External Memory Interface Handbook Volume 1: Intel® FPGA Memory Solution Introduction and Design Flow

---

For UniPHY-based Device Families

Updated for Intel® Quartus® Prime Design Suite: **17.0**

Contents

1. Introduction to this Handbook

The *External Memory Interface Handbook* describes the UniPHY-based external memory interface IP available for use with Intel®'s V-series and earlier devices using UniPHY-based IP.

1.1. Introduction to Memory Solutions

Intel FPGAs achieve optimal memory interface performance with external memory IP. Intel provides the fastest, most efficient, and lowest latency memory interface IP cores, designed to easily interface with today's higher speed memory devices.

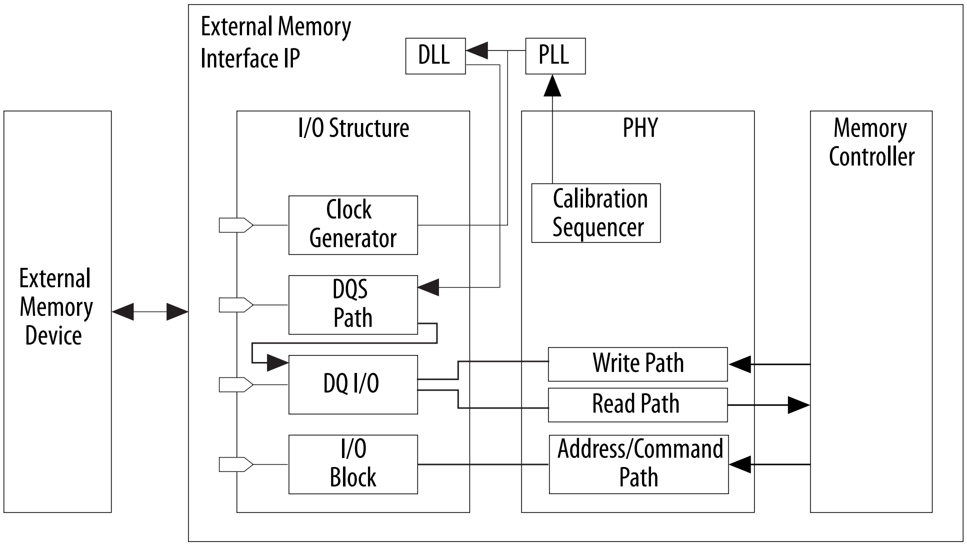
Intel supports a wide variety of memory interfaces suitable for applications ranging from routers and switches to video cameras. You can easily implement Intel’s intellectual property (IP) using the memory IP core functions through the Intel Quartus® Prime software. The Intel Quartus Prime software also provides external memory toolkits that help 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 Intel FPGAs.

The external memory interface IP provides the following components:

- Physical layer interface (PHY) which builds the data path and manages timing transfers between the FPGA and the memory device.
- Memory controller which implements all the memory commands and protocol-level requirements.
- Multi-port front end (MPFE) which allows multiple components inside the FPGA device to share a common memory interface. The MPFE is available in Arria V and Cyclone V devices.

Figure 1. Memory Interface Architecture



Intel's FPGAs provide two types of memory solutions, depending on device family: soft memory IP and hard 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.

The following table lists features of the soft and hard memory IP.

Table 1. Features of the Soft and Hard Memory IP

Soft Memory IP	Hard Memory IP
• Includes hardened PHY with soft controller. • Allows maximum flexibility in choosing location, size, and configuration of the memory interface. • Can optionally be used in PHY-only mode to integrate with a custom user-designed controller.	• Includes hardened PHY, hardened controller, and hardened MPFE. • Supports maximum performance and lowest latency. • May have a fixed location on a device and/or a fixed pinout for address and command signals. • Simplifies the overall integration of a memory interface and provides an out-of-the-box experience for every designer.

Intel provides modular memory solutions that allow you to customize your memory interface design to a variety of configurations:

- PHY with your own controller
- PHY with Intel controller
- PHY with Intel controller and a multiport front end. (MPFE is a configurable block available for hard interfaces in Arria V and Cyclone V devices.)

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

Related Information

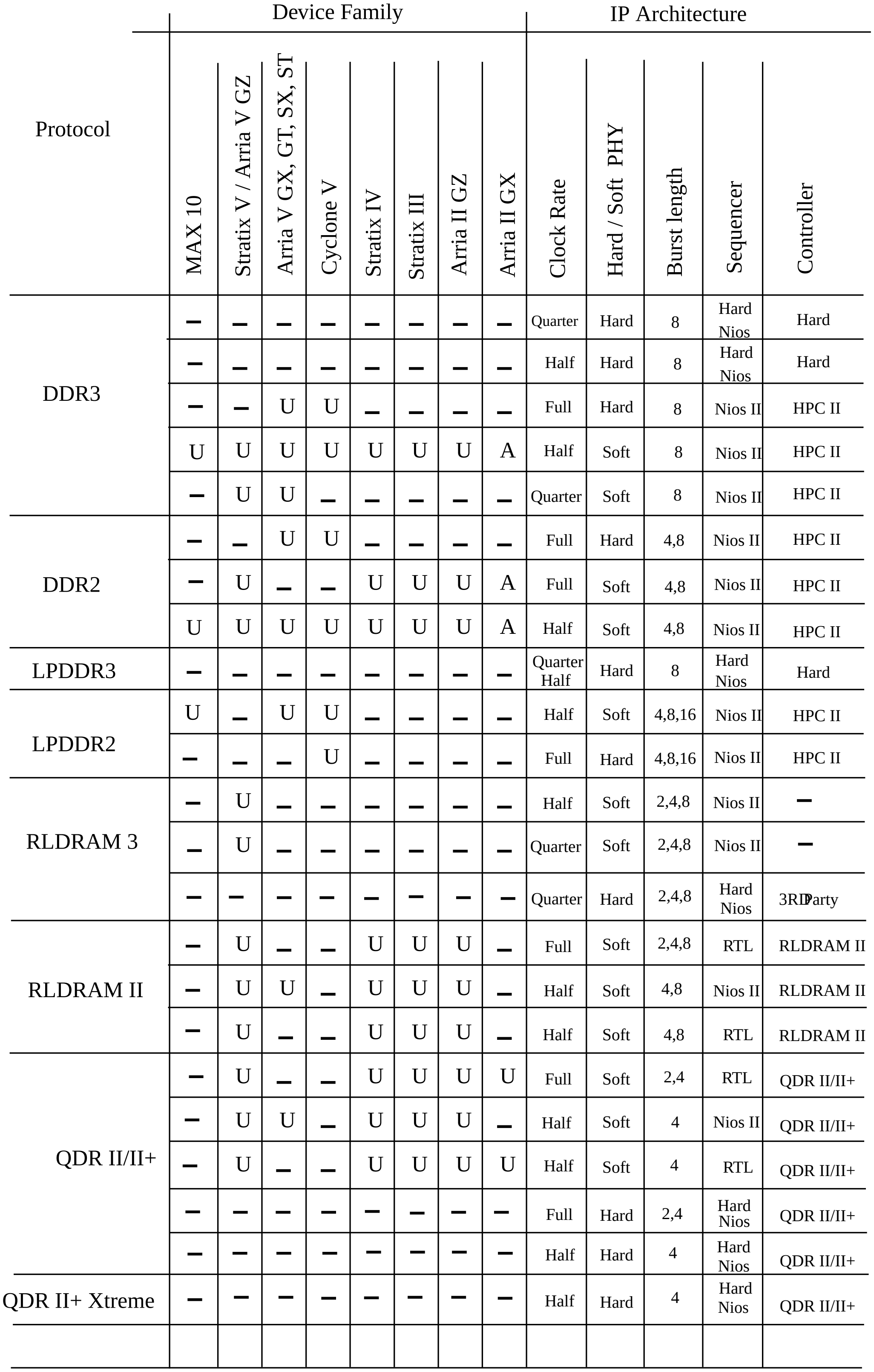
- ALTDLL and ALTDQ_DQS Megafunctions User Guide
- ALTDQ_DQS2 IP User Guide
- PHY Lite for Parallel Interfaces Intel FPGA IP User Guide

- Functional Description: MAX 10 EMIF IP
- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- Platform Designer (Standard) Simulation Scripts
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

1.2. Protocol Support Matrix

The following table lists the device family, memory protocol, and IP architecture support for the UniPHY-based external memory interface IP in the current release of the Intel Quartus Prime Design Suite.

Figure 2. Protocol Support Matrix (1) (2) (3) (4) (5)



	Device Family								IP Architecture				
Protocol	MAX 10	Stratix V / Arria V GZ	Arria V GX, GT, SX, ST	Cyclone V	Stratix IV	Stratix III	Arria II GZ	Arria II GX	Clock Rate	Hard / Soft PHY	Burst length	Sequencer	Controller
DDR3	–	–	–	–	–	–	–	–	Quarter	Hard	8	Hard Nios	Hard
	–	–	–	–	–	–	–	–	Half	Hard	8	Hard Nios	Hard
	–	–	U	U	–	–	–	–	Full	Hard	8	Nios II	HPC II
	U	U	U	U	U	U	U	A	Half	Soft	8	Nios II	HPC II
	–	U	U	–	–	–	–	–	Quarter	Soft	8	Nios II	HPC II
DDR2	–	–	U	U	–	–	–	–	Full	Hard	4,8	Nios II	HPC II
	–	U	–	–	U	U	U	A	Full	Soft	4,8	Nios II	HPC II
	U	U	U	U	U	U	U	A	Half	Soft	4,8	Nios II	HPC II
LPDDR3	–	–	–	–	–	–	–	–	Quarter Half	Hard	8	Hard Nios	Hard
LPDDR2	U	–	U	U	–	–	–	–	Half	Soft	4,8,16	Nios II	HPC II
	–	–	–	U	–	–	–	–	Full	Hard	4,8,16	Nios II	HPC II
RLDRAM 3	–	U	–	–	–	–	–	–	Half	Soft	2,4,8	Nios II	–
	–	U	–	–	–	–	–	–	Quarter	Soft	2,4,8	Nios II	–
	–	–	–	–	–	–	–	–	Quarter	Hard	2,4,8	Hard Nios	3RD Party
RLDRAM II	–	U	–	–	U	U	U	–	Full	Soft	2,4,8	RTL	RLDRAM II
	–	U	U	–	U	U	U	–	Half	Soft	4,8	Nios II	RLDRAM II
	–	U	–	–	U	U	U	–	Half	Soft	4,8	RTL	RLDRAM II
QDR II/II+	–	U	–	–	U	U	U	U	Full	Soft	2,4	RTL	QDR II/II+
	–	U	U	–	U	U	U	–	Half	Soft	4	Nios II	QDR II/II+
	–	U	–	–	U	U	U	U	Half	Soft	4	RTL	QDR II/II+
	–	–	–	–	–	–	–	–	Full	Hard	2,4	Hard Nios	QDR II/II+
	–	–	–	–	–	–	–	–	Half	Hard	4	Hard Nios	QDR II/II+
QDR II+ Xtreme	–	–	–	–	–	–	–	–	Half	Hard	4	Hard Nios	QDR II/II+

Notes to Table:
1. **U**= Supported by UniPHY-based IP.
2. **A** = Supported by ALTMEMPHY-based IP. Refer to the *External Memory Interface Handbook* for the Quartus II software version 12.1 or earlier for information about ALTMEMPHY-based IP.
3. **—** = Not supported.
4. The RTL-based sequencer is not available for QDR II or RLDRAM II interfaces targeting Arria V devices.

For more information about the controllers with the Intel UniPHY IP, refer to the *Functional Descriptions* section in Volume 3 of the *External Memory Interface Handbook*.

For more information on the Intel MAX® 10 external memory interface IP, see *Functional Description—MAX 10 EMIF IP*.

Related Information

- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- Platform Designer (Standard) Simulation Scripts
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

1.3. Document Revision History

Date	Version	Changes
March 2023	2023.03.06	• Added subtitle to volume title. • Replaced *Introduction to Memory Solutions* topic with *Introduction to this Handbook* topic. • Added note and links to *Introduction to this Handbook* topic. • Removed Intel Arria 10 and Intel Stratix 10 references from the *Protocol Support Matrix* topic. • Removed *Arria 10 EMIF Future Protocol Support* topic.
May 2017	2017.05.08	• Rebranded as Intel. • Added Stratix 10 support statement to notes at the bottom of the *Protocol Support Matrix*.
October 2016	2016.10.31	Maintenance release.
May 2016	2016.05.02	Maintenance release.
November 2015	2015.11.02	• Changed instances of *Quartus II* to *Quartus Prime*. • Changed Arria 10 EMIF current support for LPDDR3 to yes. • Added LPDDR3 to product support matrix.
May 2015	2015.05.04	Maintenance release.
December 2014	2014.12.15	• Added QDR IV and MAX 10 support to the *Protocol Support Matrix*.
August 2014	2014.08.15	• Added information for Quartus II software versions 14.0 and 14.0 Arria 10 Edition to *Altera Memory Types, PHY, and Controllers in the Quartus II Software* table. • Added QDR II, QDR II+, and QDR II+ Xtreme support for Arria 10 to the *Protocol Support Matrix*. • Updated DDR3, DDR4, QDR II+ / QDR II+ Extreme, and RLDRAM 3 support in the *Arria 10 EMIF Future Protocol Support* table.

continued...

Date	Version	Changes
December 2013	2013.12.16	• Added Arria 10 and DDR4 information to *Protocol Support Matrix* and *Memory Solutions*. • Combined *Soft and Hard Memory IP* and *Memory Solutions* sections. • Removed HardCopy III/IV from *Protocol Support Matrix*. • Added note to *Protocol Support Matrix* that RTL-based sequencer is not available for QDR II or RLDRAM II interfaces targeting Arria V devices
November 2012	2.0	• Added Arria V GZ information. • 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*. • Added Feedback icon.
November 2011	1.0	Initial release.

Related Information

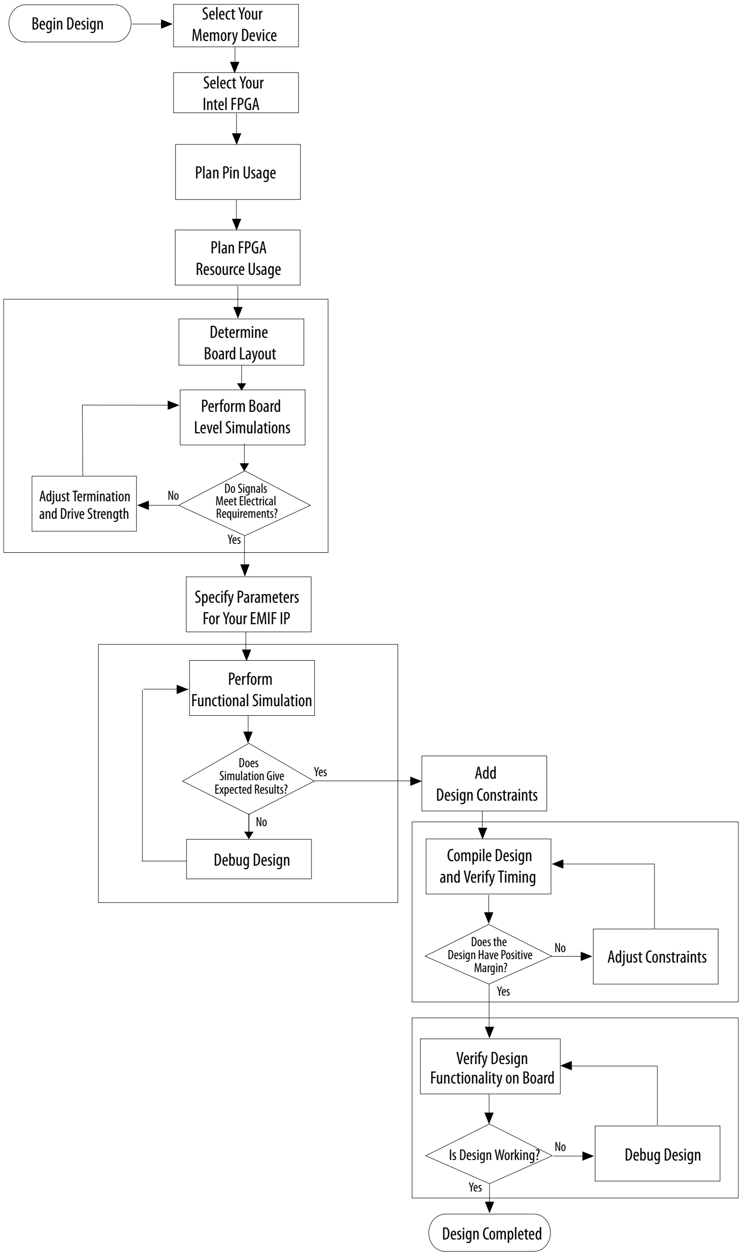
- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.
- Platform Designer (Standard) Simulation Scripts
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

2. Recommended Design Flow

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

The following figure shows the design flow to provide the fastest out-of-the-box experience with external memory interfaces in Intel FPGAs. This design flow assumes that you are using Intel IP to implement the external memory interface.

Figure 3. External Memory Interfaces Design Flowchart



Refer to *Getting Started with External Memory Interfaces* for guidance in performing the recommended steps in creating a working and robust external memory interface.

Related Information

- Getting Started With External Memory Interfaces on page 11
- Introduction to Intel FPGA IP Cores
  Provides general information about all Intel FPGA IP cores, including parameterizing, generating, upgrading, and simulating IP cores.

- Creating Version-Independent IP and Qsys Simulation Scripts
  Create simulation scripts that do not require manual updates for software or IP version upgrades.
- Project Management Best Practices
  Guidelines for efficient management and portability of your project and IP files.

2.1. Getting Started With External Memory Interfaces

To create your external memory interface, you must complete several high-level tasks. This topic outlines the major tasks in the design flow, and provides links to detailed procedures for each task.

Refer to this section for a big-picture view of the overall design process, and for links to related information for each task.

The High-Level Tasks

1. Selecting Your External Memory Device on page 11
2. Selecting Your FPGA on page 12
3. Planning Your Pin Requirements on page 13
4. Planning Your FPGA Resources on page 13
5. Determining Your Board Layout on page 14
6. Specifying Parameters for Your External Memory Interface on page 14
7. Performing Functional Simulation on page 15
8. Adding Design Constraints on page 15
9. Compiling Your Design and Verifying Timing on page 16
10. Verifying and Debugging External Memory Interface Operation on page 16

2.1.1. Selecting Your External Memory Device

Different memory types excel in different areas. As a first step in planning your external memory interface, you must determine the memory type that best meets the requirements of your system.

1. Determine your requirements for the following:
   - bandwidth
   - speed
   - data capacity
   - latency
   - power consumption
2. Compare your requirements to the specifications for available memory protocols to find the memory device appropriate for your application.

Related Information

Selecting Your Memory

2.1.2. Selecting Your FPGA

Different Intel FPGA devices support different memory types; not all Intel devices support all memory protocols and configurations. Before you start your design, you must select an Intel device, which supports the memory standard and configurations you plan to use.

1. Determine the I/O interface that best suits your design requirements.
2. Determine whether your design requires read or write levelling circuitry.

   Some Intel FPGAs support read and write levelling, to apply or remove skew from an interface on a DQS group basis.
3. Determine whether your design requires dynamic calibrated on-chip termination (OCT).

   Some Intel FPGAs provide dynamic OCT, allowing a specified series termination to be enabled during writes and parallel termination to be enabled during reads. Dynamic OCT can simplify your PCB termination schemes.
4. Consult the Intel FPGA Product Selector to find the Intel FPGA that provides the combination of features that your design requires.
5. Refer to the Ordering Information section of the appropriate device handbook, to determine the correct ordering code for the device that you require. Consider the following characteristics in determining the correct ordering code:
   - Speed grade: Affects performance, timing closure, and power consumption. The device with the smallest speed grade number is the fastest device.
   - Operating temperature: Intel FPGAs are divided into the following temperature categories:
     - Commercial grade—Used for all device families. Operating temperature ranges from 0 degreec C to 85 degrees C.
     - Industrial grade—Used for all device families. Operating temperature ranges from -40 degreec C to 100 degrees C.
     - Military grade—Used for Stratix IV device families. Operating temperature ranges from -55 degree C to 125 degrees C.
     - Automotive grade—Used for Cyclone V device families. Operating temperature ranges from -40 degreec C to 125 degrees C.
   - Package size: Refers to the physical size of the FPGA device, and corresponds to the number of pins. For example, the package size for the smallest Stratix IV device is 29 mm x 29 mm, categorized under the F780 package option, where F780 refers to a device with 780 pins.
   - Device density: Refers to the number of logic elements, such as PLLs and memory blocks. Devices with higher density contain more logic elements in less area.
   - I/O pin counts: The number of I/O pins required on an FPGA depends on the memory standard, the number of memory interfaces, and the memory data width.

Tip: For additional, device-specific, information, refer to the External Memory Interface chapter in the device handbook for your Intel device.

Related Information

- Selecting Your FPGA Device
- Intel FPGA Device Selector

2.1.3. Planning Your Pin Requirements

Before you can specify parameters for your external memory interface, you must determine the pin requirements. You should use the Intel Quartus Prime software for final pin fitting; however, you can estimate whether you have enough pins for your memory interface.

1. Determine how many read data pins are associated per read data strobe or clock pair.
2. Check the device density and packaging information for your FPGA to determine whether you can implement your interface in one I/O bank, or on one side of the device, or on two adjacent sides.
3. Calculate the number of other memory interface pins needed, including any other clocks (write clock or memory system clock), address, command, RUP, RDN, RZQ, and any other pins to be connected to the memory components. Ensure you have enough pins to implement the interface in one I/O bank or one side or on two adjacent sides.
4. Apply the General Pin-Out Guidelines, and observe any device- or protocol-specific guidelines or exceptions applicable to your design situation.

Related Information

- Planning Pin and FPGA Resources
- External Memory Interface Spec Estimator

2.1.4. Planning Your FPGA Resources

Before you can specify parameters for your external memory interface, you must determine the FPGA resource requirements. The FPGA resources required by your design depend on many factors, including the memory interface frequency, timing requirements, and the IP that your design uses.

1. Determine the PLLs and clock networks that your design requires.
2. If multiple PLLs are required for multiple controllers that cannot be shared, ensure that enough PLL resources are available within each quadrant to support your interface number requirements.
3. Determine whether cascading of PLLs is appropriate for your design.
4. Determine the appropriate DLL usage for your design. If multiple external memory interfaces must share DLL resources, ensure that the frequency and mode requirements are compatible.
5. Determine the registers, memory blocks, OCT blocks, and other FPGA resources required by your design.

Related Information

- Planning Pin and FPGA Resources
- External Memory Interface Spec Estimator

2.1.5. Determining Your Board Layout

Before you can specify parameters for your external memory interface, you must determine the necessary board-related settings for your IP.

1. Review the recommended board design guidelines for your external memory interface protocol.
2. Select the termination scheme and drive strength settings for all the memory interface signals connected between the FPGA and the external memory device.
3. Perform board-level simulations to determine the optimal settings for best signal integrity, appropriate timing margins, and sufficient eye opening.
    - Successful board-level simulation is often an iterative process, experimenting with different combinations of drive strength, terminations, IP board parameters, and timing results.
    - Ensure that your simulation applies the latest FPGA and memory device IBIS models, board trace characteristics, drive strength, and termination settings.
    - You might identify board-level timing uncertainties such as crosstalk, ISI, or slew rate deration during simulation. If you identify such timing uncertainties, adjust the Board Settings in the IP Catalog with the slew rate deration, ISI/crosstalk, and board skews to ensure the accuracy of the timing margins report.

Related Information

- DDR2 and DDR3 SDRAM Board Design Guidelines
- Dual-DIMM DDR2 and DDR3 SDRAM Board Design Guidelines
- LPDDR2 SDRAM Board Design Guidelines
- QDR II/II+ SRAM Board Design Guidelines
- RLDRAM II and RLDRAM 3 Board Design Guidelines

2.1.6. Specifying Parameters for Your External Memory Interface

After you have determined all the necessary requirements, you can parameterize your external memory interface.

1. In the parameter editor, set the parameters for the external memory IP for your target memory interface.
    - Refer to *Specifying IP Core Parameters and Options* for information about using the IP Catalog and parameter editor.
    - Refer to *Implementing and Parameterizing Memory IP* for detailed information about parameterizing external memory interface IP.
2. Specify the correct parameters for each of the following:
    - Memory interface data rate, width, and configuration.
    - Necessary deratings for tIS, tIH, tDH, and tDS parameters, as appropriate.
    - Board skew parameters based on actual board simulation.
3. Connect the local signals from the PHY and controller to your driver logic, and the memory interface signals from the PHY to the top-level pins.

- It is important that you connect the local interface signals from the PHY or controller correctly to your own logic. If you do not connect these local interface signals, you might encounter problems with insufficient pins when you compile your design.
- Logic that is not connected may be optimized away during compilation, resulting in problems later.
- If you want to use your own custom memory controller with the Intel PHY, you can refer to the example top-level file as an example for connecting your controller.

Related Information

- Implementing and Parameterizing Memory IP
- Functional Description—HPC II Controller
- Functional Description–Hard Memory Interface
- Functional Description—QDR II Controller
- Functional Description—RLDRAM II Controller
- Functional Description—RLDRAM 3 PHY-Only IP
- Functional Description—MAX 10 EMIF

2.1.7. Performing Functional Simulation

Simulate your design to determine correct operation, timing closure, and overall latency.

1. Simulate your design using the RTL functional model.
2. 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.
3. You may need to prepare the memory functional model by setting the speed grade and device bus mode.

Related Information

Simulating Memory IP

2.1.8. Adding Design Constraints

Design constraints establish the timing characteristics of your IP and the physical locations of I/O and routing resources.

1. Add timing constraints.
2. Add pin assignments.
3. Add pin location assignments.
4. Ensure that the example top-level file or your top-level logic is set as top-level entity.
5. Adjust optimization techniques, to ensure the remaining unconstrained paths are routed with the highest speed and efficiency, as follows:
    a. In the Intel Quartus Prime software, click **Assignments ➤ Settings**.
    b. In the **Settings** dialog box, select the **Compiler Settings** category.

c. In the **Compiler Settings** dialog box, click **Advanced Settings (Synthesis)** and set the **Optimization Technique** value to **Speed**.

d. In the **Compiler Settings** dialog box, click **Advanced Settings (Fitter)** and set **Optimize hold timing** to **All Paths**. Turn on **Optimize multi-corner timing**. Set **Fitter Effort** to **Standard Fit**.

Related Information

Analyzing Timing of Memory IP

2.1.9. Compiling Your Design and Verifying Timing

When you compile your design, the Timing Analyzer generates timing reports for your design.

1. Compile your design by clicking **Processing ➤ Start Compilation**.
   Memory timing scripts run automatically as part of `Report DDR`.
2. Verify timing closure using all available models, and evaluate the timing reports generated by the Timing Analyzer.

   As required, adjust the constraints described in Adding Design Constraints to resolve timing or location issues.
3. Iteratively recompile your IP and evaluate the timing results as necessary to achieve the required timing margins.

Related Information

- Analyzing Timing of Memory IP
- Implementing and Parameterizing Memory IP

2.1.10. Verifying and Debugging External Memory Interface Operation

Operational problems can generally be attributed to one of the following: resource and planning problems, interface configuration problems, functional problems, signal integrity problems, or timing problems.

- Refer to *Debugging Memory IP* and the *External Memory Interface Debug Toolkit* for information on resolving operational problems.

Related Information

- Debugging Memory IP
- External Memory Interface Debug Toolkit

2.2. Document Revision History

Date	Version	Changes
May 2017	2017.05.08	Rebranded as Intel.
October 2016	2016.10.31	Maintenance release.
May 2016	2016.05.02	Maintenance release.
November 2015	2015.11.02	Changed instances of *Quartus II* to *Quartus Prime*.

continued...

Date	Version	Changes
May 2015	2015.05.04	Maintenance release.
December 2014	2014.12.15	• Revised the *External Memory Interfaces Design Flowchart*. • Removed the *Design Checklist* and added *Getting Started With External Memory Interfaces*, and associated subtopics.
August 2014	2014.08.15	Removed MegaWizard Plug-In Manager flow and added IP Catalog Flow to *External Memory Interfaces Design Flowchart*.
December 2013	2013.12.16	• Removed references to ALTMEMPHY. • Removed references to SOPC Builder. • Removed ALTMEMPHY-related step from design checklist.
June 2012	2013.12.02	• Removed overlapping information. • Added 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. • Added timing simulation to flow chart and to design checklist.
November 2009	1.0	Initial release.

3. Selecting Your Memory

System architects must consider architecture, algorithms, and features of the available components.

Typically, one of the fundamental problems in high-performance applications is memory, because the challenges and limitations of system performance often reside in memory architecture. As higher speeds become necessary for external memories, signal integrity becomes more challenging; newer devices include several features to address this challenge. Intel FPGAs include dedicated I/O circuitry, various I/O standard support, and specialized intellectual property (IP).

When you select an external memory device, consider the following factors:

- Bandwidth and speed
- Cost
- Data storage capacity
- Latency
- Power consumption

Because no single memory type can excel in every area, system architects must determine the right balance for their design. The following table lists the two common types of high-speed memories and their characteristics.

Table 2. Differences between DRAM and SRAM

Memory Type	Description	Bandwidth and Speed	Cost	Data Storage Size and Capacity	Power consumption	Latency
DRAM	A dynamic random access memory (DRAM) cell consisting of a capacitor and a single transistor. DRAM memory must be refreshed periodically to retain the data, resulting in lower overall efficiency and more complex controllers. Generally, designers select DRAM where cost per bit and capacity are important. DRAM is commonly used for main memory.	Lower bandwidth resulting in slower speed	Lower cost	Higher data storage and capacity	Higher power consumption	Higher latency
SRAM	A static random access memory (SRAM) cell that consists of six transistors. SRAM does not need to be refreshed because the transistors continue to hold the data as long as the power supply is not cut off. Generally, designers select SRAM where speed is more important than capacity. SRAM is commonly used for cache memory.	Higher bandwidth resulting in faster speed	Higher cost	Lower data storage and capacity	Lower power consumption	Lower latency

Note: The Intel FPGA IP might or might not support all of the features supported by the memory.

To compare the performance of the supported external memory interfaces in Intel FPGA devices, refer to the *External Memory Interface Spec Estimator* on the Intel website.

Related Information

External Memory Interface Spec Estimator

3.1. DDR SDRAM Features

Double data rate (DDR) SDRAM is a 2n prefetch architecture with two data transfers per clock cycle. It uses a single-ended strobe, `DQS`, which is associated with a group of data pins, `DQ`, for read and write operations. Both `DQS` and `DQ` are bidirectional ports. Address ports are shared for read and write operations.

The desktop computing market has positioned DDR SDRAM as a mainstream commodity product, which means this memory is very low-cost. DDR SDRAM is also high-density and low-power. Relative to other high-speed memories, DDR SDRAM has higher latency-they have a multiplexed address bus, which reduces the pin count (minimizing cost) at the expense of a longer and more complex bus cycle.

3.2. DDR2 SDRAM Features

DDR2 SDRAM is a 4n prefetch architecture (internally the memory operates at half the interface frequency) with two data transfers per clock cycle. DDR2 SDRAM can use a single-ended or differential strobe, `DQS` or `DQSn`, which is associated with a group of data pins, and `DQ` for read and write operations. The `DQS`, `DQSn`, and `DQ` are bidirectional ports. Address ports are shared for read and write operations.

DDR2 SDRAM includes additional features such as increased bandwidth due to higher clock speeds, improved signal integrity on DIMMs with on-die terminations, and lower supply voltages to reduce power.

3.3. DDR3 SDRAM Features

DDR3 SDRAM is the third generation of SDRAM. DDR3 SDRAM is internally configured as an eight-bank DRAM and 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, one-clock-cycle data transfer at the internal DRAM core and eight corresponding n-bit wide, one-half clock cycle data transfers at the I/O pins. DDR3 SDRAMs are available as components and modules, such as DIMMs, SODIMMs, RDIMMs, and LRDIMMs.

DDR3 SDRAM can conserve system power, increase system performance, achieve better maximum throughput, and improve signal integrity with fly-by topology and dynamic on-die termination.

Read and write operations to the DDR3 SDRAM are burst oriented. Operation begins with the registration of an active command, which is 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, as is `DQ` for read and write operations. `DQS`, `DQSn`, and `DQ` ports are bidirectional. Address ports are shared for read and write operations.

Note: The DDR3 SDRAM high-performance controller II supports local interfaces running at full-rate, half-rate, and quarter-rate.

For more information, refer to the respective DDR, DDR2, and DDR3 SDRAM data sheets.

For more information about parameterizing the DDR2 and DDR3 SDRAM IP, refer to the *Implementing and Parameterizing Memory IP* chapter.

Related Information

Implementing and Parameterizing Memory IP

3.4. QDR, QDR II, and QDR II+ SRAM Features

Quad Data Rate (QDR) SRAM has independent read and write ports that run concurrently at double data rate. QDR SRAM is true dual-port (although the address bus is still shared), which gives this memory a high bandwidth, allowing back-to-back transactions without the contention issues that can occur when using a single bidirectional data bus. Write and read operations share address ports.

The QDR II SRAM devices are available in ×8, ×9, ×18, and ×36 data bus width configurations. The QDR II+ SRAM devices are available in ×9, ×18, and ×36 data bus width configurations. Write and read operations are burst-oriented. All the data bus width configurations of QDR II SRAM support burst lengths of two and four. QDR II+ SRAM supports only a burst length of four. Burst-of-two and burst-of-four for QDR II and burst-of-four for QDR II+ SRAM devices provide the same overall bandwidth at a given clock speed.

For QDR II SRAM devices, the read latency is 1.5 clock cycles; for QDR II+ SRAM devices, it is 2 or 2.5 clock cycles depending on the memory device. For QDR II+ and burst-of-four QDR II SRAM devices, the write commands and addresses are clocked on the rising edge of the clock, and write latency is one clock cycle. For burst-of-two QDR II SRAM devices, the write command is clocked on the rising edge of the clock, and the write address is clocked on the falling edge of the clock. Therefore, the write latency is zero because the write data is presented at the same time as the write command.

QDR II+ and QDR II SRAM interfaces use a delay-locked loop (DLL) inside the device to edge-align the data with respect to the K and K# or C and C# pins. You can optionally turn off the DLL, but the performance of the QDR II+ and QDR II SRAM devices is degraded. All timing specifications listed in this document assume that the DLL is on. QDR II+ and QDR II SRAM devices also offer programmable impedance output buffers. You can set the buffers by terminating the ZQ pin to VSS through a resistor, RQ. The value of RQ should be five times the desired output impedance. The range for RQ should be between 175 ohm and 350 ohm with a tolerance of 10%.

QDR II/+ SRAM is best suited for applications where the required read/write ratio is near one-to-one. QDR II/+ SRAM includes additional features such as increased bandwidth due to higher clock speeds, lower voltages to reduce power, and on-die termination to improve signal integrity. QDR II+ SDRAM is the latest and fastest generation. For QDR II+ and QDR II SRAM interfaces, Intel supports both 1.5-V and 1.8-V HSTL I/O standards.

For more information, refer to the respective QDRII and QDRII+ data sheets.

For more information about parameterizing the QDRII and QDRII+ SRAM IP, refer to the *Implementing and Parameterizing Memory IP* chapter.

Related Information

Implementing and Parameterizing Memory IP

3.5. RLDRAM II and RLDRAM 3 Features

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

The high performance of RLDRAM is achieved by very low random access delay (tRC), low data-bus-turnaround delay, simple command protocol, and a large number of banks. RLDRAM is optimized to meet the needs of high-bandwidth networking applications.

Contrasting with the typical four banks in most memory devices, RLDRAM II is partitioned into eight banks and RLDRAM 3 is partitioned into sixteen banks. Partitioning reduces the parasitic capacitance of the address and data lines, allowing faster accesses and reducing the probability of random access conflicts. Each bank has a fixed number of rows and columns. Only one row per bank is accessed at a time. The memory (instead of the controller) controls the opening and closing of a row, which is similar to an SRAM interface.

Most DRAM memory types need both a row and column phase on a multiplexed address bus to support full random access, while RLDRAM supports a nonmultiplexed address, saving bus cycles at the expense of more pins. RLDRAM II and RLDRAM 3 use the High-Speed Transceiver Logic (HSTL) standard with double data rate (DDR) data transfer to provide a very high throughput.

There are two types of RLDRAM II or RLDRAM 3 devices—common I/O (CIO) and separate I/O (SIO). CIO devices share a single data I/O bus, which is similar to the double data rate (DDR) SDRAM interface. SIO devices, with separate data read and write buses, have an interface similar to SRAM. Intel UniPHY Memory IP only supports CIO RLDRAM.

RLDRAM II and RLDRAM 3 use a DDR scheme, performing two data transfers per clock cycle. RLDRAM II or RLDRAM 3 CIO devices use the bidirectional data pins (DQ) for both read and write data, while RLDRAM II or RLDRAM 3 SIO devices use D pins for write data (input to the memory) and Q pins for read data (output from the memory). Both types use two pairs of unidirectional free-running clocks. The memory uses DK and DK# pins during write operations, and generates QK and QK# pins during read operations. In addition, RLDRAM II and RLDRAM 3 use the system clocks (CK and CK# pins) to sample commands and addresses, and to generate the QK and QK# read clocks. Address ports are shared for write and read operations.

RLDRAM II CIO devices are available in ×9, ×18, ×36 data bus width configurations. RLDRAM II CIO interfaces may require an extra cycle for bus turnaround time for switching read and write operations. RLDRAM 3 devices are available in ×18 and ×36 data bus width configurations.

Write and read operations are burst oriented, and all the data bus width configurations of RLDRAM II and RLDRAM 3 support burst lengths of two and four. RLDRAM 3 also supports burst length of eight at bus width ×18, and burst lengths of two and four at bus width ×36. For detailed comparisons between RLDRAM II and RLDRAM 3 for these features, refer to the *Memory Selection Overview* table.

RLDRAM II and RLDRAM 3 also inherently include the additional memory bits used for parity or error correction code (ECC).

RLDRAM II and RLDRAM 3 also offer programmable impedance output buffers and on-die termination. The programmable impedance output buffers are for impedance matching and are guaranteed to produce 25- to 60-ohm output impedance. The on-die termination is dynamically switched on during read operations and switched off during write operations. Perform an IBIS simulation to observe the effects of this dynamic termination on your system. IBIS simulation can also show the effects of different drive strengths, termination resistors, and capacitive loads on your system.

RLDRAM 3 enables a faster, more efficient transfer of data by doubling performance and reduced latency compared to RLDRAM II. RLDRAM 3 memory is suitable for operation in which high bandwidth and deterministic performance is critical, and is optimized to meet the needs of high-bandwidth networking applications. For detailed comparisons between RLDRAM II and RLDRAM 3, refer to the following table.

For more information, refer to RLDRAM II and RLDRAM 3 data sheets available from the Micron website (www.micron.com).

For more information about parameterizing the RLDRAM II and RLDRAM 3 IP, refer to the *Implementing and Parameterizing Memory IP* chapter.

Related Information

- Implementing and Parameterizing Memory IP
- www.micron.com

3.6. LPDDR2 Features

LPDDR2-S is a high-speed SDRAM device internally configured as a 4- or 8-bank memory. All LPDDR2 devices use double data rate architecture on the address and command bus to reduce the number of input pins in the system. The 10-bit address and command bus contains command, address, and bank/row buffer information. Each command uses one clock cycle, during which command information is transferred on both the positive and negative edges of the clock.

LPDDR2-S2 and LPDDR2-S4 devices use double data rate architecture on the `DQ` pins to achieve high speed operation. The double data rate architecture is essentially a 2n/4n prefetch architecture with an interface designed to transfer two data bits per `DQ` every clock cycle at the I/O pins. A single read or write access for the LPDDR2-S2/S4 consists of a single 2n-bit wide /4n-bit wide, one clock cycle data transfer at the internal SDRAM core, and two/four corresponding n-bit wide, with one-half clock cycle data transfers at the I/O pins.

3.7. Memory Selection

One of the first considerations in choosing a high-speed memory is data bandwidth. Based on the system requirements, an approximate data rate to the external memory should be determined. You must also consider other memory attributes, including how much memory is required (density), how much latency can be tolerated, what is the power budget, and whether the system is cost sensitive.

The following table lists memory features and target markets of each technology.

Table 3. Memory Selection Overview

Parameter	LPDDR2	DDR3 SDRAM	DDR2 SDRAM	DDR SDRAM	RLDRAM II	RLDRAM 3	QDR II/+ SRAM
Bandwidth for 32 bit interface *(1)*	25.6	59.7	25.6	12.8	25.6	35.8	44.8
Bandwidth at % Efficiency (Gbps) *(2)*	17.9	41.7	17.9	9	17.9	–	38.1
Performance / Clock frequency	167–400 MHz *(3)*	300–933 MHz	167–400 MHz *(3)*	100–200 MHz	200–533 MHz	200–800 MHz	154–350 MHz
Intel-supported data rate	Up to 1,066 Mbps	Up to 2,133 Mbps	Up to 1,066 Mbps	Up to 400 Mbps	Up to 1066 Mbps	Up to 1600 Mbps	Up to 1400 Mbps
Density	64 MB –8 GB	512 MB–8 GB,32 MB –8 GB (DIMM)	256 MB–1 GB,32 MB –4 GB (DIMM)	128 MB–1 GB, 32 MB –2 GB (DIMM)	288 MB,576 MB	576 MB – 1.1 GB	18–144 MB
I/O standard	HSUL- 12 1.2V	SSTL-15 Class I, II	SSTL-18 Class I, II	SSTL-2 Class I, II	HSTL-1.8V/ 1.5V	HSTL-1.2V and SSTL-12	HSTL-1.8V/ 1.5V
Data group width	8, 16, 32	4, 8, 16	4, 8, 16	4, 8, 16, 32	9, 18, 36	18, 36	9, 18, 36
Burst length	4, 8, 16	8	4, 8	2, 4, 8	2, 4, 8	2, 4, 8	2, 4
Number of banks	4, 8	8	8 (>1 GB), 4	4	8	16	—
Row/column access	Row before column	Row before column	Row before column	Row before column	Row and column together or multiplexed option	Row and column together or multiplexed option	—
CAS latency (CL)	—	5, 6, 7, 8, 9, 10	3, 4, 5	2, 2.5, 3	—	—	—

continued...

Parameter	LPDDR2	DDR3 SDRAM	DDR2 SDRAM	DDR SDRAM	RLDRAM II	RLDRAM 3	QDR II/+ SRAM
Posted CAS additive latency (AL)	—	0, CL-1, CL-2	0, 1, 2, 3, 4	—	—	—	—
Read latency (RL)	3, 4, 5, 6, 7, 8	RL = CL + AL	RL = CL + AL	RL = CL	3, 4, 5, 6, 7, 8	3-16	1.5, 2, and 2.5 clock cycles
On-die termination	—	Yes	Yes	No	Yes	Yes	Yes
Data strobe	Differential bidirectional	Differential bidirectional strobe only	Differential or single-ended bidirectional strobe	Single-ended bidirectional strobe	Free-running differential read and write clocks	Free-running differential read and write clocks	Free-running read and write clocks
Refresh requirement	Yes	Yes	Yes	Yes	Yes	Yes	No
Relative cost comparison	Higher than DDR SDRAM	Presently lower than DDR2	Less than DDR SDRAM with market acceptance	Low	Higher than DDR SDRAM,less than SRAM	Higher than DDR SDRAM,less than SRAM	Highest
Target market	Mobile devices that target low operating power	Desktops, servers, storage, LCDs, displays, networking, and communication equipment	Desktops, servers, storage, LCDs, displays, networking, and communication equipment	Desktops, servers, storage, LCDs, displays, networking, and communication equipment	Main memory, cache memory, networking, packet processing, and traffic management	Main memory, cache memory, networking, packet processing, and traffic management	Cache memory, routers, ATM switches, packet memories, lookup, and classification memories

Notes to Table:

1. 32-bit data bus operating at the maximum supported frequency in a Stratix® V FPGA.
2. 70% efficiency for DDR memories, which takes into consideration the bus turnaround, refresh, infinite burst length and random access latency and assumes 85% efficiency for QDR memories.
3. The lower frequency limit depends on the higher of the DLL frequency and the minimum operating frequency of the given EMIF protocol. (Except for DDR2 interfaces running on Stratix V devices.)

Intel supports the memory interfaces, provides various IP for the physical interface and the controller, and offers many reference designs; refer to Intel’s *Memory Solutions Center*.

For performance characteristics of the various high-speed memory interfaces, refer to the *External Memory Interface Spec Estimator* page on the Intel website.

Related Information

- Memory Solutions Center
- External Memory Interface Spec Estimator

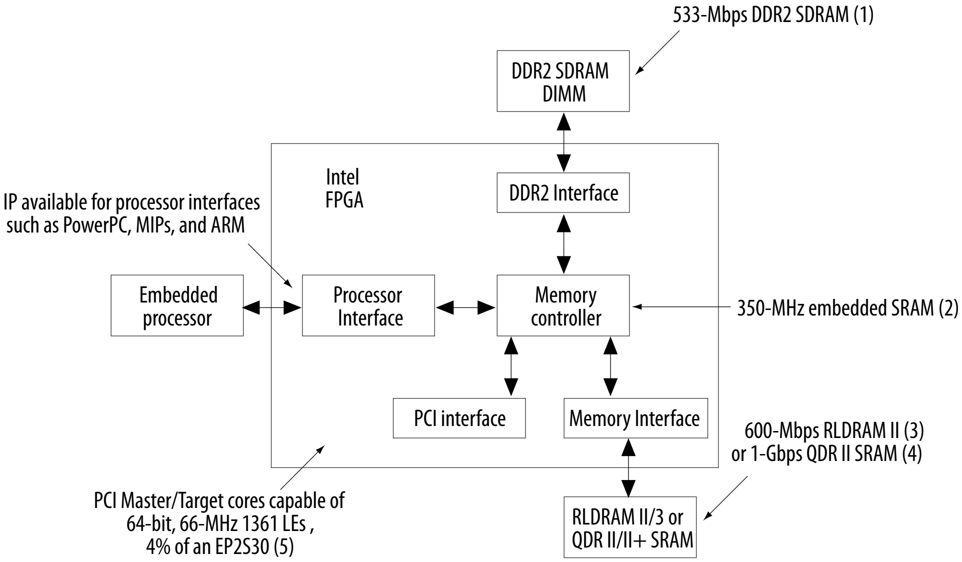
3.8. Example of High-Speed Memory in Embedded Processor

In embedded processor applications—any system that uses processors, excluding desktop processors—due to its very low cost, high density, and low power, DDR SDRAM is typically used for main memory.

Next-generation processors invest a large amount of die area to on-chip cache memory to prevent the execution pipelines from sitting idle. Unfortunately, these on-chip caches are limited in size, as a balance of performance, cost, and power must be taken into consideration. In many systems, external memories are used to add another level of cache. In high-performance systems, three levels of cache memory is common: level one (8 Kbytes is common) and level two (512 Kbytes) on chip, and level three off chip (2 Mbytes).

High-end servers, routers, and even video game systems are examples of high-performance embedded products that require memory architectures that are both high speed and low latency. Advanced memory controllers are required to manage transactions between embedded processors and their memories. Intel Arria series and Intel Stratix series FPGAs optimally implement advanced memory controllers by utilizing their built-in DQS (strobe) phase shift circuitry. The following figure highlights some of the features available in an Intel FPGA in an embedded application, where DDR2 SDRAM is used as the main memory and QDR II/II+ SRAM or RLDRAM II/3 is an external cache level.

Figure 4. Memory Controller Example Using FPGA



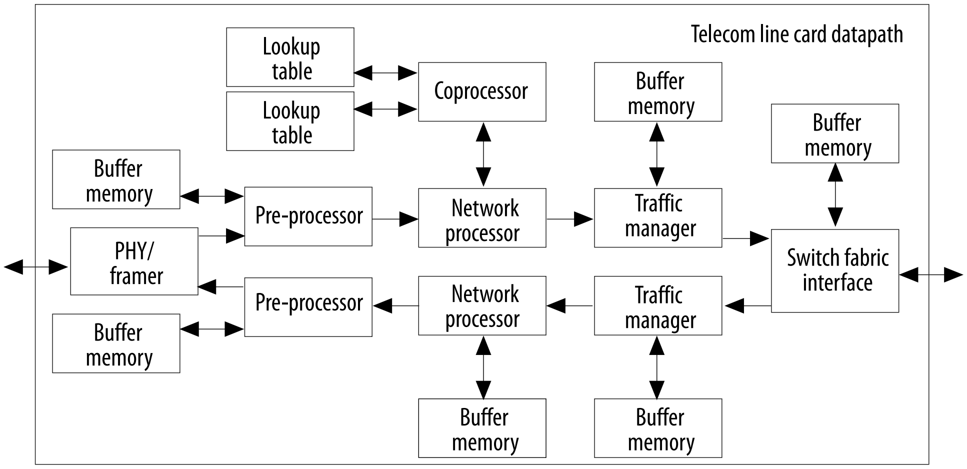
One of the target markets of RLDRAM II/3 and QDR/QDR II SRAM is external cache memory. RLDRAM II and RLDRAM 3 have a read latency close to synchronous SRAM, but with the density of SDRAM. A sixteen times increase in external cache density is achievable with one RLDRAM II/3 versus that of synchronous static RAM (SSRAM). In contrast, consider QDR and QDR II SRAM for systems that require high bandwidth and minimal latency. Architecturally, the dual-port nature of QDR and QDR II SRAM allows cache controllers to handle read data and instruction fetches completely independent of writes.

3.9. Example of High-Speed Memory in Telecom

Because telecommunication network architectures are becoming more complex, high-end network systems are running multiple 10-Gbps line cards that connect to multi-shelf switch fabrics scaling to terabits per second.

The following figure shows an example of a typical system line interface card. These line cards offer interfaces ranging from a single-port OC-192 to multi-port Gbps Ethernet, and consist of a number of devices, including a PHY/framer, network processors, traffic managers, fabric interface devices, and high-speed memories.

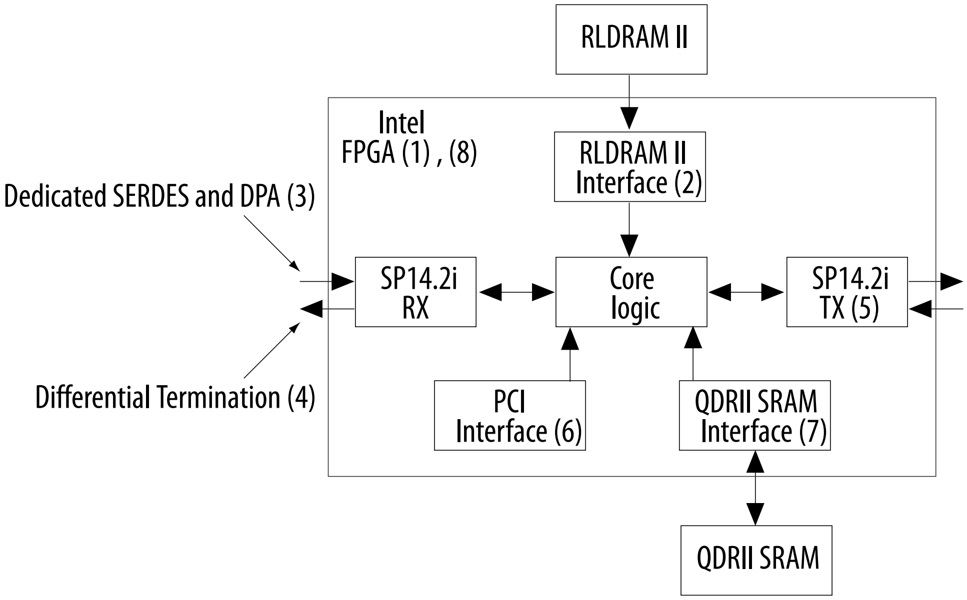
Figure 5. Typical Telecom System Line Interface Card



As packets traverse from the PHY/framer device to the switch fabric interface, they are buffered into memories, while the data path devices process headers (determining the destination, classifying packets, and storing statistics for billing) and control the flow of packets into the network to avoid congestion. Typically DDR/DDR2/DDR3 SDRAM and RLDRAM II/3 are used for large buffer memories off network processors, traffic managers, and fabric interfaces, while QDR and QDR II/II+ SRAMs are used for look-up tables (LUTs) off preprocessors and coprocessors.

In many designs, FPGAs connect devices together for interoperability and coprocessing, implement features that are not supported by ASIC devices, or implement a device function entirely. Intel Stratix series FPGAs implement traffic management, packet processing, switch fabric interfaces, and coprocessor functions, using features such as 1-Gbps LVDS I/O, high-speed memory interface support, multi-gigabit transceivers, and IP cores. The following figure highlights some of these features in a packet buffering application where RLDRAM II is used for packet buffer memory and QDR II SRAM is used for control memory.

Figure 6. FPGA Example in Packet Buffering Application



SDRAM is usually the best choice for buffering at high data rates due to the large amounts of memory required. Some system designers take a hybrid approach to the memory architecture, using SRAM to store the packet headers and DRAM to store the payload. The depth of the memories depends on the architecture and throughput of the system.

The buffer memory for the packet buffering application of an OC-192 line card (approximately 10 Gbps) must be able to sustain a minimum of one write and one read operation, which requires a memory bandwidth of 20 Gbps to operate at full line rate (more bandwidth is required if the headers are modified). The bandwidth requirement for memory is a key factor in memory selection. As an example, a simple first-order calculation using RLDRAM II as buffer memory requires a bus width of 48 bits to sustain 20 Gbps (300 MHz × 2 DDR × 0.70 efficiency × 48 bits = 20.1 Gbps), which needs two RLDRAM II parts (one ×18 and one ×36). RLDRAM II and RLDRAM 3 also inherently include the additional memory bits used for parity or error correction code (ECC). QDR and QDR II SRAM have bandwidth and low random access latency advantages that make them useful for control memory in queue management and traffic management applications. Another typical implementation for this memory is billing and packet statistics, where each packet requires counters to be read from memory, incremented, and then rewritten to memory. The high bandwidth, low latency, and optimal one-to-one read/write ratio make QDR SRAM ideal for this feature.

3.10. Document Revision History

Date	Version	Changes
March 2023	2023.03.06	• Removed Intel Arria 10 and Intel Stratix 10 references. • Removed *LPDDR3 Features* topic.
May 2017	2017.05.08	Rebranded as Intel.
October 2016	2016.10.31	Maintenance release.
May 2016	2016.05.02	Moved chapter from Volume 2 to Volume 1.
November 2015	2015.11.01	Added LPDDR3 features.
May 2015	2015.05.04	Maintenance release.
December 2014	2014.12.15	Modified note 3 on *Memory Selection Overview* table.
August 2014	2014.08.15	• Changed some values in the *Bandwidth for 32 bits*, *Bandwidth at % Efficiency*, *Performance / Clock frequency*, and *Altera-supported data rate* rows of the *Memory Selection Overview* table.
December 2013	2013.12.16	Removed references to Stratix II devices.
November 2012	6.0	Added RLDRAM 3 support.
June 2012	5.0	• Added LPDDR2 support. • Added Feedback icon.
November 2011	4.0	Moved and reorganized “Selecting your Memory” section to Volume 2: Design Guidelines.
June 2011	3.0	Added “Selecting Memory IP” chapter from Volume 2.
December 2010	2.1	• Moved protocol-specific feature information to the memory interface user guides in Volume 3. • Updated maximum clock rate information for 10.1.
July 2010	2.0	• Added specifications for DDR2 and DDR3 SDRAM Controllers with UniPHY. • Streamlined the specification tables. • Added reference to web-based Specification Estimator Tool.
January 2010	1.1	Updated DDR, DDR2, and DDR3 specifications.
November 2009	1.0	First published.

4. Selecting Your FPGA Device

Intel external memory solutions for UniPHY-based device families support three FPGA device families—Arria, Stratix, and Cyclone. These FPGA device families vary in terms of features, memory standards, speed grades, and cost.

For information on available Intel FPGA products, consult *Intel FPGAs and Programmable Devices* on the Intel website.

The following topics describe the factors that you should consider when selecting an FPGA device family. Refer to these topics together with the *Planning Pin and FPGA Resources* chapter, before you start implementing your external memory interface.

Related Information

- Intel FPGAs and Programmable Devices
- Planning Pin and FPGA Resources

4.1. Memory Standards

Intel devices support two common types of high-speed memories—dynamic random access memory (DRAM) and static random access memory (SRAM). The commonly used DRAM devices include DDR, DDR2, DDR3, and DDR4 SDRAM, LPDDR2, LPDDR3, RLDRAM II, and RLDRAM 3, while SRAM devices include QDR II, QDR II+, and QDR II + Xtreme SRAM.

For more information about these memory types, refer to the *Selecting Your Memory* chapter.

Different Intel FPGA devices support different memory types; not all Intel devices support all memory types and configurations. Before you start your design, you must select an Intel device, which supports the memory standard and configurations you plan to use.

In addition, Intel'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 suits your application.

For more information about the supported memory types and configurations, refer to the *External Memory Interface Spec Estimator* page on the Intel website.

Related Information

- Selecting Your Memory
- External Memory Interface Spec Estimator

4.2. I/O Interfaces

Ideally any interface should reside entirely in a single bank; however, interfaces that span across multiple adjacent banks or the entire side of a device are also fully supported.

Interfaces that span across sides (top and bottom, or left and right) and wraparound interfaces provide the same level of performance.

For information about the I/O interfaces supported for each device, and the locations of those I/O interfaces, refer to the *I/O Features* section in the appropriate device handbook.

4.3. Wraparound Interfaces

For maximum performance, Intel recommends that data groups for external memory interfaces should always be within the same side of a device, ideally reside within a single bank.

High-speed memory interfaces using top or bottom I/O bank versus left or right I/O bank have different timing characteristics, so the timing margins are also different. However, Intel can support interfaces with wraparound data groups that wrap around a corner of the device between vertical and horizontal I/O banks at some speeds. Some devices support wraparound interfaces that run at the same speed as row or column interfaces.

Arria II GX devices can support wraparound interface across all sides of devices that are not used for transceivers. Other UniPHY-supported Intel devices support only interfaces with data groups that wrap around a corner of the device.

4.4. Read and Write Leveling

The Arria V GZ, Stratix III, Stratix IV, and Stratix V 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.

Note: UniPHY-based designs do not require read leveling circuitry during read leveling operation.

For more information about read and write leveling, refer to *Leveling Circuitry* section in the *Functional Description - UniPHY* chapter of the *External Memory Interface Handbook*.

Related Information

Functional Description - UniPHY

4.5. Dynamic OCT

The Arria II GZ, Arria V, Cyclone V, Stratix III, Stratix IV, and Stratix V devices support dynamic calibrated OCT.

Dynamic calibrated OCT allows the specified series termination to be enabled during writes, and parallel termination to be enabled during reads. These I/O features allow you to simplify PCB termination schemes.

4.6. Device Settings Selection

After you have selected the appropriate FPGA device family for your memory interface, configure the device settings of your selected FPGA device family to meet your design needs.

Refer to the device ordering code and determine the appropriate device settings for your target device family.

For more information about the ordering code for your target device, refer to the “Ordering Information” section in volume 1 of the respective device handbooks.

The following sections describe the ordering code and how to select the appropriate device settings based on the ordering code to meet the requirements of your external memory interface.

4.6.1. Device Speed Grade

The device speed grade affects the device timing performance, timing closure, and power utilization.

The device with the smallest number is the fastest device and vice-versa. Generally, the faster devices cost more.

4.6.2. Device Operating Temperature

The operating temperature of the FPGA is divided into the following categories:

- Commercial grade—Used for all device families. The operating temperature range from 0 degrees C to 85 degrees C.
- Industrial grade—Used for all device families. The operating temperature range from -40 degrees C to 100 degrees C.
- Military grade—Used for Stratix IV device family only. The operating temperature range from -55 degrees C to 125 degrees C.
- Automotive grade—Used for Cyclone V device families only. The operating temperature range from -40 degrees C to 125 degrees C.

4.6.3. Device Package Size

Each FPGA family has a range of package sizes.

Package size refers to the actual size of an FPGA device and corresponds to the number of pin counts. For example,the package size for the smallest FPGA device in the Stratix IV family is 29 mm x 29 mm, categorized under the F780 package option, where F780 refers to a device with 780 pin counts.

For more information about the package size available for your device, refer to the respective device handbooks.

4.6.4. Device Density and I/O Pin Counts

An FPGA device of the same device family and package size also varies in terms of device density and I/O pin counts.

For example, after you have selected the Stratix IV device family with the F780 packaging option, you must determine the type of device models that ranges from EP4GX70 to EP4GX230. Each of these devices has similar speed grades that range from grade 2 to grade 4, but are different in density.

Device Density

Device density refers to the number of logic elements (LEs). For example, PLLs, memory blocks, and so on. An FPGA device with higher density contains more logic elements in less area.

I/O Pin Counts

To meet the growing demand for memory bandwidth and memory data rates, memory interface systems use parallel memory channels and multiple controller interfaces. However, the number of memory channels is limited by the package pin count of the Intel devices. Therefore, you must consider device pin count when you select a device; you must select a device with enough I/O pins for your memory interface requirement.

The number of device pins required depends on the memory standard, the number of memory interfaces, and the memory data width. For example, a ×72 DDR3 SDRAM single-rank interface requires 125 I/O pins:

- 72 DQ pins (including ECC)
- 9 DM pins
- 9 DQS, DQSn differential pin pairs
- 17 address pins (address and bank address)
- 7 command pins (CAS, RAS, WE, CKE, ODT, reset, and CS)
- 1 CK, CK# differential pin pair

Note: For more information about the number of embedded memory, PLLs and user I/O counts that are available for your device, refer to the respective device handbooks. For the number of DQS groups available for each FPGA device, refer to the respective device handbooks.

Note: For the maximum number of controllers that is supported by the FPGAs for different memory types, refer to the *Planning Pin and FPGA Resources* chapter.

Intel devices do not limit the interface widths beyond the following requirements:

- The DQS, DQ, clock, and address signals of the entire interface must reside within the same bank or side of the device if possible, to achieve better performance. Although wraparound interfaces are also supported at limited frequencies.
- The maximum possible interface width in any particular device is limited by the number of DQS and DQ groups available within that bank or side.
- 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 clock, address, and command pin placement requirements.
- The greater the number of banks, the greater the skew. Intel recommends that you always compile a test project of your desired configuration and confirm that it meets timing requirement.

Your pin count calculation also determines which device side to use (top or bottom, left or right, and wraparound).

Note: When assigning DQS and DQ pins in Arria® II GX devices, you are allowed to use only twelve of the sixteen I/O pins in an I/O module as DQ pins. The remaining four pins can be used only as input pins.

Note: For DQS groups pin-out restriction format, refer to *Arria II GX Pin Connection Guidelines*.

Note: The Arria II GX and Stratix V devices do not support interfaces on the left side of the device. There are no user I/O pins, other than the transceiver pins available in these devices.

Related Information

Arria II GX Pin Connection Guidelines

4.7. Document Revision History

Date	Version	Changes
March 2023	2023.03.06	Removed Intel Arria 10 and Intel Stratix 10 references.
October 2016	2016.10.31	Maintenance release.
May 2016	2016.05.02	Moved chapter from Volume 2 to Volume 1.
November 2015	2015.11.02	Added LPDDR3 to memory standards.
May 2015	2015.05.04	Maintenance release.
December 2014	2014.12.15	Maintenance release.
August 2014	2014.08.15	Maintenance release.
December 2013	2013.12.16	Removed references to Cyclone III and Cyclone IV devices.
June 2012	5.0	• Added LPDDR2 support. • Added Feedback icon.
November 2011	4.0	Moved and reorganized “Selecting your FPGA” section to Volume 2: Design Guidelines.
June 2011	3.0	Added “Selecting a Device” chapter from Volume 2.
December 2010	2.1	• Moved protocol-specific feature information to the memory interface user guides in Volume 3. • Updated maximum clock rate information for 10.1.

continued...

Date	Version	Changes
July 2010	2.0	• Added specifications for DDR2 and DDR3 SDRAM Controllers with UniPHY. • Streamlined the specification tables. • Added reference to web-based Specification Estimator Tool.
January 2010	1.1	Updated DDR, DDR2, and DDR3 specifications.
November 2009	1.0	First published.