# External Memory Interface Handbook

# Volume 1: Altera Memory Solution Overview and Design Flow

101 Innovation Drive
San Jose, CA 95134
www.altera.com

EMI_GS-2.0

Document last updated for Altera Complete Design Suite version: 12.1
Document publication date: November 2012

# Contents

# Chapter Revision Dates

The chapters in this document, External Memory Interface Handbook, were revised on the following dates. Where chapters or groups of chapters are available separately, part numbers are listed.

Chapter 1. Introduction to Altera Memory Solution
Revised: *November 2012*
Part Number: *EMI_GS_001-2.0*

Chapter 2. Recommended Design Flow
Revised: *June 2012*
Part Number: *EMI_GS_002-3.0*

# Section I. Overview

This section provides an overview of Altera memory solutions and the recommended memory IP design flow.

This section includes the following chapters:

- Chapter 1, Introduction to Altera Memory Solution
- Chapter 2, Recommended Design Flow

For information about the revision history for chapters in this section, refer to "Document Revision History" in each individual chapter.

# 1. Introduction to Altera Memory Solution

EMI_GS_001-2.0

This chapter describes the memory solutions that Altera provides.

Altera provides the fastest, most efficient, and lowest latency memory controllers. The controllers are designed to allow you to easily interface with today's higher speed memories.

Altera supports a wide variety of memory interfaces suitable for applications ranging from routers and switches to video cameras. You can easily implement Altera's intellectual property (IP) using the memory MegaCore functions through the Quartus II software. The Quartus II software also provides an external memory toolkit that helps you test the implementation of the IP in the FPGA device.

Refer to the External Memory Interface Spec Estimator page for the maximum speeds supported by Altera FPGAs.

## Soft and Hard Memory IP

Altera's latest devices, the 28-nm FPGAs provide two types of memory solutions: soft memory IP and hard memory IP. Arria V (GX, GT, SX, and ST) and Cyclone V devices offer both soft and hard memory IP, while Stratix V and Arria V GZ devices offer only soft memory IP.

The soft memory IP gives you the flexibility to design your own interfaces to meet your system requirements and still benefit from the industry leading performance. The hard memory IP is designed to give you a complete out-of-the-box experience when designing a memory controller.

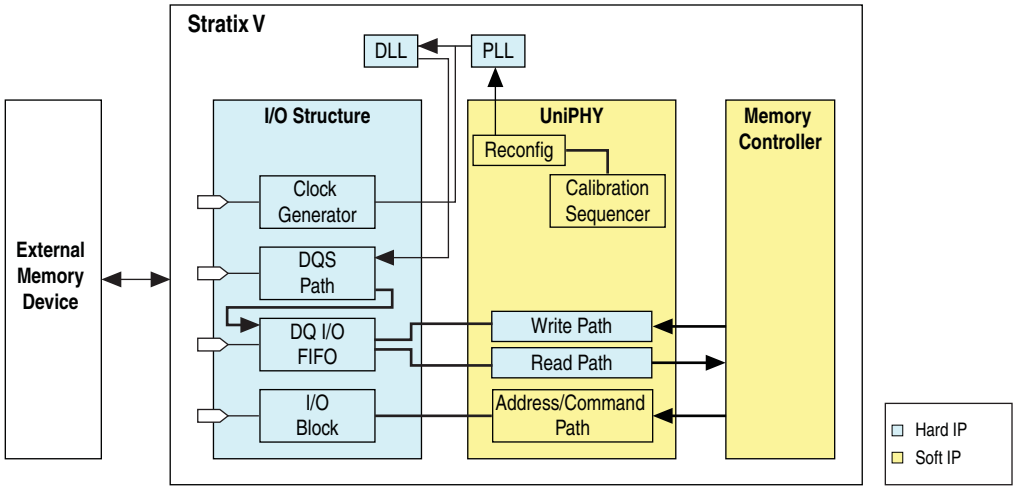
Table 1–1 lists the features of the soft and hard memory IP.

**Table 1–1. Features of the Soft and Hard Memory IP**

| Soft Memory IP | Hard Memory IP |
|---|---|
| ■ Consists of a DDR2, DDR3, or LPDDR2 SDRAM high-performance memory controller with UniPHY IP.<br>■ Has hardened read and write data paths to ensure your design meets timing at the highest speeds. The data paths include I/O, phase-locked loops (PLLs), delay-locked loop (DLL), and read and write FIFO buffers.<br>■ Allows you to choose the location to place the memory controller and the ability to size the memory controller based on the system requirements, especially in the Stratix V and Arria V GZ devices. | ■ Consists of a DDR2 DDR3, or LPDDR2 SDRAM high-performance memory controller with a hard UniPHY IP, and a multi port front-end block.<br>■ Has a fixed location on the die and a fixed maximum width; ×32 and ×40 for Arria V devices and ×16 and ×40 for Cyclone V devices.<br>■ Runs at full rate to allow decreased latency and to minimize the required bus width of signals going into the core of the device.<br>■ Simplifies the overall memory design in Arria V and Cyclone V devices, and provides a truly out-of-the-box experience for every designer. |

Figure 1–1 shows the hardened data paths in the soft memory IP of a Stratix V device.

**Figure 1–1. Hardened Data Paths in the Soft Memory IP**




# Memory Solutions

Altera FPGAs achieve optimal memory interface performance with external memory IP. The IP provides the following components:

- Physical layer interface (PHY) which handles the timing on the data path itself.
- Memory controller block which implements all the memory commands and addresses.
- Multi port front-end (MPFE) block which allows multiple processes inside the FPGA device to share a common bank of memory. The MPFE block is a new feature in Arria V and Cyclone V devices.

These MPFE blocks are critical to the design and the use of the memory interface block.

Altera provides modular memory solutions that allow you to customize your memory interface design to any of the following configurations:

- PHY with your own controller
- PHY with Altera controller
- PHY with Altera controller and the MPFE block

You can also build a custom PHY, a custom controller, or both, as desired.

Table 1–2 lists the recommended memory types and controllers that Altera offers with the PHY IP.

**Table 1–2. Altera Memory Types, PHY, and Controllers in the Quartus II Software (Part 1 of 2)**

| Quartus II Version | Memory | PHY IP | Controller IP |
|---|---|---|---|
| 12.1 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) (1) | HPC II |
| | DDR2/DDR3//LPDDR2 | UniPHY | HPC II |
| | QDR II/QDR II+ | UniPHY | QDR/RLD II controller |
| | RLDRAM 3 | UniPHY (4) | Custom (5) |
| | RLDRAM II | UniPHY | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom (5) |
| | Other | ALTDQ_DQS2 (3) | Custom (5) |
| 12.0 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) (1) | HPC II |
| | DDR2/DDR3//LPDDR2 | UniPHY | HPC II |
| | QDR II/QDR II+ | UniPHY | QDR/RLD II controller |
| | RLDRAM II | UniPHY | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom |
| | Other | ALTDQ_DQS2 (3) | Custom |
| 11.1 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) (1) | HPC II |
| | DDR2/DDR3 | UniPHY | HPC II |
| | QDR II/QDR II+ | UniPHY | QDR/RLD II controller |
| | RLDRAM II | UniPHY | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom |
| | Other | ALTDQ_DQS2 (3) | Custom |
| 11.0 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) | HPC II |
| | DDR2/DDR3 | UniPHY | HPC II |
| | QDR II/QDR II+ | UniPHY | QDR/RLD II controller |
| | RLDRAM II | UniPHY | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom |
| | Other | ALTDQ_DQS2 (3) | Custom |

**Table 1–2. Altera Memory Types, PHY, and Controllers in the Quartus II Software (Part 2 of 2)**

| Quartus II Version | Memory | PHY IP | Controller IP |
|---|---|---|---|
| 10.1 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) | HPC<br>HPC II |
| | DDR2/DDR3 | UniPHY Nios-based Sequencer | HPC II |
| | QDR II/QDR II+ | UniPHY RTL Sequencer | QDR/RLD II controller |
| | RLDRAM II | UniPHY RTL Sequencer | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom |
| | Other | ALTDQ_DQS2 (3) | Custom |
| 10.0 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) | HPC<br>HPC II |
| | DDR2/DDR3 | UniPHY Nios-based Sequencer | HPC II |
| | QDR II/QDR II+ | UniPHY RTL Sequencer | QDR/RLD II controller |
| | RLDRAM II | UniPHY RTL Sequencer | QDR/RLD II controller |
| | Other | ALTDQ_DQS (2) | Custom |
| | Other | ALTDQ_DQS2 (3) | Custom |
| 9.1 | DDR/DDR2/DDR3 | ALTMEMPHY (AFI) | HPC<br>HPC II |
| | QDR II/QDR II+ | UniPHY | QDR II controller |
| | RLDRAM II | UniPHY | RLDRAM II controller |
| | Other | ALTDQ_DQS (2) | Custom |

**Notes to Table 1–2:**

(1) AFI = Altera PHY interface
(2) Applicable for Arria II, Stratix III, and Stratix IV devices.
(3) Applicable only for Cyclone V, Arria V and Stratix V devices.
(4) Applicable only for Arria V GZ and Stratix V devices.
(5) No Altera controller support. You must create your own controllers.

For more information about the controllers with the UniPHY or the ALTMEMPHY IP, refer to the *Functional Descriptions* section in Volume 3 of the *External Memory Interface Handbook*.
For more information about the ALTDQ_DQS megafunction, refer to the *ALTDLL and ALTDQ_DQS Megafunctions User Guide*.
For more information about the ALTDQ_DQS2 megafunction, refer to the *ALTDQ_DQS2 Megafunction User Guide*.

# Protocol Support Matrix

Table 1–3 lists the device family and IP architecture support for each memory protocol.

**Table 1–3. Protocol Support Matrix** *(1) (2) (3)*

| Protocol | Family | | | | | | | | | | IP Architecture | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Stratix V/ Arria V GZ | Arria V GX, GT, SX, ST | Cyclone V | Stratix IV | Stratix III | Arria II GZ | Arria II GX | Cyclone III | Cyclone IV | HardCopy III/IV | Hard/Soft | Rate | Burst Length | Sequencer | Controller |
| DDR3 | — | U | U | — | — | — | — | — | — | — | Hard | Full | 8 | Nios II | HPC II |
| | U | U | U | U | U | U | A | — | — | U | Soft | Half | 8 | Nios II | HPC II |
| | U | U | — | — | — | — | — | — | — | — | Soft | Quarter | 8 | Nios II | HPC II |
| DDR2 | — | U | U | — | — | — | — | — | — | — | Hard | Full | 4,8 | Nios II | HPC II |
| | U | — | — | U | U | U | A | A | A | U | Soft | Full | 4,8 | Nios II | HPC II |
| | U | U | U | U | U | U | A | A | A | U | Soft | Half | 4,8 | Nios II | HPC II |
| LPDDR2 | — | U | U | — | — | — | — | — | — | — | Soft | Half | 4,8,16 | Nios II | HPC II |
| | — | — | U | — | — | — | — | — | — | — | Hard | Full | 4,8,16 | Nios II | HPC II |
| RLDRAM 3 | U | — | — | — | — | — | — | — | — | — | Soft | Half | 2,4,8 | Nios II | — |
| | U | — | — | — | — | — | — | — | — | — | Soft | Quarter | 2,4,8 | Nios II | — |
| RLDRAM II | U | — | — | U | U | U | — | — | — | U | Soft | Full | 2,4,8 | RTL | RLDRAM II |
| | U | U | — | U | U | U | — | — | — | U | Soft | Half | 4,8 | Nios II | RLDRAM II |
| | U | U | — | U | U | U | — | — | — | U | Soft | Half | 4,8 | RTL | RLDRAM II |
| QDR II/II+ | U | — | — | U | U | U | U | — | — | U | Soft | Full | 2,4 | RTL | QDR II/II+ |
| | U | U | — | U | U | U | — | — | — | U | Soft | Half | 4 | Nios II | QDR II/II+ |
| | U | U | — | U | U | U | U | — | — | U | Soft | Half | 4 | RTL | QDR II/II+ |

**Notes to Table 1–3:**

(1) **U**= Supported by UniPHY-based IP.

(2) **A** = Supported by ALTMEMPHY-based IP.

(3) **--** = Not supported.

# Low Latency

Altera generally offers low latency solutions that are significantly better than Altera's competitors. Altera's 28-nm FPGA devices have a balanced clocked network in the periphery to reduce switching noise. The hardened read data FIFO buffer guarantees timing and makes it easier for the fitter to place the controller. Together with the latest UniPHY IP, these design changes provide major reduction in latency.

Table 1–4 lists latency comparison for Altera and its closest competition.

**Table 1–4. Latency Comparison for Quarter-Rate DDR3 SDRAM Controllers**

| Latency Type | Latency (Memory Clock Cycles) | | Advantage |
|---|---|---|---|
| | Competitor (1) | Altera | |
| Write Command | 46 | 29 | Altera |
| Read Command | 46 | 29 | Altera |
| Read Data | 31 | 11 | Altera |

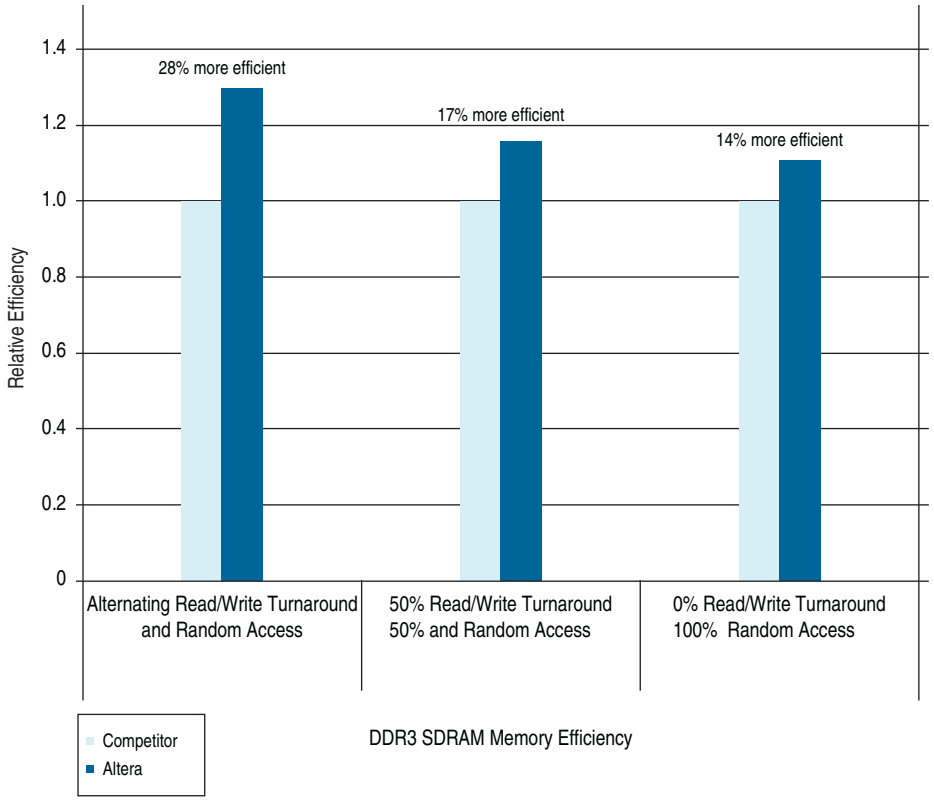
**Note to Table 1–4:**

(1) Does not include AXI latency.

# Efficiency

Altera memory controllers are also highly efficient. Figure 1–2 shows the memory efficiency of a DDR3 SDRAM memory controller with UniPHY IP.

**Figure 1–2. Memory Efficiency of DDR3 SDRAM Memory Controllers with UniPHY**



# Document Revision History

Table 1–5 lists the revision history for this document.

**Table 1–5. Document Revision History**

| Date | Version | Changes |
|---|---|---|
| November 2012 | 2.0 | ■ Added Arria V GZ information.<br>■ Added RLDRAM III information to "Protocol Support Matrix" and "Memory Solutions". |
| June 2012 | 1.2 | Change to Table 1–3. |
| June 2012 | 1.1 | ■ Added "Protocol Support Matrix".<br>■ Added Feedback icon. |
| November 2011 | 1.0 | Initial release. |

# 2. Recommended Design Flow

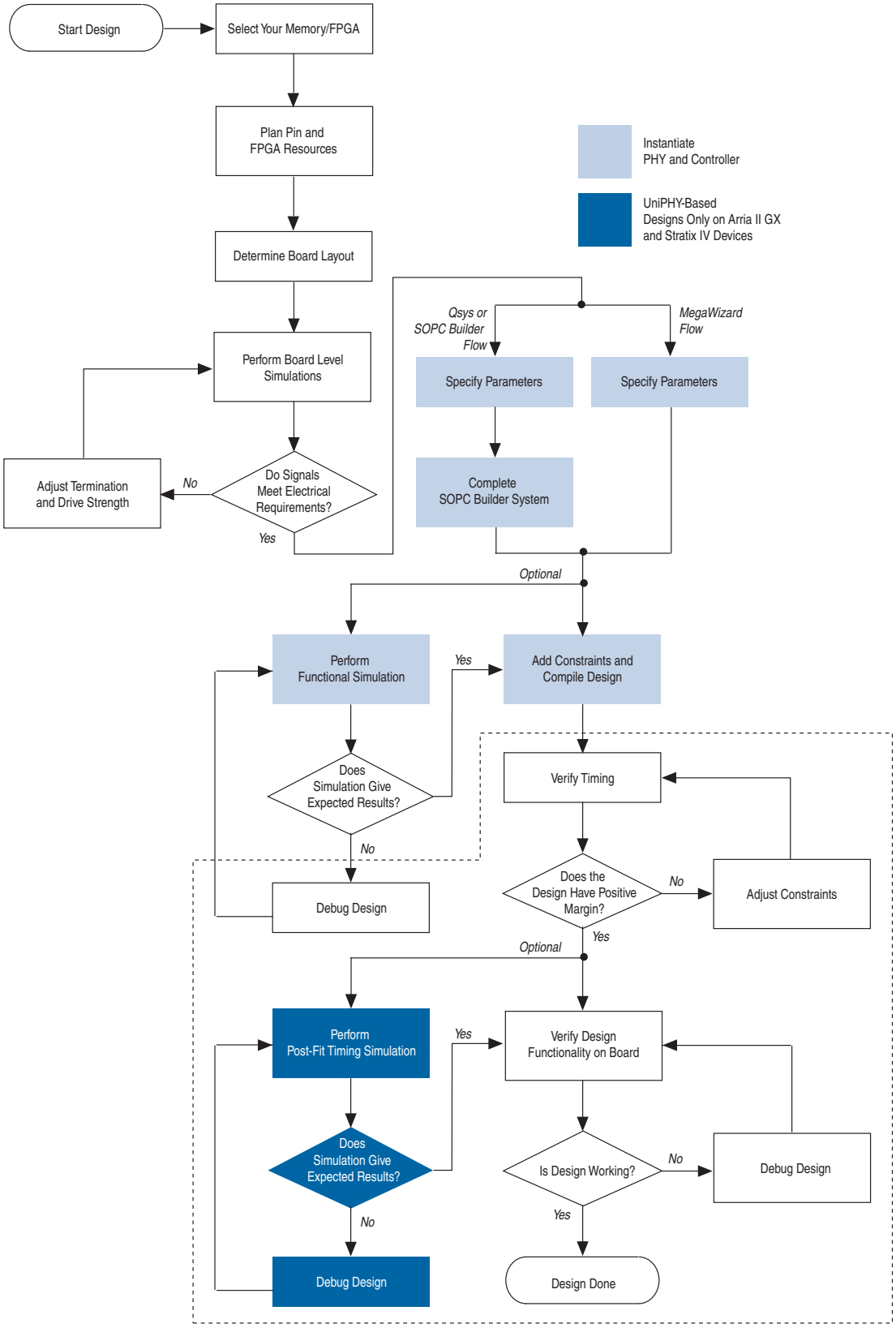
EMI_GS_002-3.0

This chapter describes the Altera-recommended design flow for successfully implementing external memory interfaces in Altera® devices. Altera recommends that you create an example top-level file 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. Use the "Design Checklist" on page 2–3, to verify whether you have performed all the recommended steps in creating a working and robust external memory interface.

## Design Flow

Figure 2–1 shows the design flow to provide the fastest out-of-the-box experience with external memory interfaces in Altera devices. This design flow assumes that you are using Altera IP to implement the external memory interface.

**Figure 2–1. External Memory Interfaces Design Flowchart**



# Design Checklist

Use this design checklist when implementing external memory interfaces in Altera devices.

| | Done | Action | Reference |
|---|---|---|---|
| | | **Select Your Memory** | |
| 1. | ☐ | Select the memory interface frequency of operation and bus width. | ■ *Selecting Your Memory* chapter in the *External Memory Interface Handbook*. |
| | | **Select Your FPGA** | |
| 2. | ☐ | Select the FPGA device density and package combination that you want to target. | ■ *Selecting Your FPGA* chapter in the *External Memory Interface Handbook*. |
| | | **Plan Pin and FPGA Resources** | |
| 3. | ☐ | Ensure that the target FPGA device supports the desired clock rate and memory bus width. Also the FPGA must have sufficient I/O pins for the DQ/DQS read and write groups. | For detailed device resource information, refer to the relevant device handbook chapter on external memory interface support. |
| | | **Determine Board Layout** | |
| 4. | ☐ | Select the termination scheme and drive strength settings for all the memory interface signals on the memory side and the FPGA side. | ■ *DDR2 and DDR3 SDRAM Board Design Guidelines* chapter in the *External Memory Interface Handbook*. |
| 5. | ☐ | Ensure you apply appropriate termination and drive strength settings on all the memory interface signals, and verify using board level simulations. | ■ *Dual-DIMM DDR2 and DDR3 SDRAM Board Design Guidelines* chapter in the *External Memory Interface Handbook*. |
| 6. | ☐ | Use board level simulations to pick the optimal setting for best signal integrity. On the memory side, Altera recommends the use of external parallel termination on input signals to the memory (write data, address, command, and clock signals). | ■ *RLDRAM II Board Design Guidelines* chapter in the *External Memory Interface Handbook*.<br>■ *QDR II SRAM Board Design Guidelines* chapter in the *External Memory Interface Handbook*. |
| 7. | ☐ | Perform board level simulations, to ensure electrical and timing margins for your memory interface | |
| 8. | ☐ | Ensure you have a sufficient eye opening using simulations. Use the latest FPGA and memory IBIS models, board trace characteristics, drive strength, and termination settings in your simulation.<br>Any timing uncertainties at the board level that you calculate using simulations must be used to adjust the input timing constraints to ensure the accuracy of Quartus II timing margin reports. For example crosstalk, ISI, and slew rate deration. | |
| | | **Parameterize and Implement the Memory IP** | |
| 9. | ☐ | Parameterize and instantiate the Altera external memory IP for your target memory interface. | ■ *Implementing and Parameterizing Memory IP* chapter in the *External Memory Interface Handbook* |

| | Done | Action | Reference |
|---|---|---|---|
| 10. | ☐ | Ensure that you perform the following actions:<br>■ Pick the correct memory interface data rates, width, and configurations.<br>■ For DDR, DDR2, and DDR3 SDRAM interfaces, ensure that you derate the tIS, tIH, tDS, and tDH parameters, as necessary.<br>■ Include the board skew parameters for your board. | |
| 11. | ☐ | Connect the PHY's local signals to your driver logic and the PHY's memory interface signals to top-level pins.<br>Ensure that the local interface signals of the PHY are appropriately connected to your own logic. If the ALTMEMPHY IP is compiled without these local interface connections, you may encounter compilation problems, when the number of signals exceeds the pins available on your target device.<br>You may also use the example top-level file as an example on how to connect your own custom controller to the Altera memory PHY. | ■ *Functional Description: HPC II* chapter in the *External Memory Interface Handbook*.<br>■ *Functional Description: QDR II and QDR II+ SRAM Controller* chapter in the *External Memory Interface Handbook*.<br>■ *Functional Description: RLDRAM II Controller* chapter in the *External Memory Interface Handbook*. |
| | | **Perform Functional Simulation** | |
| 12. | ☐ | Simulate your design using the RTL functional model.<br>Use the IP functional simulation model with your own driver logic, testbench, and a memory model, to ensure correct read and write transactions to the memory.<br>You may need to prepare the memory functional model by setting the speed grade and device bus mode. | ■ *Simulating Memory IP* chapter in the *External Memory Interface Handbook* |
| | | **Add Constraints** | |
| 13. | ☐ | Add timing constraints. The wizard-generated **.sdc** file adds timing constraints to the interface. However, you may need to adjust these settings to best fit your memory interface configuration. | |
| 14. | ☐ | Add pin settings and DQ group assignments. The wizard-generated **.tcl** file includes I/O standard and pin loading constraints to your design. | |
| 15. | ☐ | Ensure that generic pin names used in the constraint scripts are modified to match your top-level pin names. The loading on memory interface pins is dependent on your board topology (memory components). | |
| 16. | ☐ | Add pin location assignments. However, you need to assign the pin location assignments manually using the Pin Planner. | |
| 17. | ☐ | Ensure that the example top-level file or your top-level logic is set as top-level entity. | |

| | Done | Action | Reference |
|---|---|---|---|
| 18. | ☐ | Adjust optimization techniques, to ensure the remaining unconstrained paths are routed with the highest speed and efficiency:<br>a. On the Assignments menu click **Settings**.<br>b. Select **Analysis & Synthesis Settings**.<br>c. Select **Speed** under **Optimization Technique**.<br>d. Expand **Fitter Settings**.<br>e. Turn on **Optimize Hold Timing** and select **All Paths**.<br>f. Turn on **Optimize Fast Corner Timing**.<br>g. Select **Standard Fit** under **Fitter Effort**. | |
| 19. | ☐ | Provide board trace delay model. For accurate I/O timing analysis, you specify the board trace and loading information in the Quartus II software. This information should be derived and refined during your board development process of prelayout (line) simulation and finally post-layout (board) simulation. Provide the board trace information for the output and bidirectional pins through the board trace model in the Quartus II software. | |
| | | **Compile Design and Verify Timing** | |
| 20. | ☐ | Compile your design and verify timing closure using all available models. | |
| 21. | ☐ | Run the wizard-generated <*variation_name*>**_report_timing.tcl** file, to generate a custom timing report for each of your IP instances. Run this process across all device timing models (slow 0° C, slow 85° C, fast 0° C). | |
| 22. | ☐ | If there are timing violations, adjust your constraints to optimize timing | |
| 23. | ☐ | As required, adjust PLL clock phase shift settings or appropriate timing and location assignments margins for the various timing paths within the IP. | ■ *Analyzing Timing of Memory IP* chapter in the *External Memory Interface Handbook* |
| | | **Perform Post-Fit Timing Simulation** | |
| 24. | ☐ | Perform post-fit timing simulation to ensure that all the memory transactions meet the timing specifications with the vendor's memory model. | ■ *Simulating Memory IP* chapter in the *External Memory Interface Handbook*. |
| | | **Verify Design Functionality** | |
| 25. | ☐ | Verify the functionality of your memory interface in the system | ■ *Debugging Memory IP* chapter in the *External Memory Interface Handbook* |

# Document Revision History

Table 2–1 shows the revision history for this document.

**Table 2–1. Document Revision History**

| Date | Version | Changes |
|---|---|---|
| June 2012 | 3.0 | ■ Removed overlapping information.<br>■ Addd Feedback icon. |
| November 2011 | 2.1 | Updated the design flow and the design checklist. |
| July 2010 | 2.0 | Updated for 10.0 release. |
| January 2010 | 1.1 | ■ Improved description for *Implementing Altera Memory Interface IP* chapter.<br>■ Added timing simulation to flow chart and to design checklist. |
| November 2009 | 1.0 | First published. |