External Memory Interface Design Guidelines for Stratix II, Stratix II GX, and Arria GX Devices

September 2007, v1.2 Application Note 449

Introduction

Stratix® II and Stratix II GX devices offer support for double data rate (DDR) memories, such as DDR2 / DDR SDRAM, QDRII+ / QDRII SRAM, and RLDRAM II interfaces, while Arria® GX devices support DDR2 and DDR SDRAM interfaces. Applications with these memory interfaces include PCs, embedded processing, image processing, storage, communications, and networking.

Designing applications with memory interfaces can be challenging, considering the different requirements that the memory standards impose on the controller. Moreover, changing a portion of the interface can affect many other things in the application, such that designers have to pay close attention to each step of the design process. Knowing the proper design flow is crucial in reducing design and debug time to get the application up and running.

Table 1 summarizes the maximum clock rate Stratix II and Stratix II GX devices can support with external memory devices. Depending on the target clock rate performance, you can use either the DLL- or PLL-based implementation for your memory interface. While both solutions use pre-determined data read strobe or clock (DQS) and data (DQ) pins listed in the pin tables, the DLL-based solution also uses dedicated circuitry to phase-shift the data read strobe to capture the read data. The PLL-based solution allows the Stratix II side I / O banks to interface with memory devices. However, this solution ignores the data read strobe and uses a PLL output to capture the read data. Therefore, the PLL-based solution has a lower maximum supported clock rate, because the advantage of the tight skew between data and data read strobe or clock signals from the memory devices is lost when the data read strobe or clock is not being used to capture data.

Table 1. Stratix II and Stratix II GX Maximum Clock Rate Support for External Memory Interfaces (Part 1 of 2) Note (1), (2)

Memory Standards	-3 Speed Grade (MHz)		-4 Speed Grade (MHz)		-5 Speed Grade (MHz)	
	DLL-Based	PLL-Based	DLL-Based	PLL-Based	DLL-Based	PLL-Based
DDR2 SDRAM *(3)*	333 *(4)*	200	267	167	233	167
DDR SDRAM *(3)*	200	150	200	133	200	100
RLDRAM II	300	200	250 *(5)*	175	200	175

Table 1. Stratix II and Stratix II GX Maximum Clock Rate Support for External Memory Interfaces (Part 2 of 2) Note (1), (2)

Memory Standards	-3 Speed Grade (MHz)		-4 Speed Grade (MHz)		-5 Speed Grade (MHz)	
	DLL-Based	PLL-Based	DLL-Based	PLL-Based	DLL-Based	PLL-Based
QDRII+ SRAM	300	–	–	–	–	–
QDRII SRAM	300	200	250	167	250	167

Notes to Table 1:

(1) 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, as well as static timing analysis of the completed design.
(2) These specifications apply to both commercial and industrial devices.
(3) These specifications are applicable for interfaces with both modules and discrete memory devices.
(4) Stratix II and Stratix II GX devices in the -3 speed grade support DDR2 SDRAM interfaces up to 267 MHz when using the legacy integrated static data path and controller, and up to 333 MHz when using ALTMEMPHY.
(5) You must underclock a 300-MHz RLDRAM II device to achieve this clock rate.

The legacy integrated static data path and controller refers to the physical interface (PHY) that uses the static resynchronization clock, available for all DDR memory interfaces supported by Stratix II and Stratix II GX device families, while ALTMEMPHY uses a dynamically calibrated resynchronization clock. ALTMEMPHY is only available for DDR and DDR2 SDRAM interfaces when using Stratix II, Stratix II GX, or Arria GX devices.

Table 2 summarizes the maximum clock rate Arria GX devices can support with external memory devices. Arria GX devices only support DDR2 and DDR SDRAM interfaces using ALTMEMPHY. This device family does not support QDRII+ / QDRII SRAM or RLDRAM II interfaces. Furthermore, Arria GX devices do not support DDR2 and DDR SDRAM interfaces with the legacy integrated static data path and controller.

Table 2. Arria GX Maximum Clock Rate Support for External Memory Interfaces Notes (1), (2) (Part 1 of 2)

Memory Standards	–6 Speed Grade (MHz)
DDR2 SDRAM *(3)*, *(4)*	233

Table 2. Arria GX Maximum Clock Rate Support for External Memory Interfaces Notes (1), (2) (Part 2 of 2)	
Memory Standards	**–6 Speed Grade (MHz)**
DDR SDRAM (3), (4)	200

Notes to Table 2:

(1) 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, as well as static timing analysis of the completed design.
(2) These specifications are applicable to both commercial and industrial devices, although they are preliminary until characterization is final.
(3) These specifications are applicable for interfaces with both modules and components.
(4) These memory interfaces are only supported with the ALTMEMPHY-based memory controllers.

For more information about the respective memory interface implementation, refer to:

- *AN 325: Interfacing RLDRAM II with Stratix II, Stratix, and Stratix GX Devices*
- *AN 326: Interfacing QDRII SRAM with Stratix II, Stratix, and Stratix GX Devices*
- *AN 327: Interfacing DDR SDRAM with Stratix II Devices*
- *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*

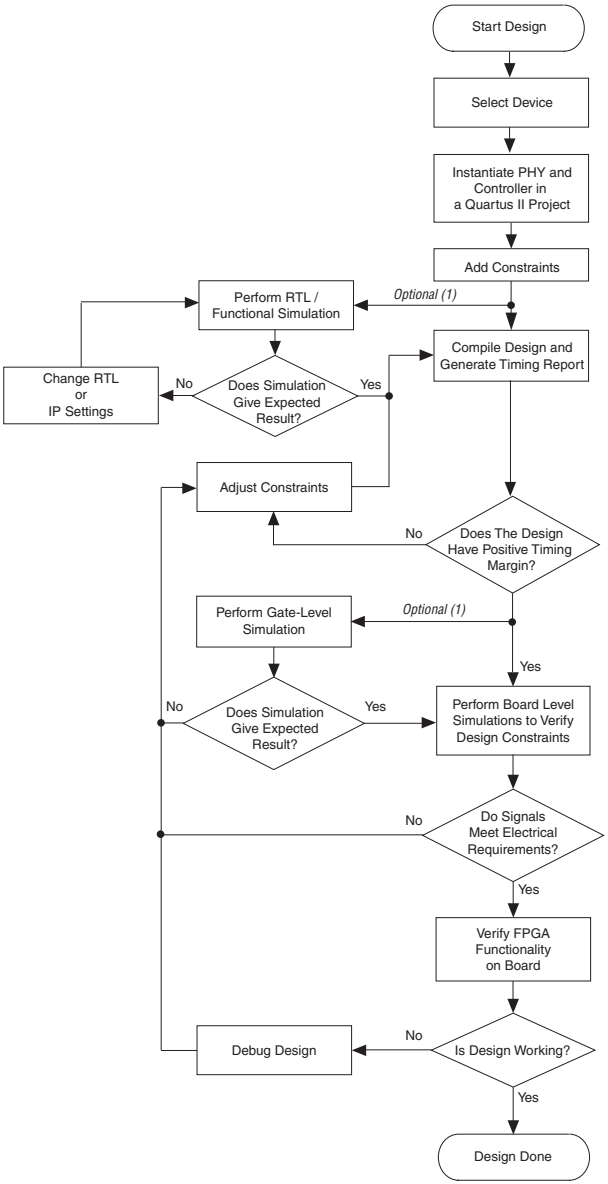
This application note describes the typical memory interface design flow for Stratix II, Stratix II GX, and Arria GX devices, and provides literature links offered by Altera® that are pertinent to the specific point you have reached in the design cycle.

Designs targeting the Stratix II device, up to 267 MHz, can be migrated to HardCopy® II structured ASICs. For the purpose of this application note, any Altera FPGA discussion referring to a Stratix II FPGA can also apply to a HardCopy II structured ASIC. Memory interfaces in HardCopy II structured ASICs, however, have additional restrictions, such as the usage of the PLL dedicated clock output pins for CK / CK# signals. These restrictions are described in *AN 463: Using the ALTMEMPHY Megafunction with HardCopy II Structured ASICs* and in *AN 413: Using Legacy Integrated Static Data Path and Controller Megafunction with HardCopy II Structured ASICs*, depending on the PHY that you are using.

Stratix II, Stratix II GX, and Arria GX Memory Interface Design Flow

Altera recommends the design guidelines illustrated in Figure 1 as best practices for successful memory interface implementation in Stratix II, Stratix II GX, and Arria GX devices. These guidelines provide the fastest out-of-the-box experience with external memory interfaces in Stratix II, Stratix II GX, and Arria GX devices. A detailed discussion of each step presented in Figure 1 appears in the following sections.

Figure 1. Design Flow for Implementing External Memory Interfaces in Stratix II, Stratix II GX, and Arria GX FPGAs



Note to Figure 1:

(1) Even though this is an optional step, Altera recommends that you perform this step to ensure design functionality.

Table 3 shows a summary of collateral reference for each design step in Figure 1.

AN328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices describes each step in the memory design flow in detail with an example to create a DDR2 SDRAM interface, using both ALTMEMPHY and the legacy integrated static data path and controllers, with the Stratix II GX PCI Express Development Board.

Table 3. Corresponding Collateral with the Memory Interface Design Flow (Part 1 of 3) *Note (1), (2)*

Design Steps	Variation of Steps	Reference
Select device	ALTMEMPHY-based memory controllers	• *Selecting the Right High-Speed Memory Technology for Your System* White Paper • *External Memory Interfaces* chapter of the *Stratix II*, *Stratix II GX*, or *Arria GX Device Handbook*
Instantiate PHY and controller in a Quartus® II project	ALTMEMPHY-based memory controllers	• *DDR and DDR2 SDRAM High Performance Controller User Guide* • *ALTMEMPHY Megafunction User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices* • *AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction*
	Legacy integrated static data path and controllers	• *DDR and DDR2 SDRAM Controller Compiler User Guide* • *QDRII SRAM Controller User Guide* • *RLDRAM II Controller User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices* • *AN 327: Interfacing DDR SDRAM with Stratix II Devices* • *AN 326: Interfacing QDRII SRAM with Stratix II, Stratix, and Stratix GX Devices* • *AN 325: Interfacing RLDRAM II with Stratix II, Stratix II GX, and Arria GX Devices* • *AN 392: Implementing Multiple Legacy DDR/DDR2 SDRAM Controller Interfaces* • *AN 398: Using DDR/DDR2 SDRAM with SOPC Builder*

Table 3. Corresponding Collateral with the Memory Interface Design Flow (Part 2 of 3) Note (1), (2)

Design Steps	Variation of Steps	Reference
Add constraints	ALTMEMPHY-based memory controllers	• *DDR and DDR2 SDRAM High Performance Controller User Guide* • *ALTMEMPHY Megafunction User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II Devices*
	Legacy integrated static data path and controllers	• *DDR and DDR2 SDRAM Controller Compiler User Guide* • *QDRII SRAM Controller User Guide* • *RLDRAM II Controller User Guide* • *DDR Timing Wizard (DTW) User Guide* • *AN 408: DDR2 Memory Interface Termination, Drive Strength and Loading Design Guidelines* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
Perform RTL/functional simulation	ALTMEMPHY-based memory controllers	• *AN 380: Test DDR or DDR2 SDRAM Interfaces on Hardware Using the Example Driver* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
Compile design and generate timing report	ALTMEMPHY-based memory controllers	• *DDR and DDR2 SDRAM High Performance Controller User Guide* • *ALTMEMPHY Megafunction User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
	Legacy integrated static data path and controllers	• *DDR Timing Wizard (DTW) User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
Perform gate-level simulation		• *Verification* section of the *Quartus II Software Handbook, Volume 3* • *DDR and DDR2 SDRAM High Performance Controller User Guide* • *DDR and DDR2 SDRAM Controller Compiler User Guide* • *QDRII SRAM Controller User Guide* • *RLDRAM II Controller User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
Adjust constraints	ALTMEMPHY-based memory controllers	• *DDR and DDR2 SDRAM High Performance Controller User Guide* • *ALTMEMPHY Megafunction User Guide* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*
	Legacy integrated static data path and controllers	• *DDR and DDR2 SDRAM Controller Compiler User Guide* • *QDRII SRAM Controller User Guide* • *RLDRAM II Controller User Guide* • *DDR Timing Wizard (DTW) User Guide* • *AN 408: DDR2 Memory Interface Termination, Drive Strength and Loading Design Guidelines*

Table 3. Corresponding Collateral with the Memory Interface Design Flow (Part 3 of 3) Note (1), (2)		
Design Steps	**Variation of Steps**	**Reference**
Perform board-level simulation to verify design constraints		• *AN 408: DDR2 Memory Interface Termination, Drive Strength and Loading Design Guidelines* • *AN 444: Dual DIMM DDR2 SDRAM Memory Interface Design Guidelines* • *AN 315: Guidelines for Designing High-Speed FPGA PCBs* • *AN 224: High-Speed Board Layout Guidelines* • *AN 75: High-Speed Board Designs*
Verify functionality on hardware		• *Chapter 13: Design Debugging Using the SignalTap II Embedded Logic Analyzer* of the *Quartus II Software Handbook, Volume 3* • *AN 380: Test DDR or DDR2 SDRAM Interfaces on Hardware Using the Example Driver* • *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices*

Note to Table 3:

(1) Go to Altera's External Memory Solutions Center at **http://www.altera.com/technology/memory/mem-index.jsp** for the most up-to-date collateral for Stratix II devices.

(2) For specific information on instantiating a memory controller or closing timing in a memory interface design, refer to the Quartus II Release Notes of the version of the Quartus II software you are using.

Step 1: Select Device

Prior to the start of designing any memory interface, determine the required bandwidth of the memory interface. Bandwidth can be expressed as:

Bandwidth = Data width (bits) × data rate transfer (1 / sec) × efficiency

The efficiency is the percentage of time the data bus is transferring data. It is dependent on the type of memory and the design's access profile. For example, in a memory interface with separate read and write ports, the efficiency is 100% when there is an equal amount of reads and writes on these memory interfaces. On the other hand, when the read and write ports are shared, the efficiency of the interface is less than 50% due to the time to switch from a read to a write operation, or vice versa.

After calculating the bandwidth of the memory interface, determine which memory and FPGA to use. Altera has a memory selection white paper, which highlights the differences between the memory types. Refer to the *Selecting the Right High-Speed Memory Technology for Your System* white paper for information about selecting the different memory types.

Altera's FPGA device families support various data widths and maximum performance 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.

Refer to the *External Memory Interfaces* chapter of the *Stratix II*, *Stratix II GX*, or *Arria GX Device Handbook* for information about the respective device density and package support for the different memory types.

Consider memory interface target performance, device resource availability, and PHY implementation support when choosing a target device for your memory controller. Decide whether all the memory interface signals can fit in the top or bottom I/O banks to take advantage of the dedicated DQS phase-shift circuitry in the FPGAs. You also need at least one PLL per memory interface depending on the PHY implementation that you use.

Table 4 shows a summary of the number of pins required for various memory interfaces.

Table 4 assumes that series OCT with calibration or parallel OCT with calibration, or dynamic calibrated OCT is used, which is shown by the usage of R_{UP} and R_{DN} pins.

Table 4. Pin Counts for Various Memory Interfaces Notes (1), (2) (Part 1 of 2)

Memory Interface	Bus Width	Number of DQ Pins	Number of DQS Pins	Number of DM/BWSn Pins	Number of Address Pins (3)	Number of Command Pins	Number of Clock Pins	R_{UP}/R_{DN} Pins	Total Pins
DDR2 SDRAM (4)	×4	4	1	0 (5)	15	9	2	2	33
	×8	8	1	1	15	9	2	2	38
	×16	16	2	2	14	9	2	2	47
	×72	72	9	9	14	12	6	2	124
DDR SDRAM (6)	×4	4	1	0 (5)	14	7	2	2	28
	×8	8	1	1	14	7	2	2	33
	×16	16	2	2	14	7	2	2	43
	×72	72	9	9	13	9	6	2	118

Table 4. Pin Counts for Various Memory Interfaces Notes (1), (2) (Part 2 of 2)

Memory Interface	Bus Width	Number of DQ Pins	Number of DQS Pins	Number of DM/BWSn Pins	Number of Address Pins (3)	Number of Command Pins	Number of Clock Pins	R_{UP}/R_{DN} Pins	Total Pins
QDRII+ SRAM	×9	18	2	1	19	3 (7)	4	2	49
	×18	36	2	2	18	3 (7)	4	2	67
	×36	72	2	4	17	3 (7)	4	2	104
QDRII SRAM	×9	18	2	1	19	2	4	2	48
	×18	36	2	2	18	2	4	2	66
	×36	72	2	4	17	2	4	2	103
RLDRAM II Common (CIO)	×9	9	2	1	22	7	4	2	47
	×18	18	2	1	21	7	6	2	57
	×36	36	2	1	20	7	8	2	76
RLDRAM II Separate (SIO)	×9	18	2	1	22	7	4	2	56
	×18	36	2	1	21	7	6	2	75
	×36	72	2	1	20	7	8	2	112

Notes to Table 4:

(1) These pin counts example pin numbers are derived from memory vendor data sheets. You need to check the exact number of addresses and command pins of the memory devices in the configuration that you are using.
(2) PLL and DLL input reference clock pins are not counted in this calculation.
(3) The number of address pins depend on the memory device density.
(4) Numbers are based on 2-GB memory devices without using differential DQS, RDQS, and RDQS# pin support.
(5) Altera FPGAs do not support DM pins in ×4 mode.
(6) Numbers are based on 1-GB memory devices.
(7) QVLD pin is included in this number.

Expanding your memory interfaces for width or depth with the same memory controller (and shared address and command bus) is supported natively by the MegaWizard Plug-In Manager.

Creating multiple memory controllers with independent memory transactions (independent address and command buses) and shared device resources between multiple memory interfaces may require RTL and design constraint modifications.

The maximum number of interfaces you can implement on any given device is limited by resource availability (number of DQ groups of desired width, user I/O pins, PLLs, DLLs, clocks, and FPGA core resources). For example, the Stratix II GX EP2SGX90FF1508C3 FPGA supports nine ×8 DQ groups each on the top and bottom I/O banks. The

DLL located on the top (or bottom) can be shared amongst all nine DQ groups on that side of the device as long as they are implementing memory interfaces running at the same frequency and using the same base phase-shift.

Stratix II, Stratix II GX, and Arria GX devices allow a base phase-shift of 22.5°, 30°, or 36°. You can use one to four delay buffers to get a maximum of 90°, 120°, or 144° from this base phase shift. In a multiple memory controller design, you can then have one controller using 90° phase-shift and another controller using the 67.5° phase shift with the 22.5° base phase shift. Each memory interface requires at least one PLL, but you can share the clocks between multiple interfaces if they are running at the same frequency and using the same PLL phase-shift.

For multiple controller considerations, refer to *AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction* or *AN 392: Implementing Multiple Legacy DDR/DDR2 SDRAM Controller Interfaces*, depending on the physical interface (PHY) chosen.

Step 2: Instantiate PHY and Controller in a Quartus II Project

When instantiating the data path for your memory interface, Altera recommends that you use the data path from the Altera memory controller MegaCore® functions and megafunctions listed in Table 5 with their corresponding user guides.

Table 5. Altera Memory Controller MegaCore Functions and Megafunctions

Memory Interface	MegaCore Function or Megafunction	User Guide
DDR and DDR2 SDRAM	DDR and DDR2 SDRAM High-Performance Controller	*DDR and DDR2 SDRAM High-Performance Controller User Guide*
	ALTMEMPHY Megafunction *(1)*	*ALTMEMPHY Megafunction User Guide*
	DDR and DDR2 SDRAM Controller Compiler *(2)*	*DDR and DDR2 SDRAM Controller Compiler User Guide*
QDRII SRAM	QDRII SRAM Controller *(2)*	*QDRII SRAM Controller MegaCore Function User Guide*
RLDRAM II	RLDRAM II Controller *(2)*	*RLDRAM II Controller MegaCore Function User Guide*

Note to Table 5:

(1) The ALTMEMPHY megafunction is the data path used in the DDR and DDR2 SDRAM High Performance Controller. This is unlike the integrated static data path and controllers where the data path is not offered as a stand-alone megafunction.

(2) These MegaCore functions, also known as the legacy integrated static data path and controllers, are not available for Arria GX devices.

Table 6 lists the differences between the legacy integrated static data path and controller and the ALTMEMPHY-based controller.

Table 6. Comparing the Legacy Integrated Static Data Path and Controller Compiler and the ALTMEMPHY-Based Controller

Legacy Integrated Static Data Path and Controller	ALTMEMPHY-Based Controller
Up to 267 MHz in Stratix II and Stratix II GX devices Not supported in Arria GX devices	Up to 333 MHz in Stratix II and Stratix II GX devices Up to 233 MHz in Arria GX devices
Integrated data path and controller	Separate datapath (available in the ALTMEMPHY megafunction) and controller
Static phase-shift selection for resynchronization	Auto-calibrated resynchronization clock
Requires board trace length measurements	Does not require board trace length information
1-PLL or 2-PLL mode	1-PLL solution
DLL-based or PLL-based read capture	DLL-based read-capture only
Full-rate controller	Full-rate or half-rate controller
Classic Timing Analyzer and TimeQuest Timing Analyzer support	TimeQuest Timing Analyzer support only
DDR2/DDR SDRAM, RLDRAM II, QDRII+/QDRII SRAM	DDR and DDR2 SDRAM only

As listed in Table 5 on page 11, there are three different functions available for DDR and DDR2 SDRAM interfaces in Stratix II and Stratix II GX devices. The ALTMEMPHY megafunction, however, is included in the DDR and DDR2 SDRAM High-Performance controller, and is only to be instantiated separately when you are going to use your own controller. Therefore, Stratix II and Stratix II GX devices actually offer two different PHYs for DDR and DDR2 SDRAM interfaces:

- DDR and DDR2 SDRAM High-Performance Controller with ALTMEMPHY

  Use this controller for all new designs, especially if you are planning to migrate to a Stratix III device or if you are running at a frequency higher than 200 MHz. If latency is more important than ease of timing closure, however, use the DDR and DDR2 SDRAM Controller Compiler MegaCore function.

  This is the only MegaCore function supported in Arria GX devices. This DDR and DDR2 SDRAM High-Performance Controller MegaCore function requires the use of the TimeQuest Timing Analyzer and only supports memory interfaces on the top and bottom I/O banks of the Stratix II, Stratix II GX, and Arria GX devices, as it requires a DLL for read capture.

This MegaCore function instantiates the ALTMEMPHY megafunction for its datapath with two options:

- Full-rate mode

  In this mode, the system clock for the PHY and the memory interface clock are of the same frequency.

- Half-rate mode

  In this mode, the system clock of the PHY is half of the frequency of the memory interface clock. The read and write data are demultiplexed and multiplexed, respectively, to run at half the frequency in the FPGA core. This mode uses more device resources than the full-rate mode because of the data multiplexing and demultiplexing. This mode also incurs more latency than the full-rate mode.

ALTMEMPHY-based controllers use one DLL and one PLL per instance. You can share the DLL and some of the clock outputs from the PLL in a multiple-interfaces design. Refer to *AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction* for more information on sharing resources when designing for multiple ALTMEMPHY-based controllers.

Table 7 shows the maximum clock rate supported for the ALTMEMPHY-based memory interfaces in Stratix II, Stratix II GX, and Arria GX devices. DDR SDRAM interfaces are supported up to 200 MHz in all speed grades of these device families.

Table 7. Maximum Clock Rate Support for ALTMEMPHY-Based DDR2 SDRAM Interfaces in Stratix II, Stratix II GX, and Arria GX Devices

Device Family	Speed Grade	Half-Rate Mode (MHz) *(1)*	Full-Rate Mode (MHz)
Stratix II and Stratix II GX	-3	333	267
	-4	267	233
	-5	233	200
Arria GX	-6	233	200

Note to Table 7:

(1) These specifications are also listed in Table 1 on page 1 and Table 2 on page 2.

There are two ways to instantiate the ALTMEMPHY megafunction: using the ALTMEMPHY MegaWizard® Plug-In Manager or using Altera's DDR and DDR2 SDRAM High Performance Controller, which already includes the ALTMEMPHY megafunction. Even if you plan to use your own controller, Altera recommends that you first create a design using Altera's DDR and DDR2 SDRAM High Performance Controller and then replace the Altera controller with your own controller. In this way, you get an example design which you can simulate and verify.

For more information on the DDR and DDR2 SDRAM High-Performance Controller MegaCore function or the ALTMEMPHY megafunction, refer to the *DDR and DDR2 SDRAM High Performance Controller User Guide* or the *ALTMEMPHY Megafunction User Guide*, respectively.

- DDR and DDR2 SDRAM Controller Compiler

  Use this controller only if you cannot use the ALTMEMPHY or the High-Performance controllers, for example, if you need a lower latency memory controller than offered by the ALTMEMPHY-based controllers, or if you want to use the PLL-based read capture. This implementation offers both DLL-based and PLL-based read capture. In PLL-based read capture, the DLL is not used, allowing the side I/O banks to interface with memory devices. The PLL-based read capture has lower maximum performance compared to the DLL-based read capture as the DQ timing has to be referenced to the memory clock, since DQS is ignored, in the PLL-based read capture.

  The Arria GX device family does not support this implementation, hence does not support memory interfaces in side I/O banks.

  The DLL-based read capture implementation of this controller is offered with two options:

  - One-PLL implementation

    This implementation, limited to 200-MHz performance, uses one of the system PLL output clocks to resynchronize data from the DQS domain to the system clock domain.

  - Two-PLL implementation

    This implementation, limited to 267 MHz, uses a second PLL, called the fedback PLL, to generate the resynchronization and postamble clocks, in addition to the system PLL.

The PLL-based read capture implementation also offers one- and two-PLL implementation options:

- One-PLL implementation

  In this implementation, the system PLL generates all the clocks needed for the interface including system, write, read-capture, and resynchronization clock.

- Two-PLL implementation

  In this implementation, the system PLL generates all the clocks needed for the interface including system, write, and resynchronization clock, while the fedback PLL generates the read-capture clock.

Since the DQS is ignored, postamble clock is not needed in a PLL-based read capture implementation.

For more information on the DDR and DDR2 SDRAM Controller Compiler MegaCore function, refer to the *DDR and DDR2 SDRAM Controller Compiler User Guide*.

The legacy integrated static data path and controllers for DDR SDRAM, DDR2 SDRAM, QDRII SRAM, and RLDRAM II include an integrated clear-text data path (or PHY) and encrypted controller, supporting both DLL-based and PLL-based implementations. These legacy integrated static data path and controllers are not supported in Arria GX devices. When using your own memory controller with the legacy integrated static data path and controllers, you must manually extract the PHY from the encrypted controller. Maximum frequency for the legacy integrated static data path and controller depends on the timing reported by DTW in the Quartus II software, which in turn depends on:

- FPGA density
- Memory speed grade
- Board trace length skew

After selecting the appropriate FPGA device, memory type, and PHY implementation (wherever applicable), create a project in the Quartus II software and instantiate the PHY and controller for your design.

You can use either the MegaWizard Plug-In Manager or the SOPC Builder to instantiate and customize your memory controller. Use SOPC Builder if you are going to use the external memory as a peripheral of a NIOS soft processor.

Refer to *AN 398: Using DDR/DDR2 SDRAM with SOPC Builder* for more information about instantiating a legacy integrated static data path and controller with the SOPC Builder.

Refer to the respective MegaCore User Guide for information on instantiating a memory controller in the FPGA. In addition, *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Device* includes step-by-step information on instantiating a DDR2 SDRAM memory interface targeted to the Stratix II GX PCI Development Board.

If you plan to have multiple memory controllers, refer to *AN 462: Implementing Multiple Memory Interfaces Using the ALTMEMPHY Megafunction* when using ALTMEMPHY-based controllers, or *AN 392: Implementing Multiple Legacy DDR/DDR2 SDRAM Controller Interfaces* when using the legacy integrated static data path and controllers.

The following sections cover design flow steps for both the legacy integrated static data path and controllers and ALTMEMPHY-based controllers.

Step 3: Add Constraints

The next step in the design process is to add all timing, location, and physical constraints related to the external memory interface. This includes timing, pin locations, I/O standards, termination, drive strength, and pin loading assignments.

Refer to *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Devices* for step-by-step information on adding constraints to a DDR2 SDRAM interface targeted to the Stratix II GX PCI Development Board.

To determine which drive strength and termination to use, refer to *AN 408: DDR2 Memory Interface Termination, Drive Strength and Loading Design Guidelines*.

ALTMEMPHY-Based Controllers

The top level of the DDR and DDR2 SDRAM High-Performance Controller design is fully timing-constrained using Synopsys design constraints (SDC), which are analyzed by the TimeQuest Timing Analyzer. These timing assignments are a function of the parameters you enter in the ALTMEMPHY MegaWizard Plug-In or the DDR and DDR2 SDRAM High Performance Controller MegaWizard Plug-In, and are derived from the memory datasheet and tolerances from your board

layout. The ALTMEMPHY megafunction uses TimeQuest timing constraints and the timing-driven fitter to achieve timing closure. After you instantiate the ALTMEMPHY megafunction, the MegaWizard Plug-In Manager generates the following script files that you must use in order to properly constrain the design.

- <*variation_name*>**_phy_ddr_timing.sdc**
- <*variation_name*>**_pin_assignments.tcl**

These script files are based on the output file name used when generating the ALTMEMPHY megafunction or the DDR and DDR2 SDRAM High Performance Controller MegaCore function. If you plan to use your own top-level design, you will need to edit the pin names in the <*variation_name*>**_pin_assignments.tcl** script to match your custom top-level design.

For more information on ALTMEMPHY or the high performance controller, refer to the *ALTMEMPHY Megafunction User Guide* or the *DDR and DDR2 SDRAM High Performance Controller User Guide.*

Legacy Integrated Static Data Path and Controllers

When using the legacy integrated static data path and controller MegaCore functions, in the MegaWizard window, select the DQS and DQ pin location from the **Constraints** dialog box. When you click **Generate**, the MegaWizard creates the pin location, I/O standard, and loading assignments based on the memory device, FPGA device, and board information in a **.tcl** file called **add_constraints_for_<***variation_name***>.tcl**. In addition, the MegaWizard generates a **.tcl** file called **auto_add_<***ddr* | *qdrii* | *rldramii***>_constraints.tcl** that contains all the constraints for multiple memory controllers of the same type in the project. For example, if the project has two DDR2 SDRAM memory controllers, constraints for both controllers can be sourced using the **auto_add_ddr_constraints.tcl** file.

To add timing constraints for the legacy integrated static data path and controllers, run the **DDR Timing Wizard** (DTW).

Although DTW supports both the classic timing analyzer and the TimeQuest Timing Analyzer, Altera recommends using the TimeQuest Timing Analyzer for more accurate results, as the DTW-generated SDC file covers the constraints for both fast and slow timing models.

For more information about how to use the DTW, refer to the *DDR Timing Wizard (DTW) User Guide.*

For more information on the **add_constraints_for_**<*variation_name*>**.tcl** script, refer to the respective legacy integrated static data path and memory controller user guides.

Step 4: Perform RTL/Functional Simulation (Optional)

Simulating the design to verify functionality is crucial in ensuring that the application performs as designed in your system. Therefore, Altera recommends that you perform this optional step for your memory interface designs.

The test bench file for both implementations is saved in the <*project_directory*>**/testbench** folder.

The models work with Altera-supported VHDL and Verilog HDL simulators. Download the model for the actual memory device that you are using from the vendor's website.

Refer to *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Device* for step-by-step information on performing RTL / functional simulation for DDR2 SDRAM interfaces in Stratix II, Stratix II GX, and Arria GX devices.

ALTMEMPHY-Based Controllers

In the ALTMEMPHY MegaWizard Plug-In Manager, there is an option to generate a simulation model of the design in either Verilog HDL or VHDL. This IP functional simulation model is a cycle-accurate HDL model file produced by the Quartus II software. Altera's DDR and DDR2 SDRAM High Performance Controller generates an example design and a testbench, in addition to the ALTMEMPHY–megafunction simulation model.

Legacy Integrated Static Data Path and Controllers

Similarly, when you instantiate the memory controller using the Altera legacy integrated static data path and memory controller MegaCore functions, you can select a cycle-accurate Verilog HDL or VHDL functional simulation model and generate a test bench for the example design.

For more information on simulating the controller, refer to the *Appendix C. Perform Functional Simulation* section of *AN 380: Test DDR or DDR2 SDRAM Interfaces on Hardware Using the Example Driver* or the respective memory controller user guides.

Step 5: Compile Design to Generate Timing Report

Once the constraints are added to your design, compile the design to ensure that it meets timing requirements.

Refer to *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Device* for step-by-step information on closing timing for a DDR2 SDRAM interface targeted to the Stratix II GX PCI Development Board.

ALTMEMPHY-Based Controllers

During the generation of the ALTMEMPHY megafunction or the High Performance SDRAM Memory Controller, the MegaWizard Plug-In Manager generates a report timing script called <*variation_name*>**_report_timing.tcl**, which you can run after compiling the design to produce a timing report for different paths, such as write data, read data and command / address, and controller timing paths in the design. You can also run the **Report DDR** macro in the TimeQuest Timing Analyzer module to get the timing report for all memory interfaces in your design.

For more information on ALTMEMPHY or the high performance controller, refer to the *ALTMEMPHY Megafunction User Guide* or the *DDR and DDR2 SDRAM High Performance Controller User Guide.*

Legacy Integrated Static Data Path and Controllers

For memory interfaces constrained using DTW, use the **dtw_timing_analysis.tcl** script, which is included with the DDR Timing Analysis (DTW User Guide). This allows the Quartus II software to report the margin for different timing paths in the interface and to recommend the ideal phase shift settings for PLL-based read capture, resynchronization, and DLL-based postamble clocks. After compiling the design, run the **dtw_timing_analysis.tcl** script to verify timing closure. You can change the clock cycles and phase shifts of the design to the script-recommended results if timing is not met.

For more information on how to use the script and DTW, refer to the *DDR Timing Wizard (DTW) User Guide.*

When using DTW, the **verify_timing.tcl** script generated by MegaWizard for legacy integrated static data path and controllers is ignored. The DTW and **dtw_timing_analysis.tcl** provide a more complete timing analysis and report compared with the **verify_timing.tcl** script.

Step 6: Perform Gate Level Simulation (Optional)

This optional step allows you to ensure that your system meets the proper timing requirements needed by each module of the design in a graphical waveform. Altera recommends that you perform this optional step for your memory interface designs.

Refer to *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Device* for information on performing gate-level simulation on a DDR2 SDRAM interface.

Refer to the *Verification* section of the *Quartus II Software Handbook, Volume 3* or the respective memory controller User Guides for more information on simulating your controller.

Step 7: Adjust Constraints

Refer to *AN 328: Interfacing DDR2 SDRAM with Stratix II, Stratix II GX, and Arria GX Device* and the *DDR Timing Wizard (DTW) User Guide* for step-by-step information on closing timing for a DDR2 SDRAM interface targeted to the Stratix II GX PCI Development Board.

In the timing report of the design, you can see the worst-case setup and hold margins for the different paths in the design. If the setup and hold margins are unbalanced, and you wish to achieve balanced setup and hold margins, adjust the phase setting of the clocks that are used to clock these paths. For example, if the report timing script (either from <*variation_name*>**_report_timing.tcl** for controllers with the ALTMEMPHY data path, or from **dtw_timing_analysis.tcl** for legacy integrated static data path and controllers) indicates that the current resynchronization clock phase shift results in more hold time than setup time, you can add more phase shift to the resynchronization clock with respect to the system clock so that there will be less hold margin. Similarly, you use less resynchronization clock phase shift with respect to the system clock if there is more setup margin.

When using the legacy integrated static data path and controller, the **dtw_timing_analysis.tcl** script recommends the ideal clock cycles and phase shifts for the resynchronization and postamble clocks, whenever applicable. The **report_timing.tcl** for controllers with the ALTMEMPHY data path only reports the margin of the interface.

Refer to the *DDR Timing Wizard (DTW) User Guide* for more information on the **dtw_timing_analysis.tcl** script.

Step 8: Perform Board Simulation to Verify Design Constraints

To ensure you determine the correct board constraints, run board level simulations with your preferred board simulator software 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 you with an initial indication of how well the memory interface performs. The simulations should be performed on the data, data strobe, command, and address signals. If the memory interface does not have good signal integrity, adjust the settings, such as the drive strength setting, termination scheme, or termination values to improve the signal integrity (realize that changing these settings affects your timing and you may have to revise the memory interface constraints if these change). There are various electronic design automation (EDA) simulation tools available to perform board level simulations.

Changing the settings for the memory interface affects the timing. If you change them, it may be necessary to re-adjust the constraints in order to achieve timing closure.

Once you close the timing for the design, evaluate the trade-offs posed by various board design choices. Different factors contribute to signal integrity and affect the overall timing margin for the memory and the FPGA. Some factors to consider that can affect the signal integrity include the termination scheme used, the drive strength setting on the FPGA, and the loading seen by the driver. Learn the trade-offs between the different types of termination schemes, the effects of output drive strengths, and loading, so that you can swiftly navigate through the multiple combinations and choose the best possible settings for your designs.

Refer to *AN 408: DDR2 Memory Interface Termination, Drive Strength and Loading Design Guidelines* for detailed information about understanding the different effects on signal integrity design.

For systems with two DIMMs connected in parallel, refer to *AN 444: Dual DIMM DDR2 SDRAM Memory Interface Design Guidelines*.

Altera offers the following application notes regarding high-speed board design:

- *AN 315: Guidelines for Designing High-Speed FPGA PCBs*
- *AN 224: High-Speed Board Layout Guidelines*
- *AN 75: High-Speed Board Designs*

For more information on Stratix II Signal Integrity, go to:
http://www.altera.com/technology/signal/devices/stratix2/sgl-st2.html

Step 9: Verify FPGA Functionality

Perform system-level verification to ensure the functionality of the memory interfaces within your FPGA design. You can use Altera's SignalTap® II Embedded Logic Analyzer to help in this effort.

Refer to *Chapter 13: Design Debugging Using the SignalTap II Embedded Logic Analyzer* of the *Quartus II Software Handbook* in *Volume 3* for detailed information about using SignalTap II.

If you would like step-by-step instructions on how to hardware-test the example design for the legacy integrated static data path and memory controller using the SignalTap II Logic Analyzer, refer to *AN 380: Test DDR or DDR2 SDRAM Interfaces on Hardware Using the Example Driver.*

Conclusion

Memory interfaces are useful in many different applications. Stratix II, Stratix II GX, and Arria GX FPGAs offer a complete solution, including hardware, software, and documentation, that can be used to help build a robust, high-performance memory interface.

Stratix II and Stratix II GX device families allow memory interfaces to run up to 333 MHz (666 Mbps), while Arria GX device family can run up to 233 MHz, with Altera's newest memory controller that implements the ALTMEMPHY data path.

With a straightforward design flow, empowered with collateral and design examples, customers can easily implement their memory interface with confidence.

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

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
September 2007 v1.2	• Added Arria GX information • Updated the design flowchart and each step information to reflect the new flowchart • Fixed typos/hyperlink problems	—
July 2007 v1.1	Change title of referenced *AN 413, Using Legacy Integrated Static Data Path and Controller Megafunction with HardCopy II Structured ASICs*, from title, Implementing External Memory Interfaces in HardCopy II Devices. Added links to all reference documentation. Changed revision number and date.	—
February 2007 v1.0	Initial Release	—