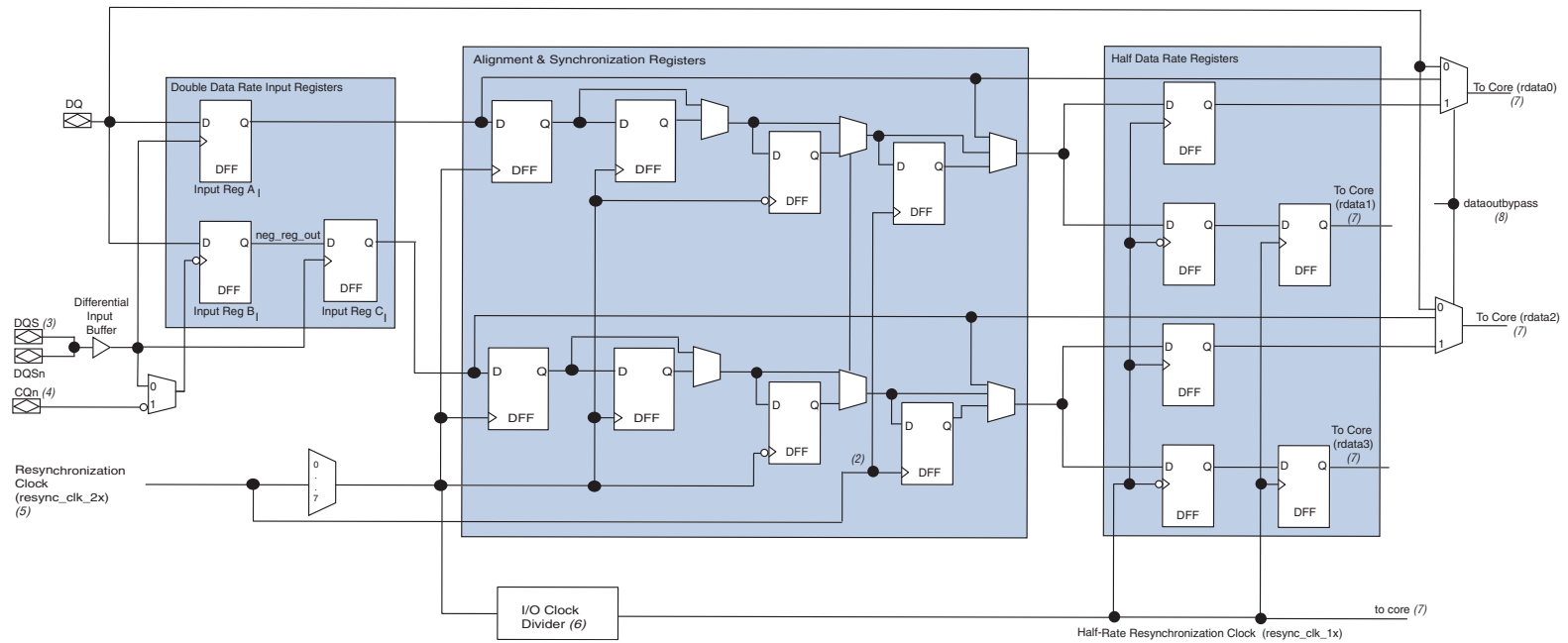
IOE Registers

Stratix III IOE registers include the following feature enhancements, over the previous generation of devices, which greatly simplify high speed memory interface design:

- Single-ended or differential DQS signaling
- Alignment and synchronization registers
- Half data rate registers
- I/O clock divider
- Programmable delay
- Read and write leveling—one per subbank. For example, bank 1a, 1b, and 1c = three circuits

Figure 2 and Figure 3 show the Stratix III IOE structure.

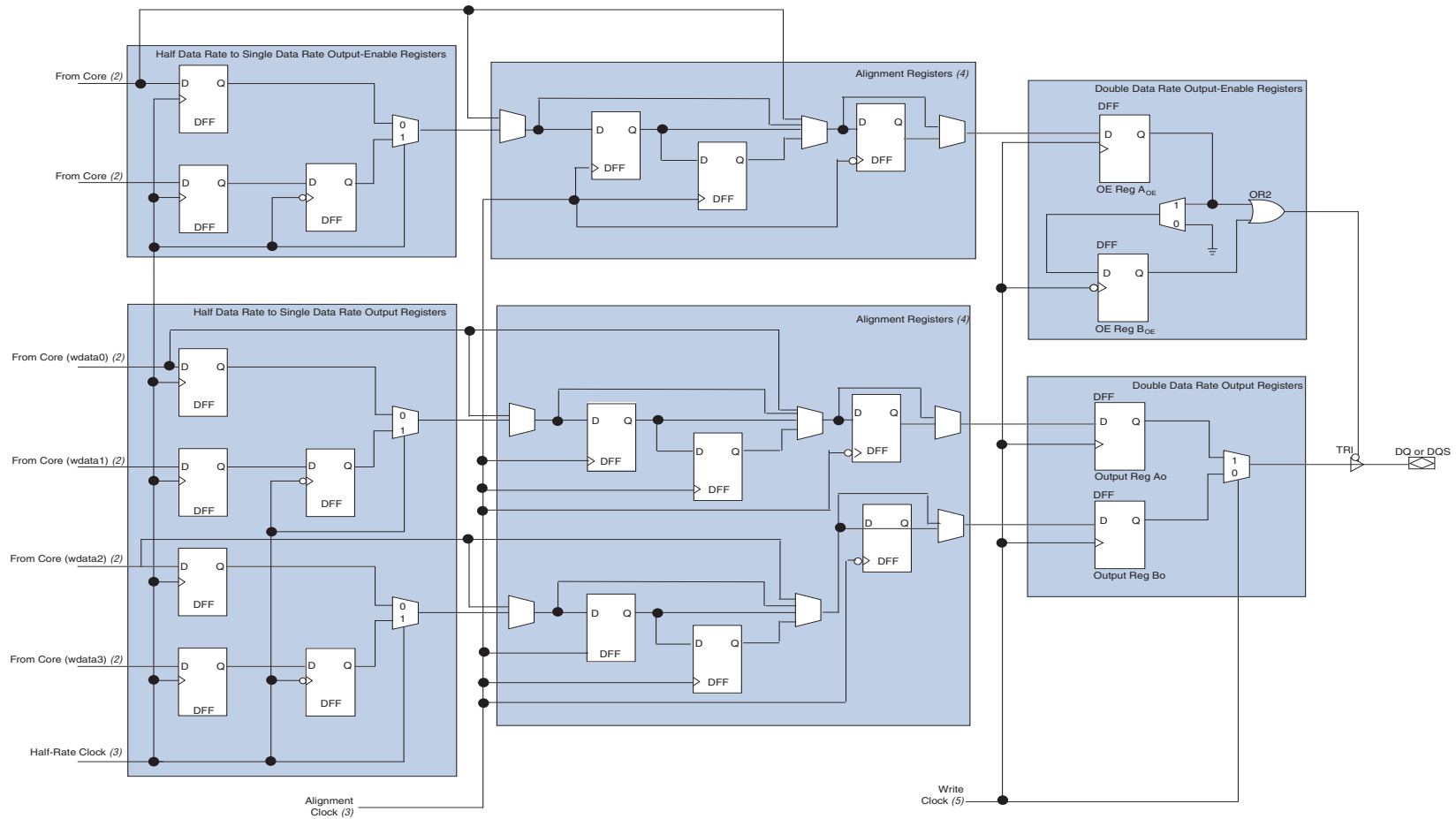
Figure 2. Stratix III IOE Input Registers



Notes:

- (1) You can bypass each register except the first in this path.
- (2) The 0-phase resynchronization clock from the read-leveling delay chain.
- (3) The input clock can be from the DQS logic block (whether the postamble circuitry is bypassed or not) or from a global clock line.
- (4) This input clock comes from the CQn logic block.
- (5) This resynchronization clock can come either from the PLL or from the read-leveling delay chain.
- (6) The I/O clock divider resides adjacent to the DQS logic block. In addition to the PLL and read-levelled resynchronization clock, the I/O clock divider can also be fed by the DQS bus or CQn bus.
- (7) The half-rate data and clock signals feed into a dual-port RAM in the FPGA core.
- (8) You can dynamically change the `dataoutbypass` signal after configuration.

Figure 3. Stratix III IOE Output Registers



Notes:

- (1) You can bypass each register block of the output and output-enable paths.
- (2) Data coming from the FPGA core are at half the frequency of the memory interface.
- (3) Half-rate and alignment clocks come from the PLL.
- (4) These registers are only used in DDR3 SDRAM interfaces.
- (5) The write clock can come from either the PLL or the write leveling delay chain. There is a 90° offset between the DQ write clock and DQS write clock.

Single-ended or Differential DQS Signaling

Stratix III devices directly support differential DQS mode and the single-ended standard supported in previous device families. DDR3 SDRAM mandates differential DQS signaling.

Differential DQS strobe operation enables improved system timing due to reduced crosstalk and simultaneous switching noise on the strobe output drivers.

DDR Registers

Similar to the previous generation of devices, DDR registers are provided on all sides of the device so that DDR I/O structures can be directly implemented in the IOE, thus saving core logic and ensuring tight skew is easily maintained, which eases timing. Stratix III devices now feature four DLLs, so DQS capture mode is now supported on every side of the device.

Alignment and Synchronization Registers

In previous device families the resynchronization registers had to be located in the core of the device, which made the placement of these registers with respect to the DDR IOE critical to ensure that timing is achieved. Stratix III devices have been enhanced to include the alignment and synchronization registers directly within the IOE, hence timing is now significantly improved and you are no longer concerned with ensuring critical register placement with respect to the DDR IOE. Typically, the resynchronization register is clocked via a dedicated output from the PLL. However, it may also be clocked directly from the read-leveling delay chain. The output alignment registers are typically clocked from the PLL.



Generally alignment and synchronization registers are optional and can be bypassed if not required; for ALTMEMPHY-based designs, these registers are required.

Regardless of interface speed, ALTMEMPHY always implements synchronization registers. Hence latency through the PHY may not be optimal for lower frequency designs.

Stratix III devices include only one leveling delay chain per I/O subbank. For example, subbank 1A includes a single leveling chain, 1B includes a second leveling chain, and so on.

If the half-rate resynchronization clock is sourced from the leveling delay chain, it may be cascaded from bank to bank, say 1A to 1B. In this configuration memory controllers must form a single contiguous block of DQS groups that are not staggered or interleaved with another memory controller. Additionally two PHYs cannot share the same sub-bank as only one leveling delay chain exists per sub-bank.

Half Data Rate Registers

As external memory interface clock speeds increase, the core f_{MAX} can become the limiting factor in interface design. A common solution, which increases core f_{MAX} timing problems, is to implement a half data rate architecture. This solution has the effect of doubling the data width on the core side interfaces compared to a full-rate SDR solution, but also halves the required operating frequency. To simplify PHY design and provide easier design constraints, Stratix III devices include dedicated full-rate to half-rate registers within the IOE.

Clock Divider

To simplify and reduce the number of clocks required, a dedicated I/O clock divider is provided on a per DQS group basis, which can directly source the half-rate resynchronization clock from the full rate version.

To ease data alignment, a single I/O clock divider phase may be used for an entire interface, as the half rate resynchronization clock can be cascaded from DQ group to the adjacent DQ group. Hence, when using a common I/O clock divider, the high and low bit order may be aligned across the entire interface. Individual I/O clock dividers require the data alignment to be performed on a DQ group basis.

Balanced CAC topologies can use a single I/O clock divider, but interfaces cannot be interleaved.



ALTMEMPHY-based designs use multiple I/O clock dividers on a DQ group basis. ALTMEMPHY-based designs do not support balanced CAC topologies.

Programmable Delay

Stratix III I/O registers include programmable delay chains that you may use to deskew interfaces. Each pin can have different delay settings, hence read and write margins can be increased as uncertainties between signals can be minimized.



ALTMEMPHY-based designs do not use dynamic delay chains to deskew interfaces.

Read and Write Leveling

Stratix III I/O registers include read- and write-leveling circuitry to enable skew to be removed or applied to the interface on a DQS group basis. There is one leveling circuit located in each I/O subbank.



ALTMEMPHY-based designs for DDR3 SDRAM directly use leveling circuitry.

IOE OCT Features

Stratix III devices support dynamic calibrated OCT—previous Stratix devices did not. This feature allows the specified series termination to be enabled during writes, and parallel termination to be enabled during reads. In addition to series OCT, Stratix III devices also allow slew rate control to be applied with drive strength options. These I/O features allow you to greatly simplify PCB termination schemes.



For further information, refer to the *Stratix III Device I/O Features* chapter in the *Stratix III Device Handbook* and *AN 520: DDR3 SDRAM Interface Termination and Layout Guidelines*.

DDR3 SDRAM Interface Termination and Topology

This section discusses signal topology and termination of DDR3 SDRAM interfaces.



For more information, refer to memory vendor application notes and *AN 520: DDR3 SDRAM Interface Termination and Layout Guidelines*.

All DDR3 SDRAM interfaces use the following two classes of signal type:

- Unidirectional class I terminated signals, which include clocks, and address and command signals
- Bidirectional class II terminated signals, which include DQS, DQ, and DM signals

Unidirectional Class I Terminated Signals

All class I signals are multiload signals—they either go to a DIMM that has multiple memory devices, or they go to all memory devices that make up the interface. Altera recommends the ideal topology is a daisy-chained serial structure. Altera gives the following recommendations for the class I termination to V_{TT} :

- Do not use for interfaces using DIMMs, as it is implemented directly on the DIMM
- Place directly after the last device in the chain for discrete devices

Memory clocks are typically chosen to ensure an even and matched number of loads on each clock pair, so that the timing to each memory device is consistent assuming equal trace delays. Each clock pair should be loaded to ensure that significant slew rate distortion does not occur. Memory clocks are typically differentially terminated with an effective 100- Ω resistance. You can achieve 100- Ω differential termination in one of the following ways:

- 100- Ω single resistor directly between the positive and negative signal
- 50- Ω single-ended resistor to V_{TT} on each positive and negative pin
- 100- Ω up to V_{CC} and 100- Ω down to ground on each positive and negative pin

Electrically all these solutions look the same to differential AC signals



For information about the electrical I/O termination, refer to the *Stratix Device I/O Features* chapter of the *Stratix III Device Handbook*.

FPGA drive strength and series termination setting should maximize edge rate while ensuring that over or undershoot are not encountered.

The combined use of drive strength and slew rate, or output series termination options mean Stratix III is ideally configurable for any Class I termination schemes.



For further information, refer to *Micron Technical Note TN4720: Point-to-Point Package Sizes and Layout Basics*.

Bidirectional Class II Terminated Signals

Class II signals are typically point-to-point, unless you are using either:

- Multiple DIMMs
- Stacked or dual rank DIMMs or topologies

Stratix III devices include on-chip series and parallel termination. So generally, discrete termination at the FPGA end of the line is not required.

DDR3 SDRAM devices support dynamic parallel ODT at the memory end of the line. So typically, discrete termination is not required.

DDR3 SDRAM DIMMs include a series terminator and include output impedance control on the DQ, DQS, and DM pins. So if you are using DIMMs, a series terminator at the memory end of the line is never required.



ALTMEMPHY-based designs do not support multiple DIMMs or dual-rank stacked topologies because calibration only takes place on the first rank.

ALTMEMPHY Megafunction Overview

The Altera ALTMEMPHY megafunction allows the rapid creation of a physical layer interface (PHY) in Stratix III devices. The PHY safely transfers data between memory and user logic. The easy-to-use ALTMEMPHY megafunction GUI enables the rapid configuration of the highly configurable PHY. You can use the ALTMEMPHY megafunction with either a user-designed controller or the Altera DDR3 SDRAM high-performance controller. You can parameterize the ALTMEMPHY megafunction to support the following features:

- Full-rate or a half-rate operation (half rate only for DDR3)
- Single-ended or differential DQS mode (differential DQS mode for DDR3)
- Dynamic termination

The ALTMEMPHY megafunction automatically parameterizes and initializes your DDR3 SDRAM interface including read and write levelling function and control.

The ALTMEMPHY megafunction supports an initial calibration sequence to minimize the effect of process variations in the FPGA and memory device. During operation, the voltage and temperature (VT) tracking mechanism eliminates the effects on timing margin of VT variation. The calibration process centers the resynchronization clock phase into the middle of the data valid window, to maximize the setup and hold margin.

Additionally, the ALTMEMPHY megafunction automatically generates all required TimeQuest timing constraints.

All published Stratix III DDR3 SDRAM performance data assume the design uses the ALTMEMPHY megafunction.

Altera recommends the use of an ALTMEMPHY-based design whenever possible. However, in some situations a simpler ALTDQ_DQS solution may be preferred and potentially more optimal.



For more information, refer to the [External DDR Memory PHY Interface \(ALTMEMPHY\) Megafunction User Guide](#).



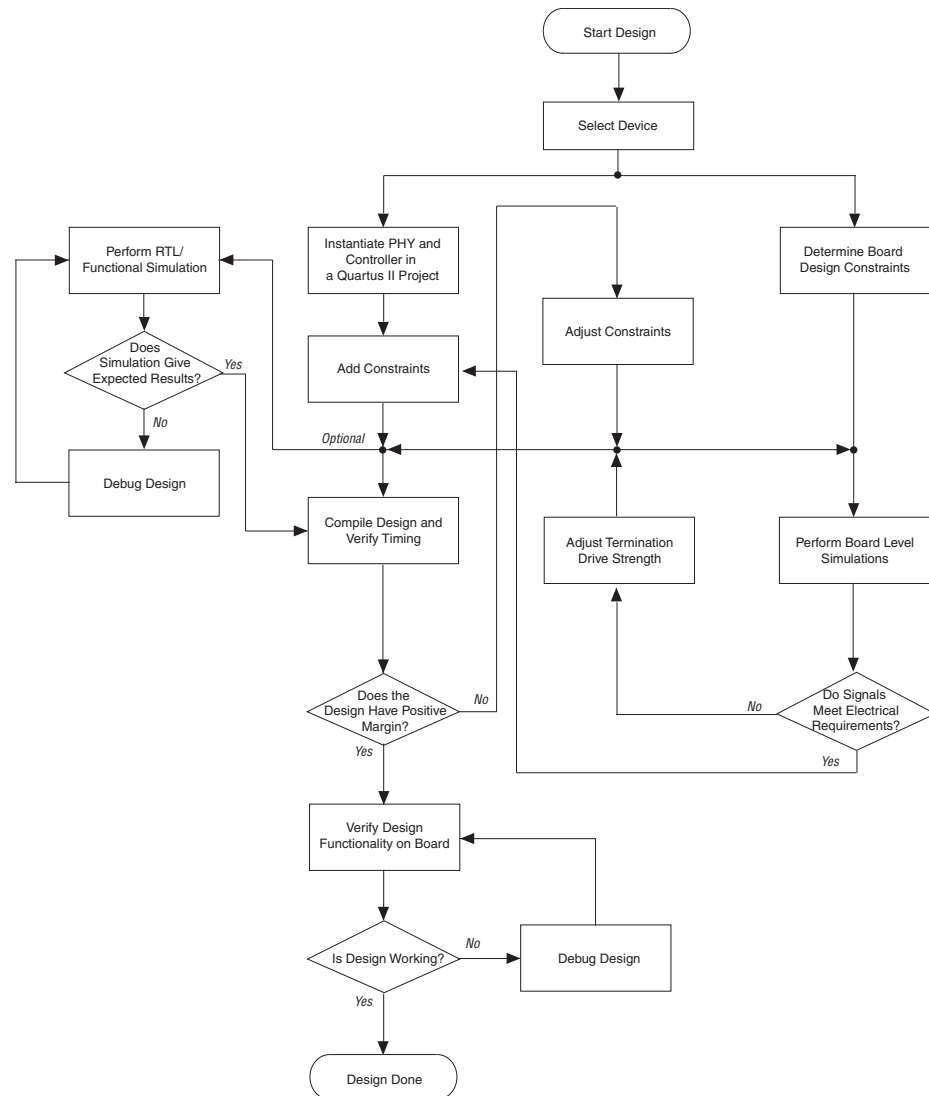
For more information, refer to the [ALTDLL and ALTDQ_DQS Megafunctions User Guide](#).

DDR3 SDRAM in Stratix III Devices Design Flow

Altera recommends the design guidelines described in this section as best practices for successful memory interface implementation in Stratix III devices. These guidelines provide the fastest out-of-the-box experience with external memory interfaces in Stratix III devices. Each step is discussed in detail in the following sections. This flow uses the DDR3 SDRAM high-performance controller.

Figure 4 shows the design flow required for Stratix III memory interfaces.

Figure 4. Design Flow for Implementing External Memory Interfaces in Stratix III Devices



Select a Device

This section discusses the following topics:

- “Bandwidth”
- “Full or Half Rate SDRAM Controller”
- “PLL and Clock Usage”

- “DLL Usage and Sharing”
- “Top, Bottom, Left, Right, and Hybrid Device Sides”
- “DQ and DQS Width Limits”
- “Address and Command, Clock, and Other Signals”

Memory controllers in Stratix III devices require access to dedicated IOE features, PLLs, and several clock networks. Stratix III devices are feature rich in all of these areas, so you must consider detailed resource and pin planning whenever implementing complex IP or multiple IP cores. This section provides an overview of what to consider in such instances.

 For more information, refer to the *Stratix III Device Handbook* and the relevant IP user guides.

Altera recommends that you create an example top-level design with the desired pin outs and all interface IP instantiated, which enables the Quartus II software to validate your design and resource allocation before PCB and schematic sign off.

As the structure of memory controllers varies considerably, this section uses the ALTMEMPHY architecture, where appropriate.

Bandwidth

Before designing any memory interface, determine the required bandwidth of the memory interface. Bandwidth can be expressed as shown in [Equation 1](#) and [Equation 2](#).


Equation 1.

$$\text{Bandwidth} = \text{data width (bits)} \times \text{data rate transfer (1/s)} \times \text{efficiency}$$

Equation 2.


$$\text{Data rate transfer (1/s)} = 2 \times \text{frequency of operation}$$

After calculating the bandwidth requirements of your system, determine which memory type and device to use. Altera has a memory selection white paper, which highlights the differences between the memory types.

 For information about selecting the different memory types, refer to the *Selecting the Right High-Speed Memory Technology for Your System* white paper.

DRAM typically has an efficiency of around 70%, but when using the Altera memory controller efficiency can vary from 10 to 92%.

In addition, Altera's FPGA devices support various data widths for different memory interfaces. The memory interface support between density and package combinations differs, so you must determine which FPGA device density and package combination best suits your application.


 For information about the FPGA density and package support for the different memory types, refer to the *External Memory Interfaces in Stratix III Devices* chapter of the *Stratix III Device Handbook*.

Full or Half Rate SDRAM Controller

When implementing memory controllers consider whether a half-rate or a full-rate datapath is optimal for your design. Full or half-rate mode have the following definitions:


- Full-rate mode presents data to the local interface at twice the width of the actual SDRAM interface at the full SDRAM clock rate
- Half-rate mode presents data to the local interface at four times the width of the actual SDRAM interface at half the SDRAM clock rate

Implementing memory controllers in half-rate mode results in the highest possible SDRAM clock frequency, while allowing the more complex core logic to operate at half this frequency. This implementation is most useful when core HDL designs are difficult to implement at the higher SDRAM clock frequency, but the required SDRAM bandwidth per I/O pin is still quite high.

 DDR3 SDRAM minimum operating frequency is 300 MHz. The ALTMEMPHY megafunction cannot achieve this frequency in full-rate implementations in Stratix III devices.

PLL and Clock Usage

The exact number of clocks and hence PLLs required in your design depends greatly on the memory interface frequency, and the IP used.

 Stratix III IOE includes dedicated circuitry for postamble protection, which is derived directly from the resynchronization clock.

In addition, some memory controller designs, like the ALTMEMPHY megafunction, use a VT tracking clock to measure and compensate for VT changes and their effects.

Consider the following points:

- PLLs in Stratix III devices connect to four maximum global clock nets
- Top or bottom PLLs in Stratix III devices connect to ten maximum regional clock nets
- Left or right PLLs in Stratix III devices connect to six maximum regional clock nets
- EP3S...80 and larger devices have two PLLs located in the middle of each side of the device
- EP3S...200 and larger device additionally have corner PLLs, which connect to six regional clock nets only
- Dual regional clock nets are created by using a regional clock net from each region. For example, a single dual regional clock net uses two regional clock nets
- If the design uses a dedicated PLL to only generate a DLL input reference clock, the PLL mode must be set to **No Compensation**, or the Quartus II software forces this setting automatically

- If the design cascades PLLs, the source (upstream) PLL should have a low-bandwidth setting, while the destination (downstream) PLL should have a high-bandwidth setting
- In Stratix III devices, two PLLs may be cascaded to each other through the clock network. In addition, where two PLLs exist adjacent to each other, there is a direct connection between them that does not require the global clock network. Using this path reduces clock jitter when cascaded PLLs. Cascaded PLLs are not recommended for ALTMEMPHY-based designs



You can only cascade PLLs between adjacent PLLs on the same side of the device.



If PLLs are cascaded in ALTMEMPHY based designs, you must use the adjacent PLL (direct connection) method.

- Input and output delays are only fully compensated for, when the dedicated clock input pins associated with that specific PLL are used as its clock source
- If the clock source for the PLL is not a dedicated clock pin for that specific PLL, jitter is increased, timing margin suffers, and the design may require an additional global or regional clock

The following additional ALTMEMPHY-specific points apply:

- ALTMEMPHY megafunctions require one global or regional clock, and five regional clock nets in Stratix III devices. Hence six clocks in total are required
- Any PLL on any side of a Stratix III device can support a single ALTMEMPHY interface. Ideally, you should pick a PLL and a PLL input clock pin that are located on the same side of the device as the memory interface pins
- As each PLL can only connect to four global clock nets, while the ALTMEMPHY megafunction requires six clock nets, an ALTMEMPHY-based design cannot cross from one side of a Stratix III device to the other side. For example, an ALTMEMPHY-based design can only exist within a dual regional side of a Stratix III device
- If a single ALTMEMPHY interface spans two side quadrants, a middle side PLL must be the source for that interface. The ten dual region clocks that the single interface requires block the design using the adjacent PLL (if available) for a second interface
- If a single ALTMEMPHY interface spans two top or bottom quadrants, a middle top or bottom PLL must be the source for that interface. The ten dual region clocks that the single interface require should not block the design using the adjacent PLL (if available) for a second interface



For more information on clock networks, refer to *Clock Networks and PLLs in Stratix III Devices* in the *Stratix III Device Handbook*.



For more information on multiple memory controllers, refer to *AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction*.

DLL Usage and Sharing

DDR3 SDRAM interfaces in Stratix III devices use DQS phase-shift circuitry for data capture. All Stratix III devices include a total of four DLLs: one located in each corner of the device. Each DLL can support two different phase offsets, and each DLL can access the two sides adjacent to its location. Hence, there are opportunities for DLL sharing or multiple different memory interface types on a single side of a Stratix III device.

DLL reference clocks must come from either dedicated clock input pins located on either side of the DLL or from specific PLL output clocks. Any clock running at the memory frequency is valid for the DLLs.



For more information on DLLs, refer to the *External Memory Interfaces* chapter in the *Stratix III Device Handbook*.

To minimize the number of clocks routed directly on the PCB, typically this reference clock is sourced from the memory controllers PLL. In general, DLLs can use the PLLs directly adjacent to them (corner PLLs when available) or the closest PLL located in the two sides adjacent to its location.

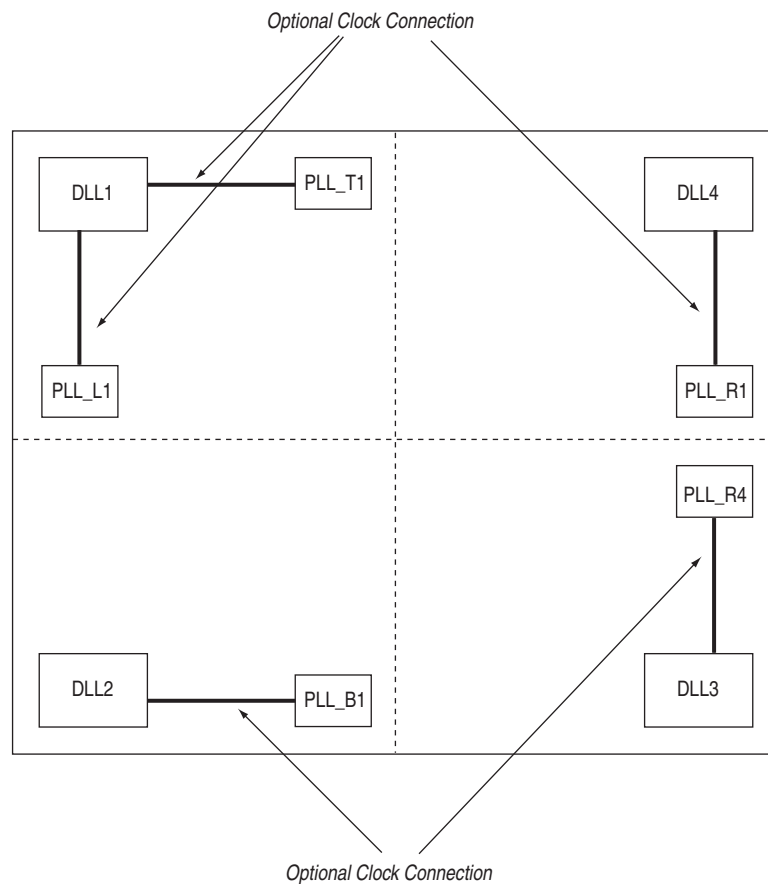


When designing for 780-pin packages with SE80, SE110 and SL150 devices, the PLL to DLL reference clock connection is limited.

DLL3 is isolated from a direct PLL connection and can only receive a reference clock externally from pins `clk [11 : 4] p`.

Figure 5 shows the 780-pin package devices PLL and DLL reference clock connections.

Figure 5. PLL and DLL Reference Clock Connections in 780-pin Package Devices



The DLL reference clock should be the same frequency as the memory interface, but the phase is not important.

The required DQS capture phase is optimally chosen based on operating frequency and external memory interface type (DDR, DDR2, DDR3, QDR II, or RLDRAM II). As each DLL supports two possible phase offsets, two different memory interface types operating at the same frequency can easily share a single DLL. More may be possible, depending on the phase shift required.



Altera memory IP always specifies a default optimal phase setting, to override this setting, refer to the respective IP user guide.

To simplify the interface to core IP connections, multiple memory interfaces operating at the same frequency usually share the same system and static clocks as each other where possible. This sharing minimizes the number of dedicated clock nets required and reduces the number of different clock domains found within the same design.

As each DLL can directly drive four banks, but each PLL only has complete C (output) counter coverage of two banks (using dual regional networks), situations can occur where a second PLL operating at the same frequency is required. As cascaded PLLs increase jitter and reduce timing margin, you are advised to first ascertain if an alternative second DLL and PLL combination is not available and more optimal.

Top, Bottom, Left, Right, and Hybrid Device Sides

This section discusses how to determine which device side to use (top and bottom, left and right, and hybrid).

Top or Bottom and Left or Right Interfaces

Ideally any interface should wholly reside in a single bank. However, interfaces that span multiple adjacent banks or the entire side of a device are also fully supported. Although vertical and horizontal timing parameters are not identical, timing closure can be achieved on all sides of the FPGA for the maximum interface frequency.

Hybrid Interfaces

The PLL regional clock net restriction and the fact that each DLL can drive its two adjacent sides suggests that an optimal PLL, DLL, and memory interface configuration resides in a single quadrant spanning two adjacent sides of the device. For maximum performance, Altera recommends that data groups for external memory interfaces should ideally reside within a single bank, but always within the same side of a device. High-speed memory interfaces in top or bottom versus left or right IOE have different timing characteristics and timing margins are affected.

However, Altera can support interfaces with hybrid data groups that wrap around a corner of the device between vertical and horizontal I/O at some speeds (see [Table 5](#))

Table 5. Hybrid Memory Interface Speeds (Half Rate) (*Note 1*), (*2*), (*3*), (*4*)

Memory Type	Speed Grade	f _{MAX} (MHz)
DDR3 SDRAM	-2	300
	-3 (<i>5</i>)	—
	-4 (<i>5</i>)	—

Notes to [Table 5](#):

- (1) Numbers are preliminary until characterization is final. The supported operating frequencies listed here are memory interface maximums for the FPGA device family. Your design's actual achievable performance is based on design and system specific factors, and static timing analysis of the completed design.
- (2) Applies for both DIMMS and components.
- (3) For the Quartus II version 8.0 hybrid functionality is not allowed and you see the following fitter error:
"Error: Cannot place DQ I/O "mem_dq[n]" to I/O location Pin_N since its memory interface I/O group cannot be placed"
- (4) The Quartus II version 8.0 SP1 includes native support for hybrid interfaces at these rates.
- (5) Hybrid DDR3 SDRAM interfaces are not supported in -3 and -4 speed grade devices, because of the 300-MHz minimum frequency requirement.

DQ and DQS Width Limits

Stratix III devices do not limit the width of DDR3 SDRAM interfaces beyond the following requirements:

- The entire interface DQ, clock, and address signals should reside within the same bank or side of the device

- Maximum possible interface width in any particular device is limited by the number of DQS groups available within that bank or side, see [Table 4](#)
- Sufficient regional clock networks are available to the interface PLL to allow implementation within the required number of quadrants
- Sufficient spare pins exist within the chosen bank or side of the device to include all other address and command, and clock pin placement requirements
- The greater the number of banks, the greater the skew, hence Altera recommends that you always generate a test project of your desired configuration and confirm that it meets timing

Address and Command, Clock, and Other Signals

This section describes the following signals:

- Address and command
- Clock
- Other signals

DDR3 SDRAM Component Additional Pins

The largest individual DDR3 SDRAM components typically available are 2GB ×8 devices. These devices usually require a maximum of 38 pins, which can be broken down in the following way:

- 8 DQ pins
- 1 DM pin
- 1 DQS pin
- 1 DQS# pin
- 15 A [14 : 0] pins
- 3 BA [2 : 0] pins
- 1 CK pin
- 1 CK# pin
- 7 CKE, CS#, RAS#, CAS#, WE#, ODT, reset# pins

DQ, DM, DQS, and DQSn should reside in a dedicated ×8 DQS group, the remaining 27 additional signals should be placed within the same bank.


DDR3 SDRAM DIMM Additional Pins

The largest DDR3 SDRAM DIMMs typically available are 4 GB ×72 dual rank modules. These modules usually require a maximum of 132 pins, which can be broken down in the following way:

- 72 DQ pins
- 9 DM [8 : 0] pins
- 9 DQS [8 : 0] pins
- 9 DQS# [8 : 0] pins

- 15 A [14 : 0] pins
- 3 BA [2 : 0] pins
- 2 CK [1 : 0] pins
- 2 CK# [1 : 0] pins
- 11 CKE [1 : 0], CS# [1 : 0], RAS#, CAS#, WE#, ODT [1 : 0], RESET#, EVENT# pins

DQ, DM, DQS, and DQSn should reside in 9 × 8 DQS groups ensuring that DQ group pin order is maintained. The remaining 33 additional signals should be placed within the same bank or side of the device.

 ALTMEMPHY-based interfaces do not directly support dual rank implementations.

R_{UP} and R_{DN} Calibration Blocks

If calibrated series, parallel, or dynamic termination is used for the I/O in your design, it requires a calibration block. This block requires a pair of R_{UP} and R_{DN} pins located within the same V_{CCIO} voltage bank.

This calibration block is not required to be within the same bank or side of the device as the IOEs it is serving. However, RUP and RDN pins are typically shared with DQ and DQS pins in Stratix III devices.

DQS and DQSn pins in some of the × 4 groups can also be used as RUP and RDN pins. You cannot use a × 4 group for memory interfaces if you are using its pin members as RUP and RDN pins for OCT calibration. You may use the × 8/× 9 group that includes this × 4 group, if you are not using DM pins with your differential DQS pins.


If you fail to correctly instantiate the required number of calibration blocks for your design, the Quartus II software automatically adds the calibration blocks during compilation. With multiple calibration blocks, the Quartus II software does not know which calibration blocks are associated with which blocks of logic.

If the design only requires a single calibration block, its pins are in the following format:

```
termination_blk0~_rup_pad
termination_blk0~_rdn_pad
```

A × 8/× 9 group is comprised of 12 pins, as the groups are formed by stitching two groups of × 4 mode with 6 total pins each. A typical × 8 DDR3 SDRAM device consists of one DQS, one DQS#, one DM, and 8 DQ pins, which totals 11 pins. The two additional RUP and RDN pins cannot fit in a single × 8/× 9 group, if you are using DM. If you are not using the DM pin, only 10 of the possible 12 pins are required. So if you choose your pin assignment carefully, you can use the 2 extra pins for RUP and RDN.

If you are using DM, pick different pin locations for RUP and RDN pins—for example, in the bank that contains address and command pins.

 You need to pick your DQS and DQ pins manually for the × 8, × 16 and × 18, or × 32 and × 36 groups, if they have pins they are using for RUP and RDN. The Quartus II software may not place these pins correctly and may give you a no-fit.

- For more information on calibration blocks, refer to the *Stratix III Device I/O Features* chapter in the *Stratix III Device Handbook*, the *ALTOCT Megafunction User Guide*, and *AN 465: Implementing OCT Calibration in Stratix III Devices*.

Instantiate PHY and Controller in a Quartus II Project

After selecting the appropriate device and memory type, create a project in the Quartus® II software that targets the device and memory type.

When instantiating the datapath for DDR3 SDRAM interfaces in Stratix III devices, Altera recommends that you use the ALTMEMPHY megafunction for the datapath and PHY. The ALTMEMPHY megafunction features a license-free PHY that you may use with the Altera SDRAM high-performance controllers or your own custom controller.

The Altera high-performance controllers automatically include the ALTMEMPHY megafunction. Even if you plan to use your own controller, Altera recommends that you first create a design using a SDRAM high-performance controller and then replace the Altera controller with your own controller. This method gives you an example design, which you can simulate and verify on your own PCB.

- For more information about instantiating the PHY, refer to the *External DDR Memory PHY Interface (ALTMEMPHY) Megafunction User Guide*.

Perform Board-Level Simulations and Line Simulation

This design flow indicates that you determine board design constraints and perform board-level simulations at the end of the flow. However, Altera recommends prelayout SI simulations (line simulations) should take place before board layout and that you use these parameters and rules during the initial design development cycle. Advanced I/O timing and board trace models now directly impact device timing closure.


Add Constraints

The next step in the design flow is to add the timing, location, and physical constraints related to the external memory interface. These constraints include timing, pin locations, I/O standards, and pin loading assignments. The ALTMEMPHY megafunction only supports timing analysis using the TimeQuest Timing Analyzer with Synopsys Design Constraints (.sdc) assignments. These constraints are derived from the parameters you entered for the ALTMEMPHY megafunction or the SDRAM high-performance controller, based on the DDR3 SDRAM data sheet and tolerances from the board layout. The ALTMEMPHY megafunction uses TimeQuest timing constraints and the timing driven fitter to achieve timing closure.

After instantiating the ALTMEMPHY megafunction, the ALTMEMPHY MegaWizard® generates the following files that you need to properly constrain the design:

- `<variation_name>_phy_ddr_timing.sdc` to set timing constraints
- `<variation_name>_pin_assignments.tcl` to add I/O standard setting assignments
- `<variation_name>_phy_assign_dq_groups.tcl` to add the DQ group assignments to relate the DQ and DQS pin groups together for the Quartus II fitter to place them correctly

These script files are based on the design name used when instantiating the ALTMEMPHY megafunction. If you plan to use your own top-level design, you must edit the scripts to match your custom top-level design.

 For more information about creating, generating, and setting the constraints for the design, refer to *External DDR Memory PHY Interface (ALTMEMPHY) Megafunction User Guide*.

 To determine which drive strength and termination to use, refer to *AN 520: DDR3 SDRAM Interface Termination and Layout Guidelines*.

Plan Resources

This section describes planning resources.

Table 6 shows the pin placements that Altera recommends.

Table 6. Stratix III DDR3 SDRAM Pin Placement Recommendations

Signal	Pin on FPGA	Pin on Memory Device
Data (mem_dq)	DQ	DQ
Data mask (mem_dm)	DQ (1)	DM
Data strobe (mem_dqs)	DQS or DQSn	DQS or DQS#
Memory clock (mem_clk)	DQ, or DQS, or DQSn (2), (3)	CK or CK#
Address	Any user I/O (4)	A or BA
Command	Any user I/O (4)	CS#, RAS#, CAS#, WE#, CKE, or ODT

Notes to Table 6:

- (1) The DM pins must be in the DQ group.
- (2) Any unused DQ or DQS pins with DIFFIO_RX capability for mem_clk[0] and mem_clk_n[0].
- (3) Any unused DQ or DQS pins with DIFFOUT capability for mem_clk[n:1] and mem_clk_n[n:1]. Where *n* is greater than or equal to 1.
- (4) Ensure that address and command pins are placed on the same side of the device as the memory clock pins. Also if OCT is used, ensure that the RUP and RDN pins are assigned correctly.

The SDRAM high-performance controllers do not generate pin assignments for non-memory signals such as clock sources or pin location assignments for the design. Launch **Pin Planner** to make these assignments to the design.

Advanced IO Timing

As part of I/O planning, especially with high-speed designs, you should take board-level signal integrity and timing into account. When adding an FPGA device with high-speed interfaces to a board design, the quality of the signal at the far end of the board route, and the propagation delay in getting there, are vital for proper system operation.


The advanced I/O timing option is turned on by default for Stratix III devices.

Ensure that the overall board trace models are a reasonable approximation for each I/O standard on each PCB. For high-speed complex interfaces like DDR3 SDRAM, ensure that the board trace models are accurate for each specific signal class by using Pin Planner. Pin Planner includes a GUI schematic representation of the board trace model that you are modifying.

Board trace models include two transmission line segments (near and far). These line segments are ideal for SDRAM interfaces. You can use the near transmission line to represent the PCB and the far transmission line to represent the DIMM.

The board trace model should only include PCB or off chip information. Do not include the Stratix III I/O pin and package capacitance, OCT, or drive strength settings, as the Quartus II software ascertains these dynamically.


ODT at the memory should be included as external discreet termination and the capacitive loading of the memory should be calculated for each net and also added.

 Ideally, the distributed capacitance and inductance of your PCB traces should be ascertained from your PCB development tool. However, in general a 50-Ω trace is approximately 3 pF and 8 nH per inch.

Trace delay information can be entered on a per net basis if desired, but in general a net group basis should be sufficient. Multiple nets can be selected at the same time and then have their respective board trace models all entered simultaneously.

Altera suggests the following net groups:

- mem_clk
- mem_addr (mem_a and mem_ba)
- mem_ctrl (mem_cas#, mem_cke, mem_cs_n, mem_odt, mem_ras_n, mem_we_n)
- mem_dq_group0 (mem_dq[7..0], mem_dm[0])
- mem_dq_group1 (mem_dq[15..8], mem_dm[1])
- mem_dq_group
- mem_dqs0 and mem_dqsn0

 The DQS pin can be combined with the respective DQ group as a single-ended signal, otherwise each differential DQS pin pair should be entered separately.

DIMM board trace models and SDRAM component capacitive loading information should be obtained from your memory vendor directly and must be included into your Quartus II board trace model parameters.

More precise board trace models result in more accurate TimeQuest timing analysis.

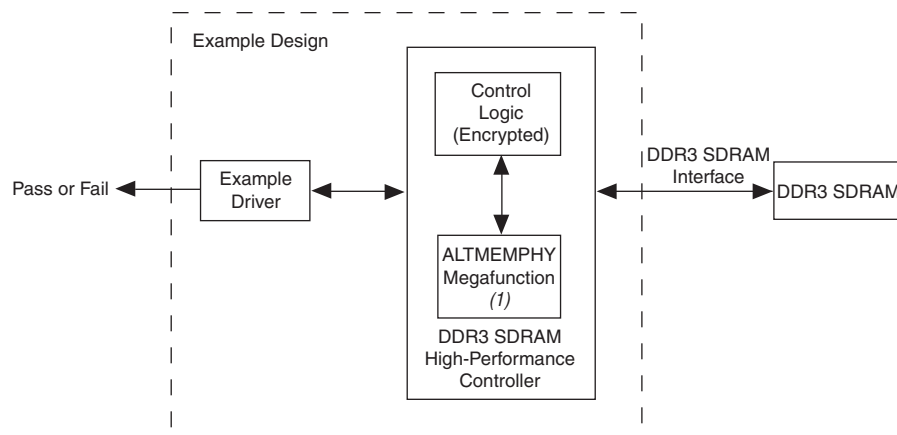
 For more information, refer to the *I/O Management* chapter of the *Quartus II Handbook*.

Perform RTL or Functional Simulation (Optional)

After instantiating the SDRAM high-performance controller, it generates an example design and driver for testing the memory interface.

Figure 6 shows a system-level diagram of the example design that the SDRAM high-performance controller creates for the design.

Figure 6. DDR3 SDRAM Controller System-Level Diagram



Note to Figure 6:

(1) The ALTMEMPHY megafunction automatically generates the PLL and the PLL is part of the ALTMEMPHY megafunction.

For more information about the different files generated by the DDR3 SDRAM high-performance controller, refer to the *DDR3 SDRAM High-Performance Controller User Guide*.

During the parameterization of the DDR3 SDRAM high-performance controller, there is an option to generate a simulation model of the ALTMEMPHY megafunction, an example design, and a testbench, so that functional simulation may be performed on the design.


Compile Design and Verify Timing

After constraining the design, compile the design in the Quartus II software. During the generation of the ALTMEMPHY megafunction or the SDRAM high-performance controller, the MegaWizard Plug-In Manager generates a verify timing script `<variation_name>_phy_report_timing.tcl`. After compiling the design in the Quartus II software, run the timing script to produce the timing report for different paths, such as write data, read data, address and command, and core (entire interface) timing paths in the design.

The verify timing script reports about margins on the following paths:

- Address and command setup and hold margin
- Half-rate address and command setup and hold margin
- Core setup and hold margin
- Core reset and removal setup and hold margin
- Write setup and hold margin
- Read capture setup and hold margin
- Read resynchronization setup and hold margin

- Write leveling t_{DQSS} setup and hold margin
- Write leveling t_{DSS} and t_{DSH} setup and hold margin

 For detailed information about timing analysis and reporting using the ALTMEMPHY megafunction, refer to *AN 438: Constraining and Analyzing Timing for External Memory Interfaces*.

Adjust Constraints

In the timing report of the design, you can see the worst case setup and hold margin for the different paths in the design. If the setup and hold margin are unbalanced, achieve a balanced setup and hold margin by adjusting the phase setting of the clocks that clock these paths.

For example, for the address and command margin, the address and command outputs are clocked by an address and command clock that can be different with respect to the system clock, which is 0° . The system clock clocks the clock outputs going to the memory. If the report timing script indicates that using the default phase setting for the address and command clock results in more hold time than setup time, adjust the address and command clock to be less negative than the default phase setting with respect to the system clock so that there is less hold margin. Similarly, adjust the address and command clock to be more negative than the default phase setting with respect to the system clock if there is more setup margin.

 For detailed information about the clocks that the ALTMEMPHY megafunction uses, refer to the *ALTMEMPHY Megafunction User Guide*.

Determine Board Design Constraints and Perform Board-Level Simulations

To determine the correct board constraints, run board-level simulations to see if the settings provide the optimal signal quality. With many variables that can affect the signal integrity of the memory interface, simulating the memory interface provides an initial indication of how well the memory interface performs. There are various electronic design automation (EDA) simulation tools available to perform board-level simulations. The simulations should be performed on the data, data strobe, control, command, and address signals. If the memory interface does not have good signal integrity, adjust the settings, such as drive strength setting, termination scheme or termination values to improve the signal integrity (realize that changing these settings affects the timing and it may be necessary to go back to the timing closure if these change).

 For detailed information about understanding the different effects on signal integrity design, refer to *AN 520: DDR3 SDRAM Interface Termination and Layout Guidelines*.

Trace information from your board-level simulation should be fed back into the Quartus II advanced I/O timing information.

Device-Side Termination

The Stratix III devices support both series and parallel OCT resistors to improve signal integrity. The Stratix III OCT eliminates the need for external termination resistors on the FPGA side, which simplifies board design and reduces overall board cost. You can dynamically switch between the series and parallel OCT resistor depending on whether the Stratix III devices are performing a write or a read operation. The OCT features offer user-mode calibration to compensate for any variation in VT during normal operation to ensure that the OCT values remain constant. The parallel and series OCT features on the Stratix III devices are available in either 25 or 50- Ω settings.

Memory-Side Termination

On the DDR3 SDRAM, there is a dynamic parallel ODT feature that you can turn on when the FPGA is writing to the DDR3 SDRAM and turn off when the FPGA is reading from the DDR3 SDRAM. To further improve signal integrity, DDR3 SDRAM supports calibrated output impedance drive control so that the driver can better match the transmission line.



For more information on available settings of the ODT, the output impedance drive control features, and the timing requirements for driving the ODT pin, refer to your DDR3 SDRAM datasheet.

Adjust Termination Drive Strength

Altera recommends the following termination scheme for single rank DDR3 SDRAM interfaces:

- FPGA side:
 - DQ and DQS: calibrated 50- Ω dynamic OCT
 - DM: calibrated 50- Ω series OCT
 - Command and address: maximum drive strength
 - Memory clock: uncalibrated 50- Ω series OCT
- DDR3 SDRAM side:
 - DQ, DQS, and DM: 60- Ω ODT and 34- Ω output impedance drive control
 - Command and address: 50- Ω external discrete parallel termination to V_{TT} (included on DIMM)
 - Memory clock: 100- Ω differential termination (included on DIMM)



Memory clocks use uncalibrated 50- Ω series OCT to ensure that the memory device does not observe glitches during power-up and initialization.

Although the recommendations are based on the simulations and experimental results, you must perform simulations, either using I/O buffer information specification (IBIS) or HSPICE models, to determine the quality of signal integrity on your designs.

Verify Design Functionality

Perform system level verification to correlate the system against your design targets using the Altera SignalTap® II logic analyzer.



For detailed information about using the SignalTap II, refer to the *Design Debugging Using the SignalTap II Embedded Logic Analyzer* chapter in volume 3 of the *Quartus II Software Handbook*.

Example Project Walkthrough

This walkthrough shows how to use the design flow (see “[DDR3 SDRAM in Stratix III Devices Design Flow](#)” on page 16) to design a 72-bit wide, 400-MHz, 800-Mbps DDR3 SDRAM interface. This example design also provides some recommended settings, including termination scheme and drive strength setting, to simplify the design.

The example design targets the Stratix III Memory Demonstration Kit, which includes a DIMM module (MT9JSF12872AY-1G1BZES). This flow applies to any other development kit or PCB.



The Stratix III Memory Demonstration Kit is not available for purchase.



Early versions of the Stratix III Memory Demonstration Kit included a MT16JTF25664AY-1G1D1 DDR3 SDRAM DIMM, which is a dual-rank DIMM and is not supported by ALTMEMPHY-based solutions. Ensure the MT9JSF12872AY-1G1BZES DDR3 SDRAM DIMM is fitted.

Software Requirements

This walkthrough assumes that you have experience with the Quartus II software. In addition, ensure you have the following software installed:

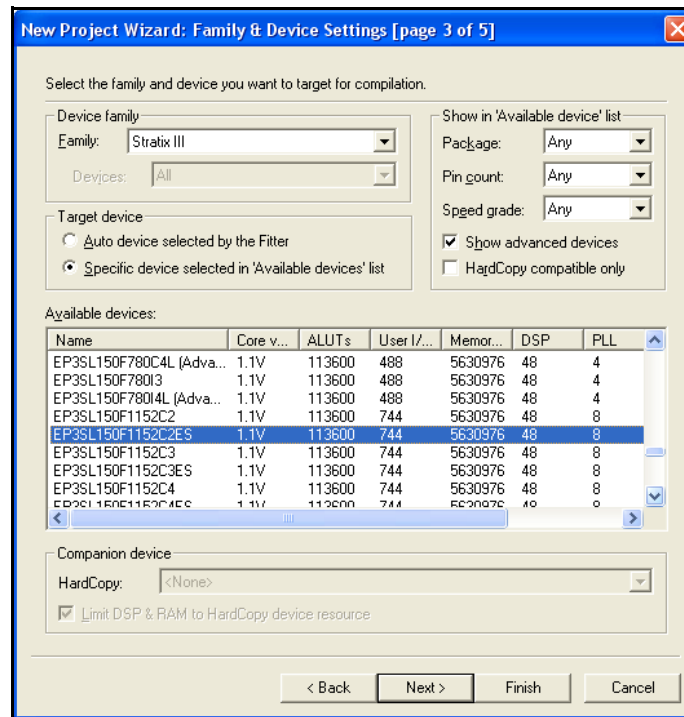
- Quartus II software v8.0 SP1
- DDR3 SDRAM high-performance controller v8.0 SP1

Select Device

This example design uses the EP3SL150F1152-C2ES device, which supports 72-bit wide DDR3 SDRAM at 400 MHz. The design uses a 72-bit wide 1-GB Micron MT9JSF12872AY-1G1BZES 533-MHz DDR3 SDRAM DIMM.

Create a Quartus II Project

To create a project in the Quartus II software that targets the EP3SL150F1152-C2ES device, see [Figure 7](#).

Figure 7. Create a Quartus II Project Targeting the EP3SL150F1152-C2ES Device

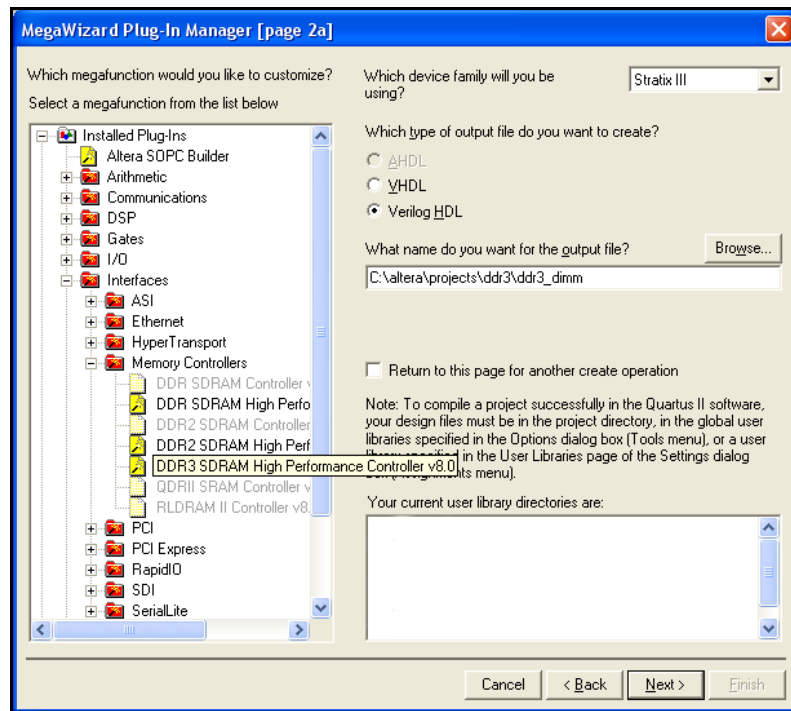
For detailed step-by-step instructions on how to create a Quartus II project, refer to the Tutorial in the Quartus II software, which is available by clicking the Help menu in the Quartus II window and selecting **Tutorial**.

Instantiate a PHY and a Controller

After creating a Quartus II project, instantiate the DDR3 SDRAM controller. This example design uses the DDR3 SDRAM high-performance controller, which instantiates the ALTMEMPHY megafunction automatically.

Before you open the MegaWizard Plug-In, you must add the **S3MB1_Derated (Micron MT9JSF12872AY-1G1BZES).xml** file to your `<installation directory>\80\ip\ddr3_high_perf\lib` directory. The .xml file is included in the application note .zip file.

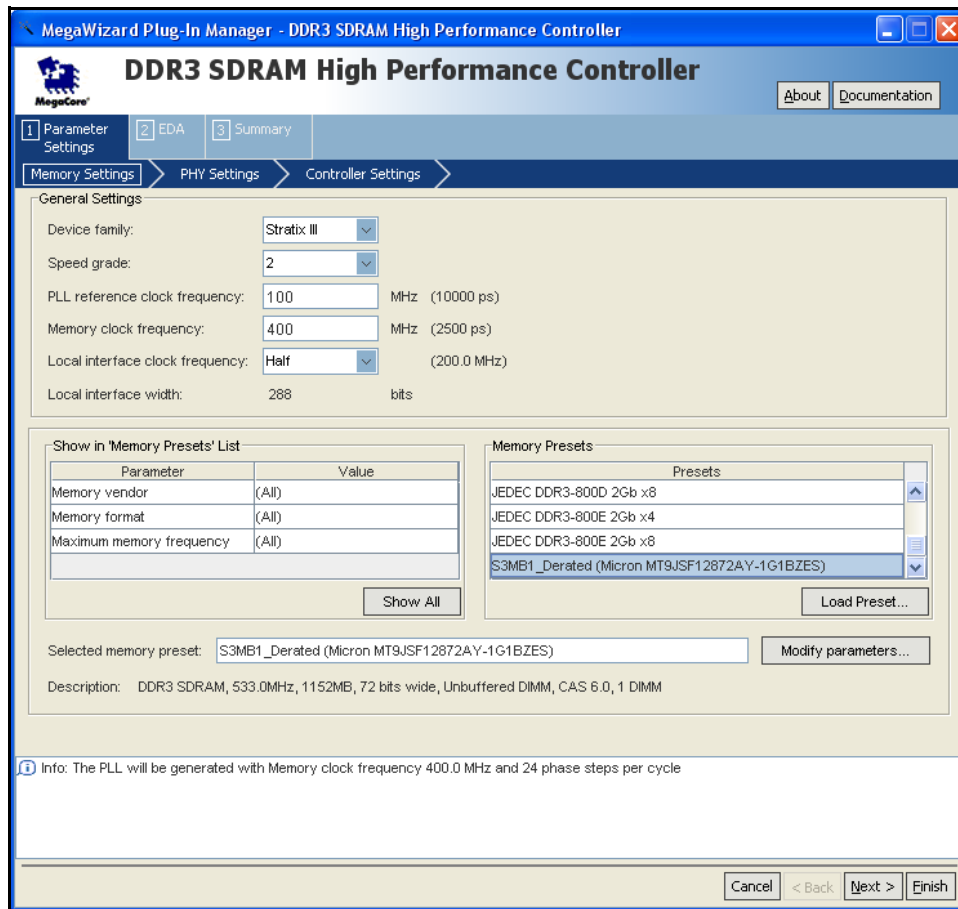
To select the **DDR3 SDRAM High Performance Controller** in the Interfaces section of the MegaWizard Plug-In Manager, see [Figure 8](#). For this example, enter `ddr3_dimm` for the name of the DDR3 SDRAM high-performance controller.

Figure 8. Select the DDR3 SDRAM High-Performance Controller

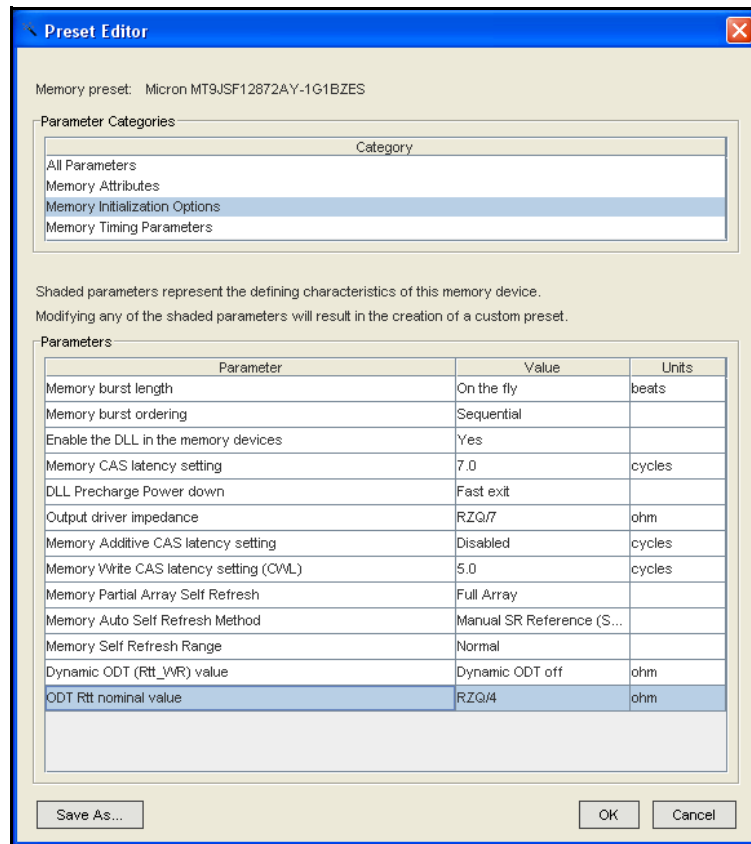
Parameterize the **DDR3 SDRAM High-Performance Controller** to interface with a 400-MHz, 72-bit wide DDR3 SDRAM interface.

1. In the **Memory Setting** tab, set **Speed grade** to **2**.
2. For **PLL reference clock frequency**, enter **100** (to match the on-board oscillator).
3. For **Memory clock frequency**, enter **400** (the maximum frequency supported for DDR3 SDRAM interfaces on Stratix III devices).
4. For the memory preset, select **S3MB1_Derated (Micron MT9JSF12872AY-1G1BZES)**, which gives a 72-bit wide 1,152-MB 533-MHz DDR3 unbuffered DIMM, see [Figure 9](#).

Figure 9. Parameterize the DDR3 SDRAM High-Performance Controller




5. To create a memory preset click **Modify parameters**. In the **Preset Editor** dialog box, you can modify the memory presets, see [Figure 10](#).

Figure 10. Modify the Memory Presets to Create a Custom Memory

t_{AC} and t_{QHS} are often not defined by memory vendors, as these values are only of use in static RTD calculations and non-DQS capture mode. The wizard does not require these parameters, so use the default values.

The t_{IS} , t_{IH} , t_{DS} , and t_{DH} parameters typically require slew rate derating.


-  For more information on slew rate derating and how to perform slew rate derating calculations, refer to the memory vendor datasheet.

Simulation and measurement show the following slew rate for the clock, address and command, and DQ and DQS pins on the Stratix III Memory Demonstration Board when using the default I/O standard and drive options:

- Address and command = 1.5 V/ns
- CLK and CLK# = 3 V/ns (differential)
- DQ = 2 V/ns
- DQS = 3 V/ns (differential)

Hence, the correct t_{IS} and t_{IH} values for this design are:

- $t_{IS} = 300 + 59 = 359$ ps
- $t_{IH} = 300 + 34 = 334$ ps
- $t_{DS} = 200 + 88 = 288$ ps
- $t_{DH} = 200 + 50 = 250$ ps

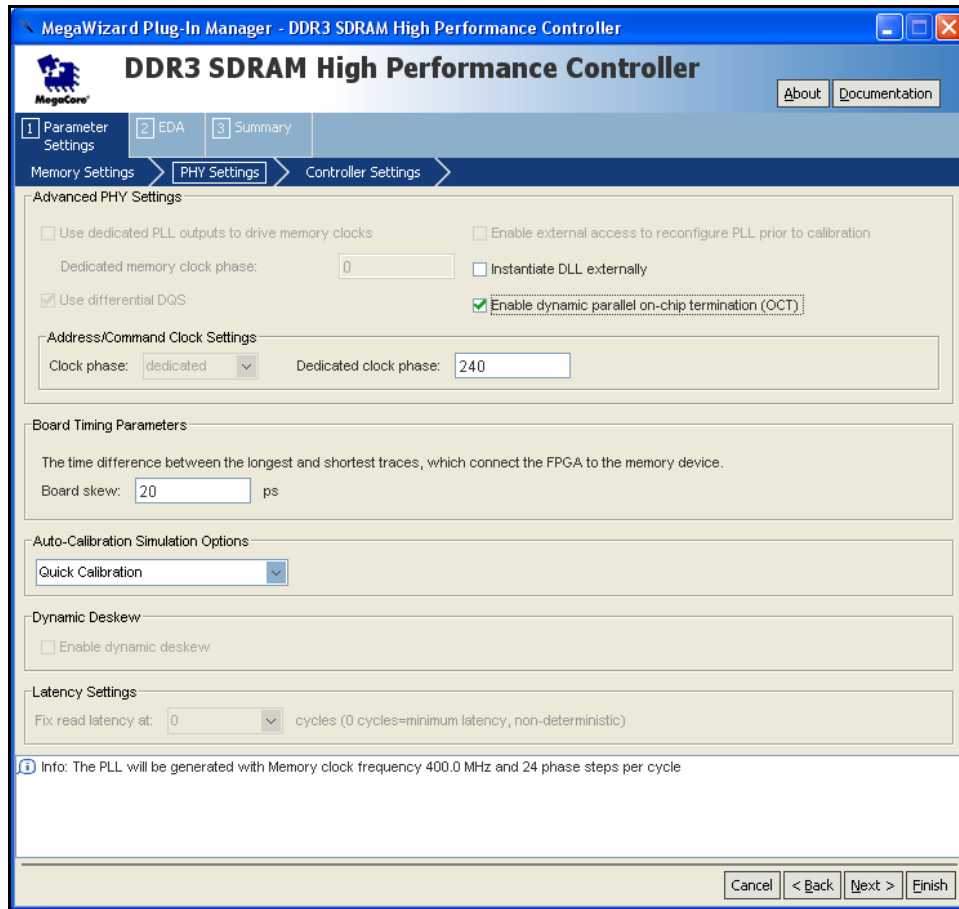
-  You should always simulate or measure your own design and topology to ensure accurate timing information and analysis.

The DDR3 SDRAM has a write requirement (t_{DQSS}) that states the positive edge of the DQS signal must be within $\pm 25\%$ ($\pm 90^\circ$) of the positive edge of the DDR3 SDRAM clock input. To achieve this skew requirement, ALTMEMPHY-based designs always use DDR IOE registers to generate the CK and CK# signals.

6. To set the ODT settings for the DDR3 SDRAM interface on your board, in the **Preset Editor**, select **Memory Initialization Options**. This example does not use the dynamic ODT (Rtt_WR) feature, which is useful for multiple-rank designs:
 - a. For **Output driver impedance** select **RZQ/7** (which is 34).
 - b. For **Dynamic ODT (Rtt_WR) value** select **Dynamic ODT off**.
 - c. For **ODT Rtt nominal value** select **RZQ/4** (which is 60).
7. In the **PHY Settings** panel, add the board skew parameter for the board in the **Board Timing Parameters** section. This timing parameter is the board trace variation between CK, CK#, CAC, DQ, DQS and DQS#. The default value is 20 ps. If your board can perform better or worse than this number, update it accordingly. The wizard uses this number to calculate the overall system timing margin. For this example design, enter the value of 20 ps as the board skew tolerance target is 20 ps.

Figure 11 shows generation and board timing parameters.


Figure 11. Set CK and CK# Generation and Board Timing Parameters



8. Turn on **Enable dynamic parallel OCT** (see Figure 11) for this example as the Stratix III memory demonstration board does not include discrete external termination on the DQ, DQS, DQS#, or DM pins, as the board was designed to use OCT.
9. Enter **240** in **Dedicated clock phase** for the **Address/Command Clock Settings**. Timing analysis shows that 240° is optimal for the Stratix III memory demonstration board.

The settings in **Auto-Calibration Simulation Options** are for RTL simulation only and are not applicable for gate-level simulation.

10. Click **Finish** to generate your MegaCore® function variation. The MegaWizard Plug-In Manager generates all the files necessary for your DDR3 SDRAM controller, and generates an example top-level design, which you may use to test or verify board operation.

 For detailed step-by-step instructions for parameterizing the DDR3 SDRAM high-performance controller, refer to the *DDR3 SDRAM High-Performance Controller User Guide*.


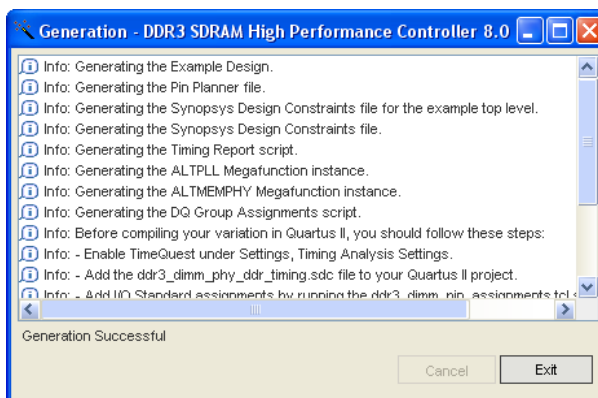
 The Altera generic “auto-generated” DDR3 memory model does not support all DDR3 features. The auto-generated generic DDR3 memory model works correctly in Quick Calibration mode as this model supports burst length of 4 and 8. This auto-generated model does not work in Full Calibration mode as the model does not support multipurpose register (MPR) readout required for DDR3 calibration. You must replace the Altera generic DDR3 memory model with a vendor’s component models for Full Calibration mode.

Figure 12 shows generation messages including tips on Quartus II settings.

Figure 12. Generation



Add Constraints

After instantiating the **DDR3 SDRAM High-Performance Controller**, the **ALTMEMPHY** megafunction generates the constraints files for the example design. Apply these constraints to the design before compilation.

Add Timing Constraints

When you instantiate an SDRAM high-performance controller, it generates a timing constraints file, *<variation_name>_phy_ddr_timing.sdc*. The timing constraint file constrains the clock and input and output delay on the SDRAM high-performance controller.

To add timing constraints, follow these steps:

1. On the Assignments menu click **Settings**.
2. In the **Category** list, expand **Timing Analysis Settings**, and select **TimeQuest Timing Analyzer**.
3. Select the *<variation_name>_phy_ddr_timing.sdc* file and click **Add**.
4. Click **OK**.

Add Pin and DQ Group Assignments

The pin assignment script, *<variation_name>_pin_assignments.tcl*, sets up the I/O standards for the DDR3 SDRAM interface. It also launches the DQ group assignment script, *<variation_name>_phy_assign_dq_groups.tcl*, which relates the DQ and DQS pin groups together for the fitter to place them correctly in the Quartus II software.

This script does not create a clock for the design. You need to create a clock for the design and provide pin assignments for the signals of both the example driver and testbench that the MegaCore variation generates.

Run the `<variation_name>_pin_assignments.tcl` to add the pin, I/O standards, and DQ group assignments to the example design.

Set Top-Level Entity

Before compiling the design, set the top-level entity of the project to the correct entity. The ALTMEMPHY megafunction entity is `<variation_name>_phy.v` or `vhd`; the SDRAM high-performance controller entity is `<variation_name>.v` or `vhd`.

The example top-level design, which instantiates the SDRAM high-performance controller and an example driver, is `<variation_name>_example_top.v` or `vhd`.

To set the top-level file, follow these steps:

1. Open the top-level entity file, `<variation_name>_example_top.v` or `vhd`.
2. On the Project menu click **Set as Top-Level Entity**.

Set Optimization Technique

To ensure the remaining unconstrained paths are routed with the highest speed and efficiency, set the optimization technique to **Speed**. To set the optimization technique, follow these steps:

1. On the Assignments menu click **Settings**.
2. Select **Analysis & Synthesis Settings**.
3. Select **Speed** under **Optimization Technique**. Click **OK**.

Set Fitter Effort



To set the fitter effort to Standard Fit, follow these steps:

1. On the Assignments menu click **Settings**.
2. Expand **Fitter Settings**.
3. Turn on **Optimize Hold Timing** and select **All Paths**.
4. Turn on **Optimize Fast Corner Timing**.
5. Select **Standard Fit** under **Fitter Effort**.
6. Click **OK**.

Enter Pin Location Assignments

To enter the pin location assignments, follow these steps:

1. Run **Analysis and Synthesis**. On the Processing menu, point to **Start** and click **Start Analysis and Synthesis**.
2. Assign all of your pins, so the Quartus II software fits your design correctly and gives correct timing analysis. To assign pin locations for the Stratix III memory demonstration board, run the Altera-provided `S3_MB1_DDR3_PinLocations.tcl` file or manually assign pin locations by using the Pin Planner.

-  The SDRAM high-performance controller autogenerated scripts do not make any pin location assignments.
-  If you are at the design exploration phase of your design cycle and do not have any PCB defined pin locations, you should still manually define an initial set of pin constraints, which can become more specific during your development process.

To manually assign pin locations, follow these steps:



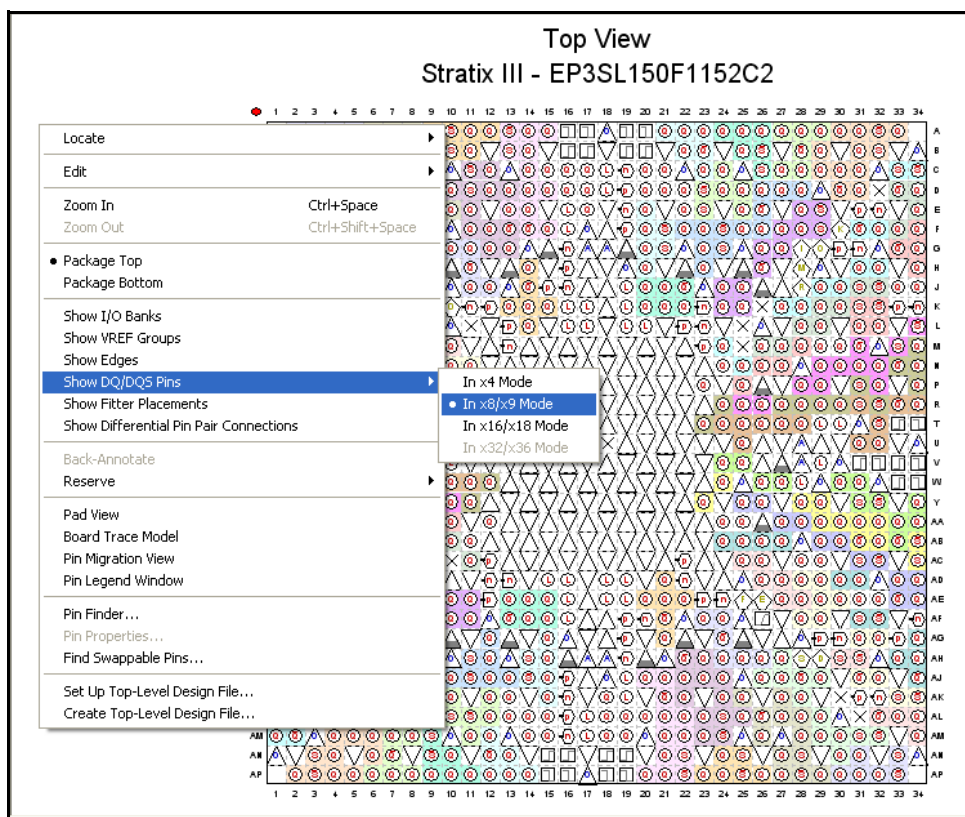
1. Open **Pin Planner**. On the Assignments menu, click **Pin Planner**.
2. Assign DQ and DQS pins.
 - a. To select the device DQS pin groups that the design uses, assign each DQS pin in your design to the required DQS pin in the Pin Planner. The Quartus II Fitter then automatically places the respective DQ signals onto suitable DQ pins within each group. To see DQS groups in Pin Planner, right click, select **Show DQ/DQS Pins**, and click **In x8/x9 Mode**. Pin Planner shows each DQS group in a different color and with a different legend: S = DQS pin, Sbar = DQSn pin and Q = DQ pin (see [Figure 13](#)).
 -  Most DDR3 SDRAM devices operate in x8/x9 mode, however as some DDR3 SDRAM devices operate in x4 mode, refer to your specific memory device datasheet.
 - b. Select the DQ mode to match the DQ group width (number of DQ pins/number of DQS pins) of your memory device. DQ mode is not related to the memory interface width.
 -  DQ group order and DQ pin order within each group is not important. However, you must place DQ pins in the same group as their respective strobe pin.

Figure 13. Quartus II Pin Planner, Show DQ/DQS Pins, In x8/x9 Mode



3. Place DM pins within their respective DQ group.
4. Place address and control command pins on any spare I/O pins ideally within the same bank or side of the device as the mem_clk pins.
5. Ensure you place mem_clk pins on differential I/O pairs for the CK/CK# pin pair. To identify differential I/O pairs, right-click in Pin Planner and select **Show Differential Pin Pair Connections**. Pin pairs show a red line between each pin pair.



You must place mem_clk[0] and mem_clk_n[0] on a DIFFIO_RX pin pair.

6. Place the clock_source pin on a dedicated PLL clock input pin with a direct connection to the SDRAM controller PLL and DLL pair—usually on the same side of the device as your memory interface. This recommendation reduces PLL jitter, saves a global clock resource, and eases timing and fitter effort.
7. Place the global_reset_n pin (like any high fan-out signal) on a dedicated clock pin.



For more information on how to use the Quartus II Pin Planner, refer to the *I/O Management* chapter in volume 2 of the *Quartus II Handbook*.

Virtual Pins

The example top-level design, which is autogenerated by the high-performance controller, includes an example driver to stimulate the interface. This example driver is not part of the SDRAM high-performance controller IP, but allows easy testing of the IP.

The example driver outputs several test signals to indicate its operation and the status of the stimulated memory interface. These signals are `pnf`, `pnf_per_byte`, and `test_complete`. These signals are not part of the memory interface, but are for testing. You should take these signals to either a debug header or set the signals to virtual pin using the Quartus II Assignment Editor. When using the example driver for testing, do not remove these signals from the top-level signal list. Otherwise the Quartus II software optimizes the driver away, and the example driver fails.

To assign virtual pin assignments for the Stratix III memory demonstration board, run the Altera-provided `s3_MB1_ddr3_exdriver_vpin.tcl` file or manually assign virtual pin assignments using the Assignment Editor.



The memory interface pins (DQ, DQS, DM, CK, CK#, address and command) cannot be assigned as virtual pins.

Advanced I/O Timing

ALTMEMPHY-based designs assume that the memory address and command signals are matched length to the memory clock signals. Typically, this length match is not true for DIMM-based designs. You should verify the difference in your design.

To amend the TimeQuest `.sdc` file, `<variation name>_phy_ddr_timing.sdc`, to include this difference, follow these steps:

1. Open the `ddr3_dimm_phy_ddr_timing.sdc` file in a text editor and find the following line (usually line 31):

```
set t(additional_addresscmd_tpd) 0.000
```

2. Change the line to the following text:

```
set t(additional_addresscmd_tpd) 0.300
```

3. Save the file.



If the DDR3 SDRAM controller `.sdc` file is regenerated, this change is lost and you must re-edit the file.

Board Trace Delay Models

For accurate I/O timing analysis, the Quartus II must be aware of the board trace and loading information. This information should be derived and refined during your PCB development process of prelayout (line) simulation and finally post-layout (board) simulation. For external memory interfaces that use memory modules (DIMMs), this information should include the trace and loading information of the module in addition to the main and host platform, which you can obtain from your memory vendor.

To enter board trace information, follow these steps:

1. In **Pin Planner**, select the pin or group of pins that you want to enter the information for.

2. Right-click and select **Board Trace Model**.



The Quartus II software does not support daisy chain board trace models. Hence the indicated board trace model for both CAC and CLK and CLK# are a simplified approximation of the DIMM topology into a lumped load at the far end of the line. The use of this simplified model is pending characterization and validation.

Figure 14 through Figure 17 show a typical board trace model for a CAC, mem_clk, DQ, and DQS pin on the Stratix III memory demonstration board including the data for the MT9JSF12872AY-1G1 memory module.

Figure 14. Stratix III Memory Demonstration Board CAC Signal Board Trace Model

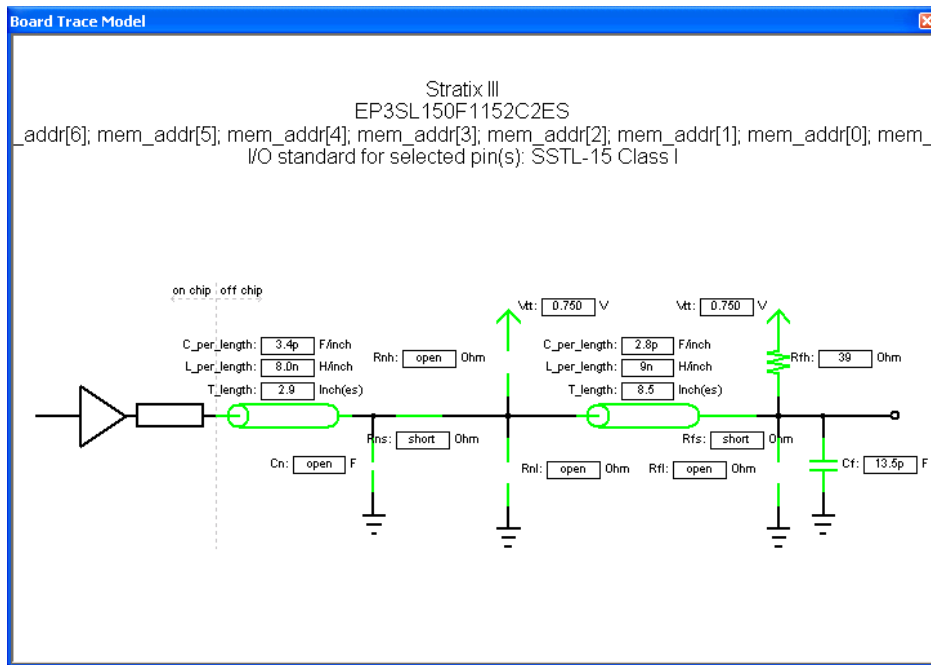


Figure 15. Stratix III Memory Demonstration Board Memory Clock Signal Board Trace Model

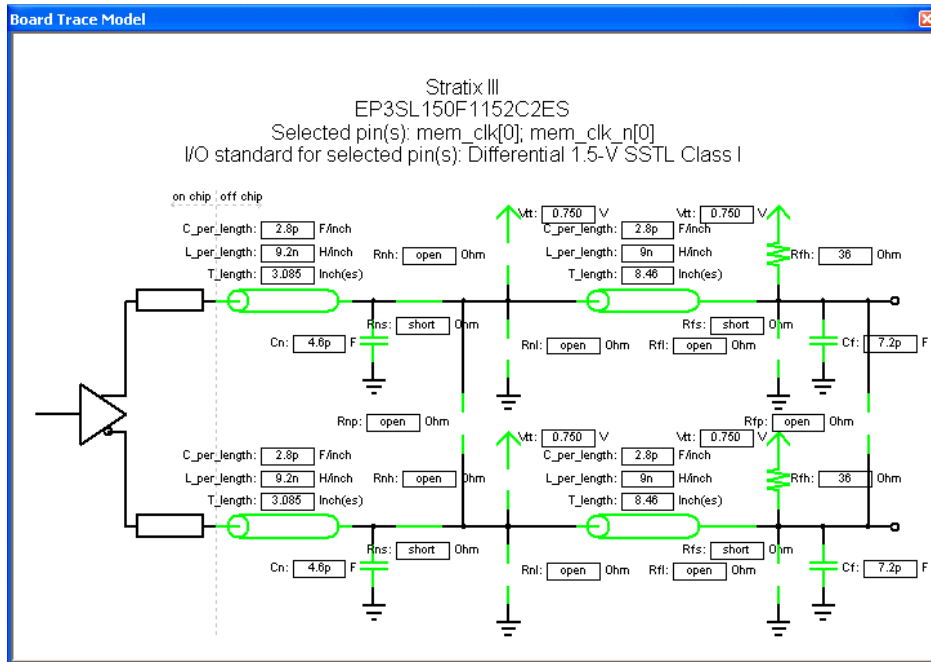


Figure 16. Stratix III Memory Demonstration Board DQ Signal Board Trace Model

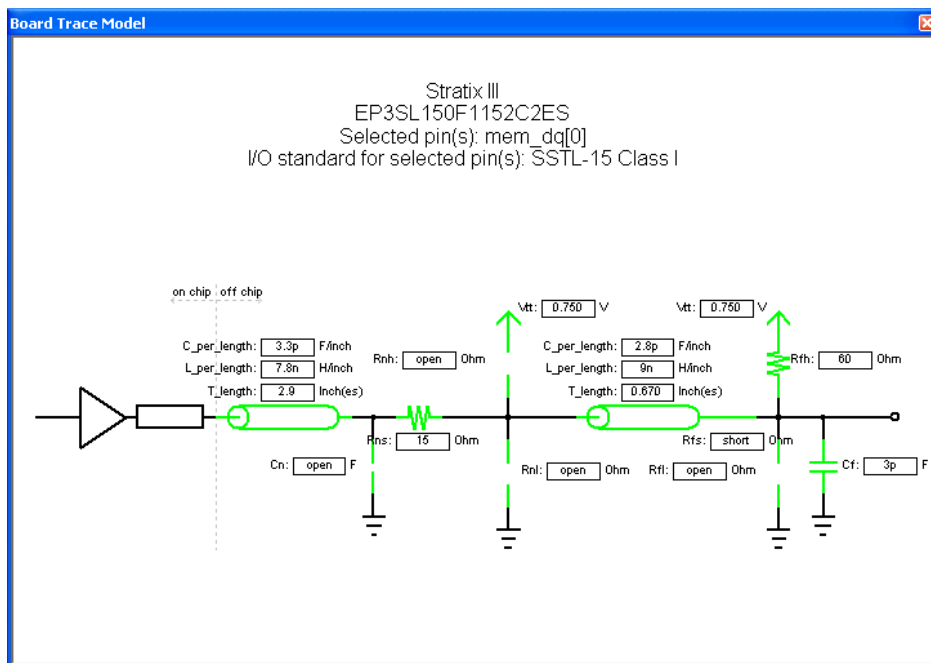


Figure 17. Stratix III Memory Demonstration Board DQS Signal Board Trace Model

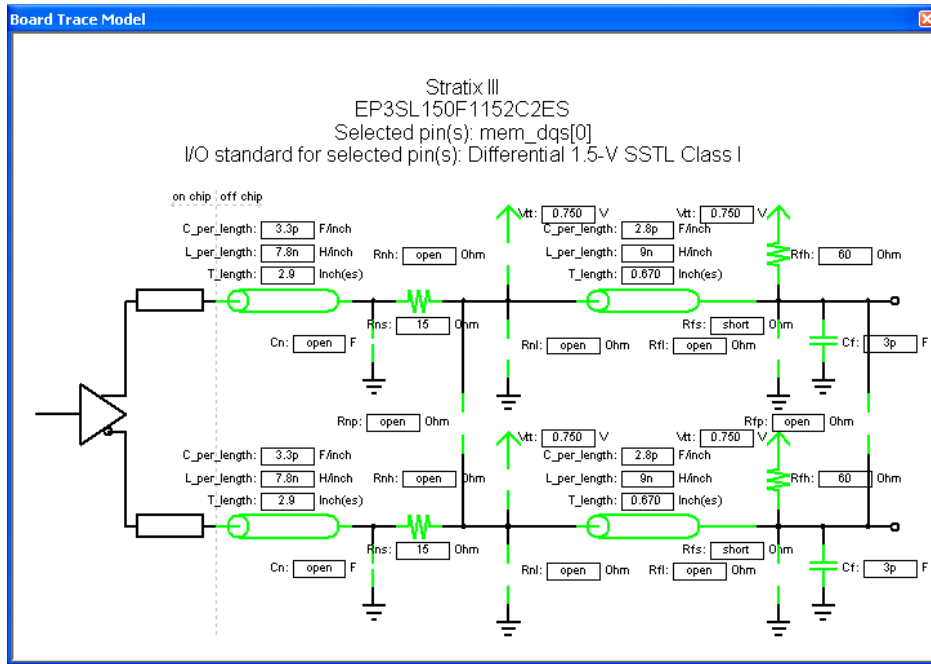


Table 7 shows the board trace model parameters for the Stratix III development board.

Table 7. Stratix III Development Board Trace Model Summary (Note 1)

Net	Near (FPGA End of Line)						Far (Memory End of Line)				
	Length	C_per_length	L_per_length	Cn	Rns	Rnh	Length	C_per_length	L_per_length	Cf	Rfh/Rfp
Addr (2)	2.904	3.5p	8.3n	—	—	—	8.488	3.75p	8.9n	13.5p	39
CLK	3.07	3.1p	9.3n	4.6p (3)	—	—	8.488	3.75p	8.9n	7.2p	36
CKE/CS#	2.937	3.5p	8.3n	—	—	—	8.480	3.75p	8.9n	13.5p	39
ODT	2.853	3.5p	8.3n	—	—	—	8.480	3.75p	8.9n	13.5p	39
DQS0	2.905	3.5p	8.3n	—	15	—	0.661	3.0p	10.7n	3p	60
DQS1	2.793	3.5p	8.3n	—	15	—	0.780	3.0p	10.7n	3p	60
DQS2	2.893	3.5p	8.3n	—	15	—	0.913	3.0p	10.7n	3p	60
DQS3	2.778	3.5p	8.3n	—	15	—	1.106	3.0p	10.7n	3p	60
DQS4	2.877	3.5p	8.3n	—	15	—	1.051	3.0p	10.7n	3p	60
DQS5	2.936	3.5p	8.3n	—	15	—	0.870	3.0p	10.7n	3p	60
DQS6	3.072	3.5p	8.3n	—	15	—	0.728	3.0p	10.7n	3p	60
DQS7	3.080	3.5p	8.3n	—	15	—	0.665	3.0p	10.7n	3p	60

Table 7. Stratix III Development Board Trace Model Summary (Note 1)

Net	Near (FPGA End of Line)						Far (Memory End of Line)				
	Length	C_per_length	L_per_length	Cn	Rns	Rnh	Length	C_per_length	L_per_length	Cf	Rfh/Rfp
DQS8	2.708	3.5p	8.3n	—	15	—	0.661	3.0p	10.7n	3p	60

Note to Table 7:

- (1) All DIMM data is preliminary and based on JEDEC R/C A (64-bit ×8 one rank DIMM). Actual DIMM is JEDEC R/C D (72-bit ×8 1 rank DIMM).
- (2) Addr = Addr, ba, we#, ras#, odt, cas.
- (3) mem_clk Cn value of 4.6 pF comprises 7 pF on memory demonstration board plus 2.2 pF on the DIMM, which is 9.2 pF differential or 4.6 pF SE.

Altera recommends you use the **Board Trace Model** assignment on all DDR3 SDRAM interface signals. To apply board trace model assignments for the Stratix III memory demonstration board, run the Altera-provided **S3_MB1_DDR3_BTModels.tcl** file or manually assign virtual pin assignments using the Quartus II Pin Planner.

The Stratix III development board has the following compensation capacitors fitted to its DDR3 SDRAM CLK and CLK# signals:

- CLK and CLK# = 7 pF (differential) compensation capacitors

These capacitors are typically fitted to designs that use nonsymmetrical DIMM designs. You should simulate your design to see if compensation capacitors are required. Stratix III devices have various programmable drive strength and OCT I/O options, so compensation capacitors should not usually be required. Fitting compensation capacitors reduces the edge rate of your signals, so you should observe memory vendor derating guidelines.



For more information on compensation capacitors, refer to *Micron Technical Note TN_47_01*.

Perform RTL or Functional Simulation (Optional)

This section describes RTL and functional simulation.

Set Up Simulation Options

To set up simulation option, follow these steps:

1. Obtain and copy the vendors memory model to a suitable location. For example, obtain the **ddr3.v** and **ddr3_parameters.vhd** memory model files from the Micron website and save them in the testbench directory.



Some vendor DIMM models do not use DM pin operation, which can cause calibration failures. In these cases, use the vendors component models directly.

2. Open the memory model file in a text editor and add the following define statements to the top of the file:

```
'define sg25
'define x8
```

The two define statements prepare the DDR3 SDRAM interface model.

The first statement specifies the memory device speed grade as –25. The second statement specifies the memory device width per DQS.

3. Open the testbench in a text editor, instantiate the downloaded memory model, and connect its signals to the rest of the design.
4. Delete the `START` and `END MEGAWIZARD` comments to ensure the MegaWizard Plug-In Manager does not overwrite the changes when the controller is regenerated.

Run Simulation with NativeLink

To run the simulation with NativeLink, follow these steps:

1. On the Assignments menu, point to **EDA Tool Settings** and click **Simulation**.
2. In the **Category** list expand **EDA Tool Settings** and click **Simulation**.
3. Under **Tool Name**, select a simulator.
4. In **NativeLink** settings, select **Compile test bench** and click **Test Benches**.
5. Click **New**.
6. Enter a name for the **Test bench name**.
7. Enter the name of the automatically generated testbench, *<variation name>_example_top_tb*, in **Top level module in test bench**.
8. Enter the name of the top-level instance in Design instance in test bench.
9. Under **Simulation period**, set **End simulation** to 100 μ s.
10. Add the testbench files and automatically-generated memory model files. In the File name field browse to the location of the memory model and the testbench, click **Open** and then click **Add**. The testbench is *<variation name>_example_top_tb.v*; memory model is *<variation name>_mem_model.v*.
11. In the **New Testbench Settings** dialog box, click **OK**.
12. Click **OK**.
13. On the Processing menu point to **Start** and click **Start Analysis and Elaboration**.
14. On the Tools menu, point to **EDA Simulation Tool** and click **Run EDA RTL Simulation**. This step creates the `\simulation` directory in your project directory and a script that compiles all necessary files and runs the simulation.



For example waveforms, refer to the *DDR3 SDRAM High-Performance Controller User Guide*.

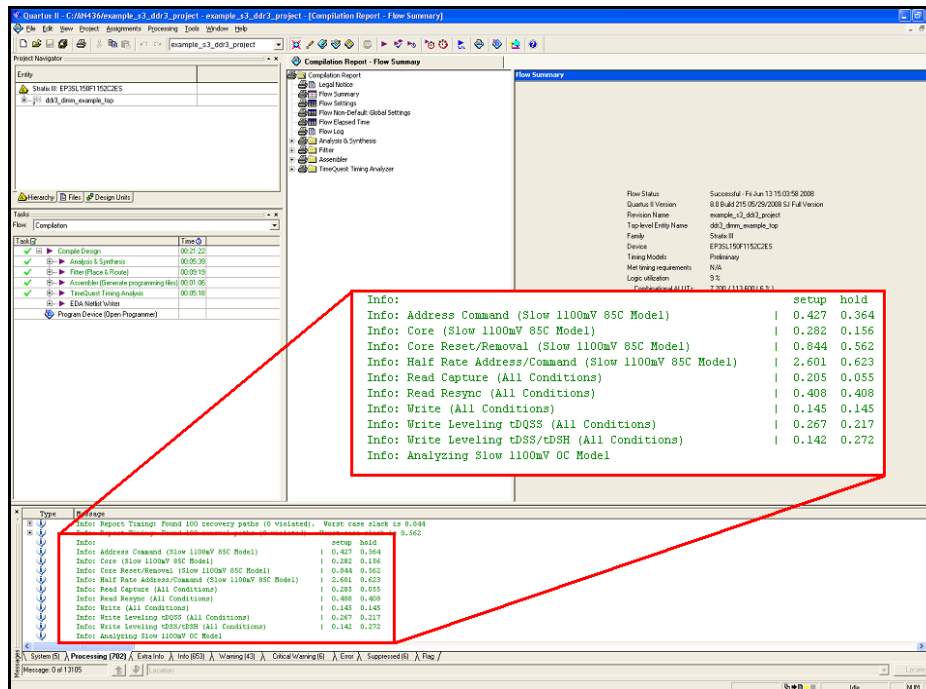
Compile Design and Verify Timing

To compile the design, on the Processing menu, click **Start Compilation**.

After successfully compiling the design, run the MegaWizard-generated verify timing script, `ddr3_dimm_phy_report_timing.tcl`, which produces a timing report for the design.

Figure 18 shows the timing margin report in the message window in the Quartus II software.

Figure 18. Timing Margin Report in the Quartus II Software



The report timing script performs the following tasks:

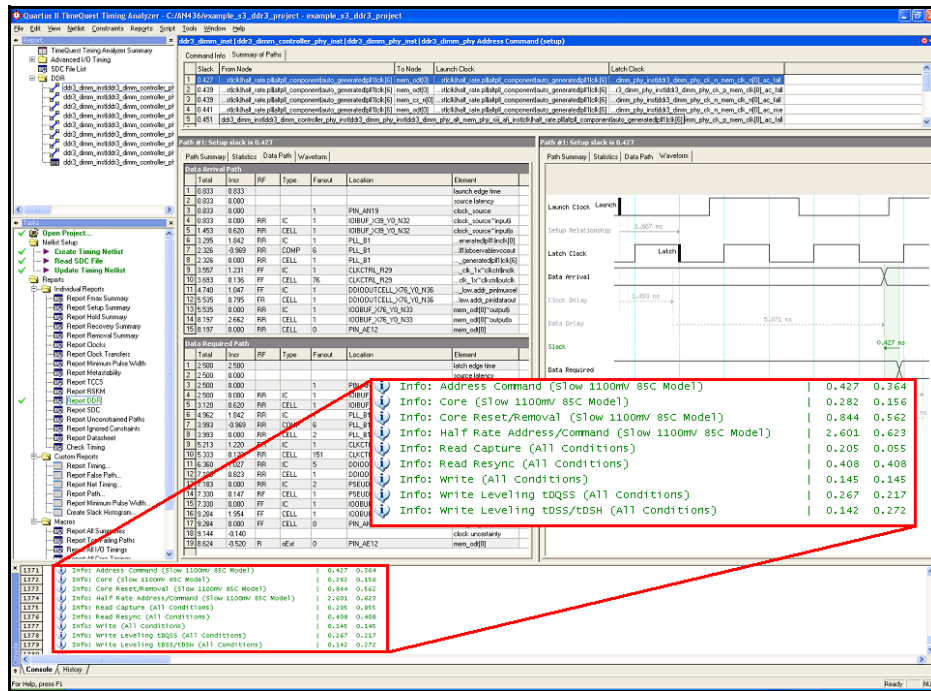
- Creates a timing netlist.
- Reads the `.sdc` file.
- Updates the timing netlist.

To run the report timing script in the TimeQuest Timing Analyzer window, follow these steps:

1. Open the panel in the Quartus II software.
2. Double-click **Update Timing Netlist** in the left pane, which automatically runs **Create Timing Netlist** and **Read SDC**. After a task is executed, it turns green.
3. After completing the tasks, run the report timing script by going to the Script menu and clicking **Run Tcl Script**.

Figure 19 shows the timing margin report in the TimeQuest Timing Analyzer window after running the report timing script. The results are the same as the Quartus II software results, as shown in Figure 18.

Figure 19. Timing Margin Report in TimeQuest Timing Analyzer



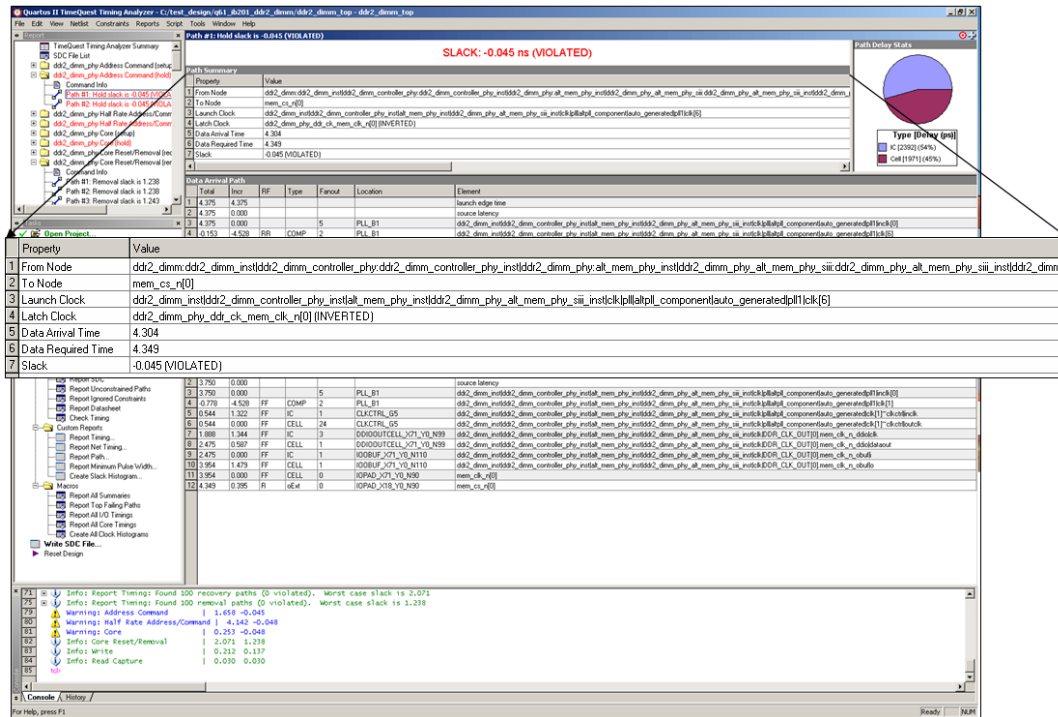
- For more information about the TimeQuest Timing Analyzer window, refer to the *Quartus II TimeQuest Timing Analyzer* chapter in volume 3 of the *Quartus II Handbook*.
- For detailed information about timing analysis, refer to *AN 438: Constraining and Analyzing Timing for External Memory Interfaces*.

Adjust Constraints

For example, if the timing margin report shows negative hold time on the address and command datapath, adjusting the clock that is regulating the address and command output registers can improve the hold margin on the address and command datapath.

To find out which clock is clocking the address and command registers, click on the address and command report in the **Report** panel in TimeQuest timing analyzer and select the path that violates the hold time, as shown in Figure 20.

Figure 20. Report on the Path that Violates Hold Time

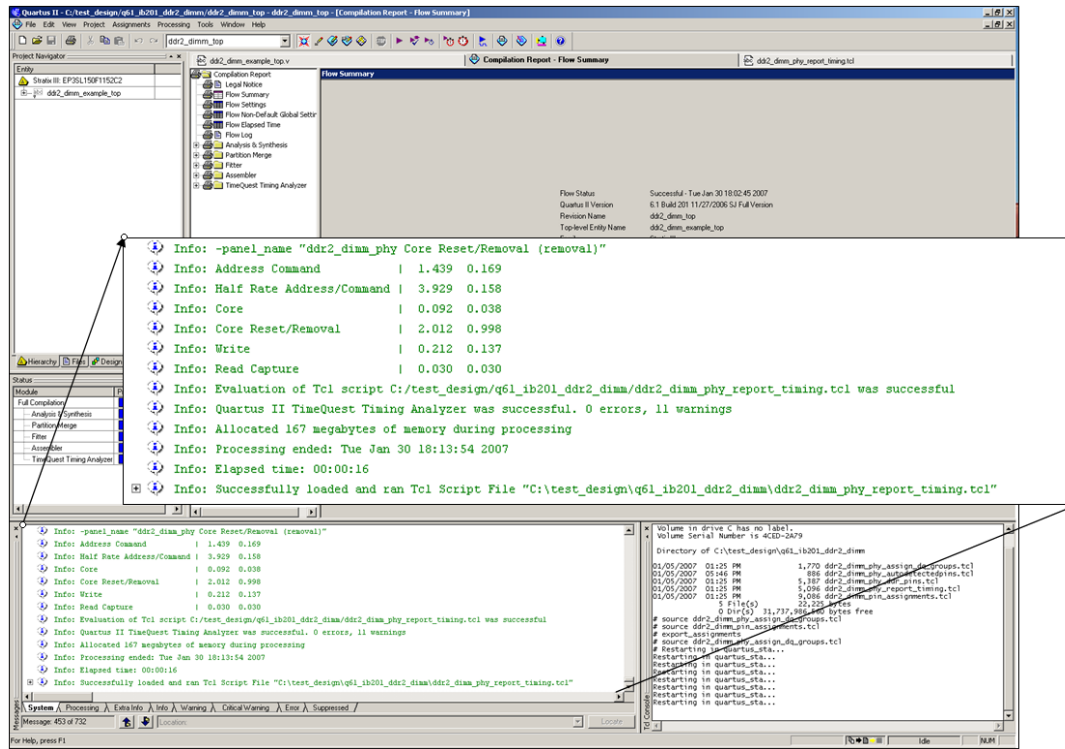


The report indicates that `clk6` of the PLL is clocking the address and command registers. Go to the PLL megafunction and change the phase setting of `clk6`. For this design, the initial phase setting of `clk6` is set to 315° , resulting in the address and command being launched too early, which causes a hold time violation. To remedy this violation, delay the launch of the address and command by delaying `clk6`, by increasing the phase setting. The negative hold margin reported is -45 ps. Therefore, delay `clk6` by an amount larger than that. Using the frequency of `clk6`, translate the amount of time delay to degrees in the PLL setting. For this example, `clk6` is 200 MHz which 45 ps translates to 3° . To ensure positive margin for hold, delay `clk6` by more than 3° , which means the new phase setting for `clk6` is larger than 318° . For this example, set the new phase setting for `clk6` to 330° so there is sufficient hold time. Alternatively, you can select a phase that balances setup and hold times.

After modifying the `clk6` phase setting, recompile the design for the new PLL setting to take effect. Run the report timing script again.

Figure 21 shows the timing margin reported in the Quartus II software after adjusting the phase setting of clk6.

Figure 21. Timing Margin Reported After Adjusting clk6



The timing report shows that all the timing margins are met.

Determine Board Design Constraints and Perform Board-Level Simulations

Stratix III devices support both series and parallel OCT resistors to improve signal integrity. Another benefit of the Stratix III OCT resistors is eliminating the need for external termination resistors on the FPGA side. This feature simplifies board design and reduces overall board cost. You can dynamically switch between the series and parallel OCT resistor depending on whether the Stratix III devices are performing a write or a read operation. The OCT features offer user-mode calibration to compensate for any variation in voltage and temperature during normal operation to ensure that the OCT values remain constant. The parallel and series OCT features of the Stratix III devices are available in either a 25-Ω or 50-Ω setting.

For more information about the OCT features, refer to the *Stratix III Device I/O Features* chapter of the *Stratix III Device Handbook*.

On DDR3 SDRAM, there is a parallel ODT feature that you can turn on when the FPGA is writing to the DDR3 SDRAM (RQZ/7 or RZQ/6) and a calibrated output impedance feature for when the FPGA is reading from the DDR3 SDRAM (RQZ/2 or RZQ/4 or RZQ/6). The parallel ODT features are available in settings of 120, 60, 40, 30, and 20-Ω.

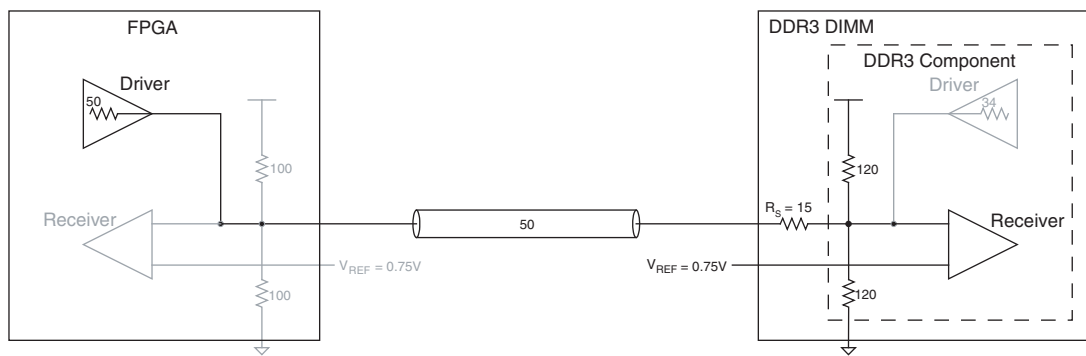
The series ODT features are available in settings of 34 and 40- Ω , although 40- Ω is not supported by all vendors.

For additional information about the available settings of the ODT, output driver impedance features and the timing requirements to drive the ODT pin in DDR3 SDRAM, refer to the respective memory data sheet .

In this setup, the transmitter (FPGA) is properly terminated with matching impedance to the transmission line, thus eliminating any ringing or reflection. The receiver (DDR3 SDRAM) is also properly terminated when the parallel ODT setting is at 60- Ω .

Figure 22 illustrates the write operation to the DDR3 SDRAM with the ODT feature turned on and using the 50- Ω series OCT feature of the Stratix III FPGA device.

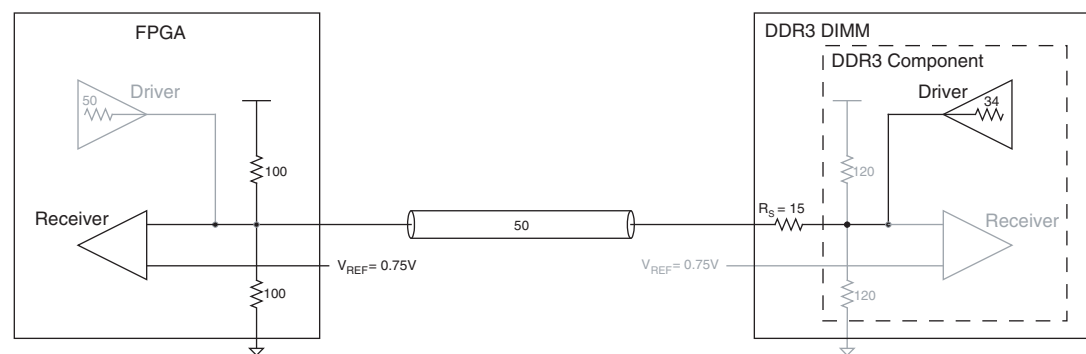
Figure 22. Write Operation Using Parallel ODT and 60- Ω Series OCT of the Stratix III FPGA Device



In this setup, the driver's (DDR3 SDRAM) output impedance is set to 34- Ω (RZQ/7) which combines with the on-DIMM series 15- Ω resistor to match the transmission line resulting in optimal signal transmission to the receiver (FPGA). On the receiver (FPGA) side, it is properly terminated with 50- Ω , which matches the impedance of the transmission line, thus eliminating any ringing or reflection.

Figure 23 illustrates the read operation from the DDR3 SDRAM using the parallel ODT feature of the Stratix III device.

Figure 23. Read Operation From DDR3 SDRAM Using the DDR3 SDRAM Output Driver Impedance Control Feature and the Stratix III Parallel OCT Feature



Finally, the loading seen by the FPGA during writes to the memory is different between a system using DIMMs versus a system using components. The additional loading from the DIMM connector can reduce the edge rates of the signals arriving at the memory thus affecting available timing margin.

For more information about Stratix III devices signal integrity, refer to www.altera.com/technology/signal/devices/stratix3/sgl-stratix3.html.

Adjust Drive Strength

Due to the loading of the line, the Quartus II software may report that the default or chosen drive strength cannot drive the line to the specified toggle rate or minimum pulse width, as shown in Figure 24. If you encounter this error, use the stronger drive strength I/O standard. Ensure that you re-simulate your design with the new drive strength to ensure that signal quality is still acceptable.

The Quartus II software v8.1 has a bug that results in an incorrect calculation for the toggle rate for differential I/O standards.

Figure 24. Minimum Pulse Width Error

Slack	Actual Width	Required Width	CCFP	Type	Clock	Clock Edge	Target
1	-15.900	2.500	10.400	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_ck_p_mem_ck[0]_ac_fall	Rise mem_ck[0]
2	-15.900	2.500	10.400	0.000	Port Rate	d02_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_ck_p_mem_ck[0]_ac_fall	Rise mem_ck[0]
3	-15.900	2.500	10.400	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_ck_p_mem_ck_r[0]_ac_fall	Rise mem_ck_r[0]
4	-15.900	2.500	10.400	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_ck_p_mem_ck_r[0]_ac_fall	Rise mem_ck_r[0]
5	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[0]	Rise mem_dq[0]
6	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[1]	Rise mem_dq[1]
7	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise mem_dq[2]
8	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[3]	Rise mem_dq[3]
9	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[4]	Rise mem_dq[4]
10	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[5]	Rise mem_dq[5]
11	-1.670	2.500	4.170	0.000	Port Rate	d02_dimm_int5592_dimm_controller_phy_int5592_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[6]	Rise mem_dq[6]
12	-1.670	2.500	4.170	0.000	Port Rate	d02_dimm_int5592_dimm_controller_phy_int5592_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[7]	Rise mem_dq[7]
13	-1.670	2.500	4.170	0.000	Port Rate	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[8]	Rise mem_dq[8]
14	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
15	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
16	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
17	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
18	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
19	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
20	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d03_dimm_d03_dimm_int5592
21	0.482	0.979	0.710	0.213	High Pulse Width	d02_dimm_int5592_dimm_controller_phy_int5592_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[0]	Rise d02_dimm_d03_dimm_int5592
22	0.482	0.979	0.710	0.213	High Pulse Width	d02_dimm_int5592_dimm_controller_phy_int5592_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[1]	Rise d02_dimm_d03_dimm_int5592
23	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
24	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
25	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
26	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
27	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
28	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
29	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[1]	Rise d03_dimm_d03_dimm_int5592
30	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
31	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
32	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
33	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
34	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
35	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
36	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
37	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
38	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
39	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5593_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592
40	0.482	0.979	0.710	0.213	High Pulse Width	d03_dimm_int5593_dimm_controller_phy_int5593_dimm_phy_int5592_dimm_phy_d03_dqin_mem_dq[2]	Rise d03_dimm_d03_dimm_int5592

Verifying Design on a Board

The SignalTap II logic analyzer shows read and write activity in the system.

For more information on using the SignalTap II logic analyzer, refer to the following documents:

- *Design Debugging Using the SignalTap II Embedded Logic Analyzer* chapter in the *Quartus II Handbook*

- *AN 323: Using SignalTap II Embedded Logic Analyzers in SOPC Builder Systems*
- *AN 446: Debugging Nios II Systems with the SignalTap II Logic Analyzer*

To add the SignalTap II logic analyzer, follow these steps:



For this design Altera provides a Tcl file, **S3_MB1_DDR3_SignalTap.tcl**, to automate the following steps. The **.stp** file is included in the application note **.zip** file.

1. On the Tools menu click **SignalTap II Logic Analyzer**.
2. In the **Signal Configuration** window next to the **Clock** box, click ... (Browse Node Finder).
3. Type `*phy_clk` in the **Named** box, for **Filter** select **SignalTap II: pre-synthesis** and click **List**.
4. Select `ddr3_dimm | ddr3_dimm_inst | phy_clk` in **Nodes Found** and click > to add the signal to **Selected Nodes**.
5. Click **OK**.
6. Under Signal Configuration, specify the following settings:
 - For **Sample depth**, select **512**
 - For **RAM type**, select **Auto**
 - For **Trigger flow control**, select **Sequential**
 - For **Trigger position**, select **Center trigger position**
 - For **Trigger conditions**, select **1**
7. On the Edit menu, click **Add Nodes**.
8. Search for specific nodes by typing `*local*` in the **Named** box, for **Filter** select **SignalTap II: pre-synthesis** and click **List**.

9. Select the following nodes in **Nodes Found** and click > to add to **Selected Nodes**:

- local_address
- local_rdata
- local_rdata_valid
- local_read_req
- local_ready
- local_wdata
- local_wdata_req
- local_write_req
- pnf
- pnf_per_byte
- test_complete (trigger)
- ctl_cal_success
- ctl_cal_fail
- ctl_wlat
- ctl_rlat



Do not add any DDR3 SDRAM interface signals to the SignalTap II logic analyzer. The load on these signals increases and adversely affects the timing analysis.

10. Click **OK**.

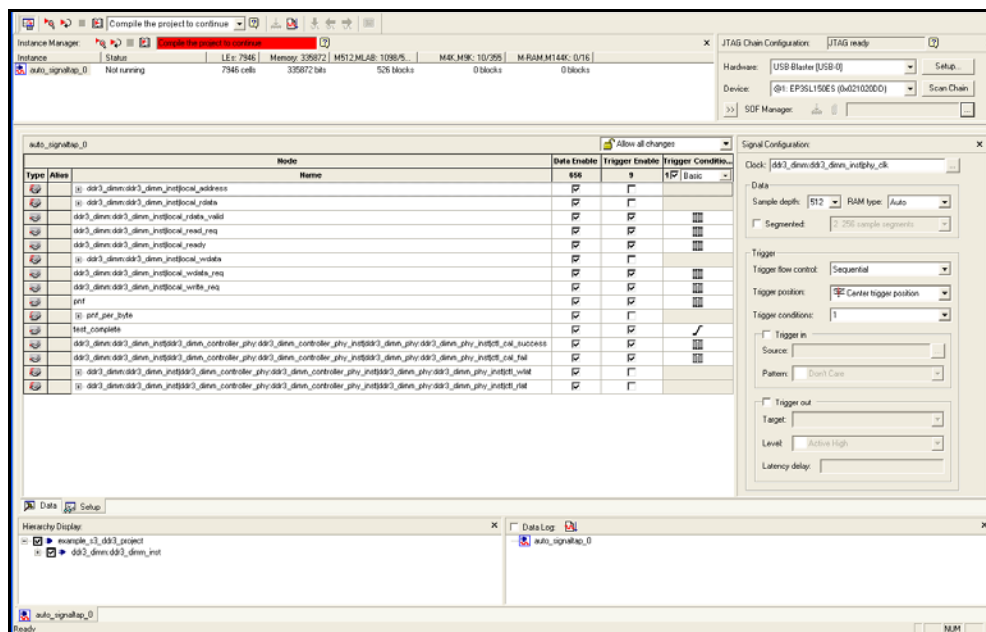
11. To reduce the SignalTap II logic size, turn off **Trigger Enable** on the following bus signals:

- local_address
- local_rdata
- local_wdata
- pnf_per_byte
- ctl_wlat
- ctl_rlat

12. Right-click **Trigger Conditions** for the test_complete signal and select **Rising Edge**.

Figure 25 shows the completed SignalTap II logic analyzer.

Figure 25. SignalTap II Logic Analyzer



13. On the File menu, click **Save**, to save the SignalTap II **.stp** file to your project.



If you see the message **Do you want to enable SignalTap II file "stp1.stp" for the current project**, click **Yes**.

Compile the Project

Once you add signals to the SignalTap II logic analyzer, recompile your design, on the Processing menu, click **Start Compilation**.

Verify Timing

Once the design compiles, ensure that TimeQuest timing analysis passes successfully. In addition to this FPGA timing analysis, check your PCB or system SDRAM timing. To run timing analysis, run the `*_phy_report_timing.tcl` script.

1. On the Tools menu, click **Tcl Scripts**.
2. Select `<variation name>_phy_report_timing.tcl` and click **Run**.

Connect the Development Board

Connect the development board to your computer.

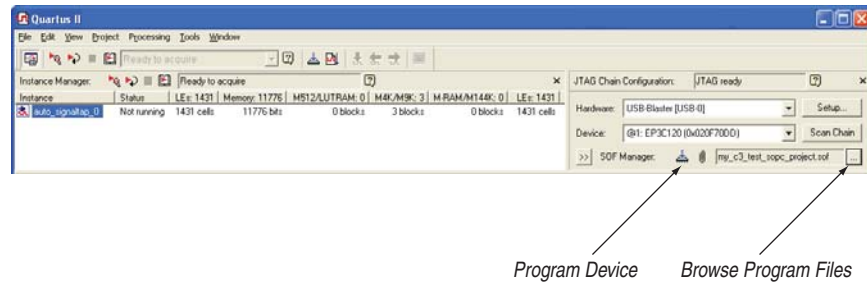
Download the Object File

On the Tools menu, click **SignalTap II Logic Analyzer**. The SignalTap II dialog box appears.

The SOF Manager should contain the `<your project name>.sof` file. To add the correct file to the SOF Manager, follow these steps:

1. Click ... to open the **Select Program Files** dialog box (see Figure 26).
2. Select <your project name>.sof.
3. Click **Open**.
4. To download the file, click the **Program Device** button (see Figure 26).

Figure 26. Install the SRAM Object File in the SignalTap II Dialog Box

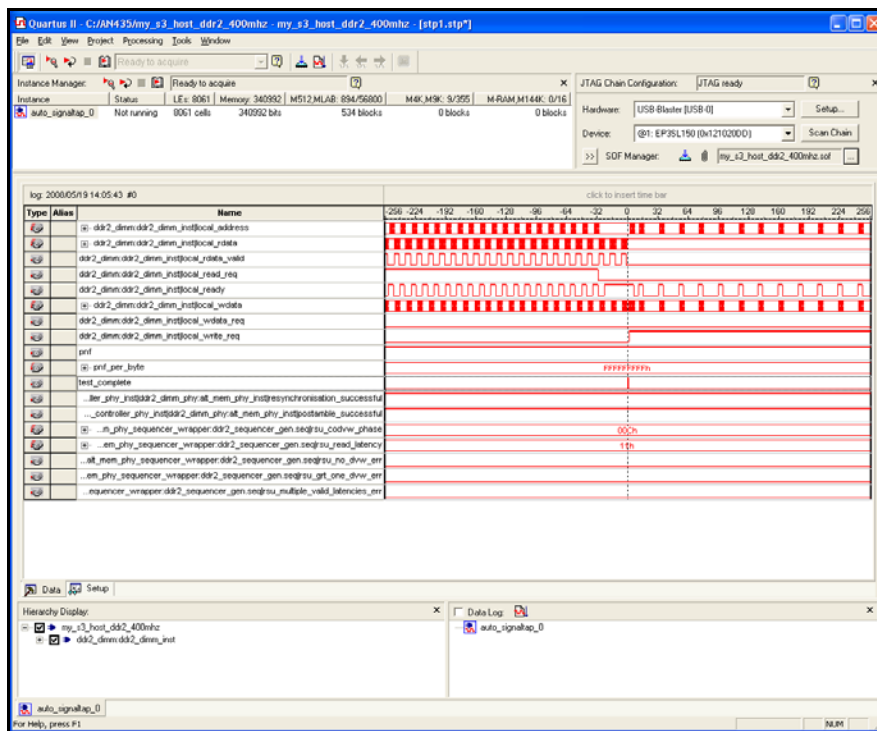


Test the Example Design in Hardware

When the example design including SignalTap II successfully downloads to your development board, click **Run Analysis** to run once, or click **Autorun Analysis** to run continuously.

Figure 27 shows the design analysis.

Figure 27. SignalTap II Example DDR3 SDRAM Design Analysis



Conclusion

Stratix III devices have dedicated circuitry to interface with DDR3 SDRAM at speeds up to 400 MHz (800 Mbps) with comfortable and consistent margins. The advanced clocking features available in Stratix III devices allow for a high-performance, versatile interface to DDR3 SDRAM. System designers can enhance their Stratix III system performance through the use of commercial off-the-shelf SDRAM without increasing cost. Altera offers a complete, proven memory solution in Stratix III devices for DDR3 SDRAM, which allows you to use these devices in applications requiring lower power consumption and greater bandwidth.

References

- *External Memory Interfaces* chapter of the *Stratix III Device Handbook*
- *External Memory Interfaces* chapter of the *Stratix IV Device Handbook*
- External DDR Memory PHY Interface (ALTMEMPHY) Megafunction User Guide
- DDR3 SDRAM High-Performance Controller User Guide
- ALTDLL and ALTDQ_DQS Megafunctions User Guide
- AN 520: DDR3 SDRAM Interface Termination and Layout Guidelines
- AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction
- AN 438: Constraining and Analyzing Timing for External Memory Interfaces in Cyclone III and Stratix III Devices
- Stratix III Device I/O Features
- The Quartus II TimeQuest Timing Analyzer
- *Design Debugging Using the SignalTap II Embedded Logic Analyzer* chapter in the *Quartus II Software Handbook*
- JEDEC Standard Publication JESD79-3A, DDR3 SDRAM Specification, JEDEC Solid State Technology Association
- MT41J128M8, 1 GB: ×8, DDR3 SDRAM Data Sheet, Micron Technology, Inc
- MT9JSF12872AY-1G1, 1 GB: ×72 DDR3 SDRAM DIMM Data Sheet, Micron Technology, Inc
- *Selecting the Right High-Speed Memory Technology for Your System* white paper

Document Revision History

Table 8 shows the revision history for this document.

Table 8. Document Revision History

Date and Document Version	Changes Made	Summary of Changes
October 2008 v4.0	<ul style="list-style-type: none"> ■ Updated “DDR3 SDRAM Overview” chapter ■ Replaced Figure 2 on page 10 ■ Replaced Figure 3 on page 11 ■ Updated “Instantiate PHY and Controller in a Quartus II Project” chapter 	—
August 2008 v3.0	Significant rewrite.	<ul style="list-style-type: none"> ■ Updated for the Quartus II software version 8.0 ■ Added Stratix IV and HardCopy III devices ■ Updated walkthrough to target Stratix III Memory Demonstration Kit
December 2007 v2.0	Updated and added new figures, and added a new section.	—
February 2007 v1.0	Initial release	—



101 Innovation Drive
San Jose, CA 95134
www.altera.com
Technical Support
www.altera.com/support

Copyright © November 2008. Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.



I.S. EN ISO 9001