

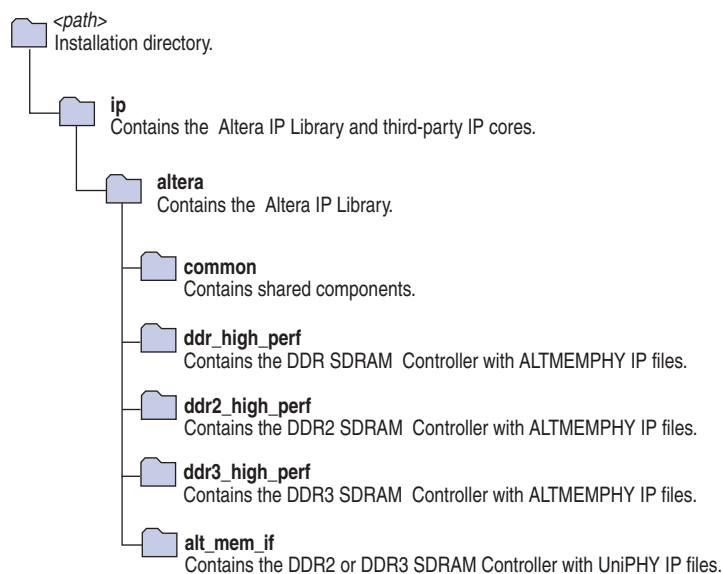
本章节对 Altera® IP 内核设计流程作了概述，帮助您快速着手 Altera IP 内核的设计。Altera IP Library 是 Quartus® II 安装进程的一部分，您可以从库中选择并参数化任意 Altera IP 内核。Altera 提供了一个集成的参数编辑器，用于定制支持各种应用的 IP 内核。参数编辑器将指导您设置参数值及选择端口。下面章节介绍了基本设计流程以及 Altera IP 内核的使用。

安装和许可授权

通过 Altera 网站 (www.altera.com) 上的 Quartus II 软件来分布、下载和安装 Altera IP Library。

图 9-1 显示了安装具有存储器 IP 的存储控制器之后的目录结构，其中 *<path>* 是安装目录。在 Windows 上的默认安装目录是 `c:\altera\<version>`；在 Linux 上是 `/opt/altera<version>`。

图 9-1. 目录结构



您可以在仿真中和硬件中对 IP 内核进行不断评估，直到对其功能性和性能满意为止。如果您想将设计转化到生产，那么就需要购买 IP 内核的许可。购买后，从 Altera 网站的 [Altera Licensing](#) 页面生成一个许可文件并安装到您的计算机中。关于详细信息，请参考 [Altera Software Installation and Licensing](#)。

免费评测

Altera's OpenCore Plus 评估特性仅适用于 DDR、DDR2 和 DDR3 SDRAM 存储控制器。将 Altera 存储控制器作为 Altera 器件中的硬核 IP 来实现时不需要许可密钥以及费用。通过使用 OpenCore Plus 评估特性，您可以执行以下操作：

- 仿真您系统中的宏功能行为（Altera MegaCore® 功能或者 AMPPSM 宏功能）。
- 验证您的设计功能性，快速而方便地评估其大小及速度。
- 对于包含 MegaCore 功能的设计，生成有时限的器件编程文件。
- 编程器件并在硬件中验证您的设计。

如果您完全满意 MegaCore 的功能性和性能，并打算将您的设计转化到生产，那么您需要购买 MegaCore 的使用许可。

OpenCore Plus 超时行为

OpenCore Plus 硬件评估支持以下两种操作模式：

- 未束缚 (Untethered) — 设计在有限时间内运行
- 束缚 (tethered) — 需要电路板与主机之间的一个连接。如果设计中的所有宏功能都支持 tethered 模式，那么器件能够运行更长时间，甚至无限期地运行。

达到最严格的评估时间后，器件中所有宏功能将同时超时。如果器件中有一个以上的宏功能，那么一个特定宏功能的超时 (time-out) 行为可能会被其它宏功能的超时行为屏蔽。



对于 MegaCore 功能，未束缚超时 (untethered time-out) 为 1 小时；束缚超时 (tethered time-out) 值为无限期。

硬件评测时间到期，设计将停止运行，local_ready 输出则变低。

设计流程

通过使用下面其中的一个流程，您可以实现具有 ALTMEMPHY IP 或者 UniPHY IP 的存储控制器。

- MegaWizard™ Plug-In Manager 流程
- SOPC Builder 流程
- Qsys 流程

图 9-2 显示了使用可用流程在 Quartus II 软件中创建一个系统的流程。

图 9-2. 设计流程 (1)

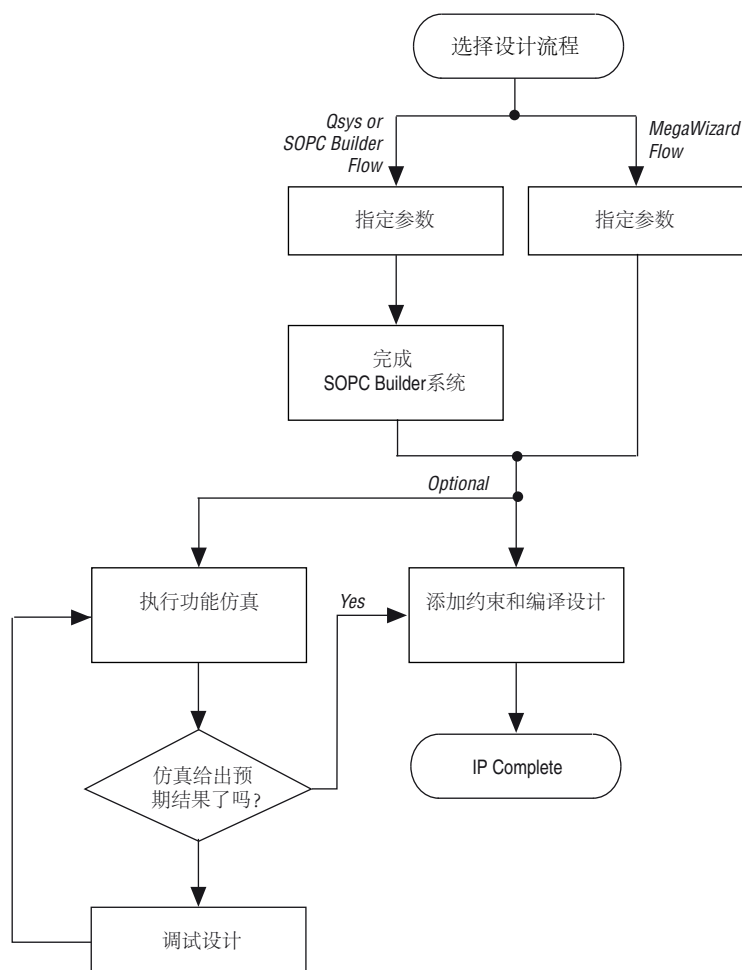


图 9-2 注释：

(1) Altera IP 内核可能支持或不支持 Qsys 和 SOPC Builder 设计流程。

MegaWizard Plug-In Manager 流程具有以下优点：

- 使您能够参数化 IP 内核，并例化到现有设计中。
- 对于某些 IP 内核，此流程生成一个完整的实例设计和测试台。

SOPC Builder 流程具有以下优点：

- 生成仿真环境
- 使您能够集成 Altera 提供的定制组件
- 使用 Avalon® 存储器映射 (Avalon-MM) 接口

与 SOPC Builder 流程相比，Qsys 流程还具有以下优点：

- 提供层次化设计的可视化
- 通过互联单元和流水线支持更高性能
- 提供与 Quartus II 软件更紧密的集成

MegaWizard Plug-In Manager 流程

MegaWizard Plug-In Manager 流程使您能够定制具有 ALTMEMPHY 或者 UniPHY IP 的存储控制器，并能够手动将功能集成到设计中。

指定参数

使用 MegaWizard Plug-In Manager 流程指定参数的步骤是：

1. 使用 File 菜单中的 **New Project Wizard** 创建一个 Quartus II 工程。
2. 在 Quartus II 软件中，运行 Tools 菜单中的 **MegaWizard Plug-in Manager**，根据 MegaWizard Plug-In Manager 界面中的提示创建或编辑一个定制的 IP 内核。
3. 在 **External Memory** 文件夹中的 **Installed Plug-Ins** 列表选择一个具有存储器 IP 的存储控制器。
4. 指定 **Parameter Settings** 标签中全部页面上的参数。



关于这些参数的详细说明，请参考第 9-39 页“参数化基于 ALTMEMPHY IP 的存储控制器”和第 9-54 页“参数化基于 UniPHY IP 的存储控制器”。



UniPHY IP 内核对特定应用提供了预置参数 (preset parameter)。您需要根据操作频率对某些预置参数进行修改。一系列需要修改的典型参数 — 存储器参数 (Memory Parameter)：Memory CAS 延迟设置和 Memory CAS 写延迟设置。存储器时序 (Memory Timing)：tWTR, tFAW, tRRD 和 tRTP。

5. 根据需要修改预置参数。
 - a. 点击箭头，展开 **Presets** 列表。
 - b. 选择所需的预置，然后点击 **Apply**。
6. 若要创建，修改或者删除您定制的预置，则需要点击 **Presets** 列表底部的 **New**, **Update** 或者 **Delete**。
7. 如果 IP 内核提供一个仿真模型，那么需要在向导中指定相应的选项以生成仿真模型。



Altera IP 支持多种仿真模型，包括仿真特定的 IP 功能模型和加密的 RTL 模型，以及纯文本 RTL 模型。这些都是周期精确 (cycle-accurate) 模型。通过使用业界标准的 VHDL 或者 Verilog HDL 仿真器，这些模型能够支持您 IP 内核实例的快速功能仿真。对于某些内核，仅生成纯文本 RTL 模型，您可以仿真此模型。



要了解关于 Altera IP 内核的功能仿真模型的详细信息，请参考 *Quartus II Handbook* 卷 3 中的 *Simulating Altera Designs* 章节。



仿真模型只能用于仿真，不能用于综合或者任何其它用途。如果用于综合，将生成一个非功能设计。

8. 这一步骤适用于具有 ALTMEMPHY IP 的存储控制器。如果参数编辑器包括 **EDA** 和 **Summary** 标签，那么需要执行下面步骤：
 - a. 某些第三方综合工具能够使用一个包含 IP 内核结构，但不包含详细逻辑的网表来对包括其设计的时序和性能进行优化。如果您的综合工具和 IP 内核支持这一特性，那么通过开启 **Generate netlist** 可以使用此特性。



当采用 VHDL 仿真模型时，MegaWizard Plug-In Manager 仍然生成 `<variation_name>_alt_mem_phy.v` 用于 Quartus II 综合。不要将此文件用于仿真，而要将 `<variation_name>.vho` 用于仿真。

ALTMEMPHY 宏功能仅支持功能仿真。使用 ALTMEMPHY 宏功能时，您不能执行时序或门级仿真。

- b. 在 **Summary** 标签上，选择想要生成的文件（如果可选）。一个灰色钩形符号表明一个自动生成的文件。所有其它文件都是可选的。



如果您的 IP 内核支持文件选择，那么在生成内核后，一个生成报告（`<variation_name>.html`）会出现在您的工程目录中。此文件包含生成文件的相关信息。

9. 点击 **Finish** 按钮，参数编辑器会生成您 IP 内核的顶层 HDL 代码和包括仿真文件的仿真目录。



在更正信息窗口中列出的全部参数化错误以前，**Finish** 按钮不可用。

10. 如果提示添加 `.qip` 到当前的 Quartus II 工程中，那么点击 **Yes**。您也可以开启 **Automatically add Quartus II IP Files to all projects**。
11. 这一步骤适用于具有 ALTMEMPHY IP 的存储控制器。

将 `<variation_name>_example_top.v` 或者 `.vhd` 设置成工程顶层设计文件。

 - a. 在 File 菜单中，点击 **Open**。
 - b. 找到 `<variation_name>_example_top`，点击 **Open**。
 - c. 在 Project 菜单中，点击 **Set as Top-Level Entity**。

现在，您就可以将您定制的 IP 内核整合到您的设计中，进行仿真并编译。整合过程中您必须进行相应的管脚分配。仿真期间，将设计映射到硬件之前，您可以创建一个虚拟管脚来避免对顶层信号进行特定的管脚分配。

对于 IP 内核，生成过程也会创建完整的实例设计。一个用于硬件测试的实例设计位于 `<variation_name>_example_design/example_project/` 目录中。一个用于 RTL 仿真的实例设计位于 `<variation_name>_example_design/simulation/` 目录中。



要了解关于 Quartus II 软件，包括虚拟管脚和 MegaWizard Plug-In Manager 的详细信息，请参考 *Quartus II Help*。

约束设计

生成存储器 IP MegaCore 功能后，您需要使用 Quartus II TimeQuest Timing Analyzer 设置时序约束并执行时序分析。当生成 MegaCore 功能时，MegaWizard Plug-In Manager 也生成一个 Synopsis Design Constraint File (`.sdc`)，`<variation_name>.sdc` 和管脚分配脚本，`<variation_name>.pin_assignments.tcl`。`.sdc` 和 `<variation_name>.pin_assignments.tcl` 脚本都支持多个实例。这些脚本循环访问内核的所有实例，并对它们应用相同的约束。您可以从外部器件数据表得到时序约束，从电路板布局得到容差。

要了解关于时序约束和分析的详细信息，请参考 *Analyzing Timing of Memory IP* 章节。

添加管脚和 DQ 组约束

`<variation_name>.pin_assignments.tcl` 脚本设置存储器 IP 的 I/O 标准和输入 / 输出匹配。此脚本帮助关联 DQ 管脚组，使 Quartus II Fitter 能够对它们正确布局。

管脚分配脚本不创建设计的 PLL 参考时钟。您必须对设计创建时钟，并对 MegaCore variation 生成的实例驱动器和测试台的信号提供管脚分配。

运行 `<variation_name>.pin_assignments.tcl` 脚本以添加输入和输出匹配，I/O 标准和 DQ 组分配到实例设计。请按照以下步骤运行管脚分配脚本：

1. 在 Processing 菜单中指向 **Start**，点击 **Start Analysis and Synthesis**。
2. 在 Tools 菜单上点击 **Tcl Scripts**。
3. 指定 `pin_assignments.tcl` 并点击 **Run**。



如果 PLL 输入参考时钟没有与存储器接口 I/O 相同的 I/O 标准，那么由于不兼容的 I/O 标准不能被布局在同一 I/O bank 中，可能出现 no-fit。



如果从 Quartus II 的较早版本更新您的存储器 IP，那么需要执行以下步骤：


- 对于 UniPHY IP，在 Quartus II 的较新版本中重新运行 `pin_assignments.tcl` 脚本。
- 对于 ALTMEMPHY IP，删除所有的存储器 I/O 非位置约束，重新运行 `pin_assignments.tcl` 脚本。

编译设计

约束您的设计后，在 Quartus II 中编译您的设计以生成时序报告来验证时序是否已被满足。

在 Processing 菜单中点击 **Start Compilation** 来编译设计。

编译顶层文件后，您可以执行 RTL 仿真或者编程您的目标 Altera 器件，验证硬件中顶层文件。

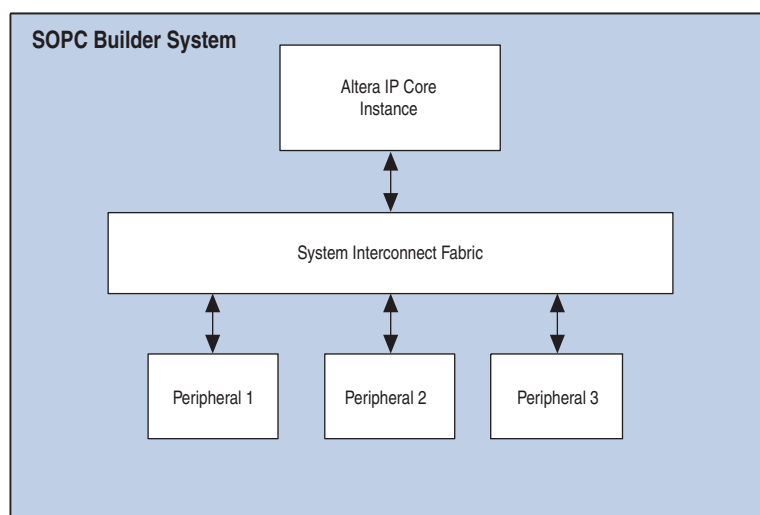
 要了解关于仿真存储器 IP 的详细信息，请参考 *Simulating Memory IP* 章节。


SOPC Builder 流程


您可以使用 SOPC Builder 构建一个包含您定制的 IP 内核的系统。您能够轻松添加其它组件并快速创建 SOPC Builder 系统。SOPC Builder 自动生成 HDL 文件，此文件包含所有特定组件和互联。SOPC Builder 对能够修改的默认连接进行定义。HDL 文件可以通过 Quartus II 软件编译，以生成用于 Altera 器件编程的输出文件。

图 9-3 显示了一个 SOPC Builder 系统实例的结构图。

图 9-3. SOPC Builder 系统




 要了解关于系统互联架构的详细信息，请参考 *SOPC Builder User Guide* 中的 *System Interconnect Fabric for Memory-Mapped Interfaces* 和 *System Interconnect Fabric for Streaming Interfaces* 章节以及 *Avalon Interface Specifications*。

 要了解关于 SOPC Builder 和 Quartus II 软件的详细信息，请参考 *SOPC Builder User Guide* 中的 *SOPC Builder Features* 和 *Building Systems with SOPC Builder* 章节以及 Quartus II Help。

指定参数

在参数编辑器中指定所需的参数，关于这些参数的详细说明，请参考 SOPC Builder 流程中的“指定 IP 内核参数”，执行下列步骤：

1. 使用 File 菜单中的 **New Project Wizard** 创建一个新的 Quartus II 工程。
2. 在 Tools 菜单中，点击 **SOPC Builder**。
3. 对于新的系统，需要指定系统名称和语言。
4. 在 **System Contents** 标签上，双击您的 IP 内核名，将其添加到您的系统中。出现相关的参数编辑器。
5. 在参数编辑器中指定所需的参数。关于这些参数的详细说明，请参考第 9-39 页“参数化基于 ALTMEMPHY IP 的存储控制器”和第 9-54 页“参数化基于 UniPHY IP 的存储控制器”。

 UniPHY IP 内核对特定应用提供了预置参数。您可能需要根据操作频率来修改某些预置参数。需要修改的一系列典型参数是 — 存储器参数 (Memory Parameters):Memory CAS 延迟设置和 Memory CAS 写延迟设置。存储器时序 (Memory Timing): tWTR, tFAW, tRRD 和 tRTP。

6. 要使用预置参数，需要点击箭头展开 **Presets** 列表，选择所需的预置，然后点击 **Apply**。

 您也可以添加和修改您自己的预置。

7. 要创建，修改或者删除您自己定制的预置，需要点击 **Presets** 列表底部的 **New**, **Update** 或者 **Delete**。

 当参数化这些内核时，您也必须开启 **Controller Settings** 标签上的 **Generate SOPC Builder compatible resets**。

8. 点击 **Finish** 来完成 IP 内核实例，并将其添加到系统中。

在更正信息窗口中列出的全部参数化错误以前，**Finish** 按钮不可用。

完成 SOPC Builder 系统

执行下列步骤来完成 SOPC Builder 系统：

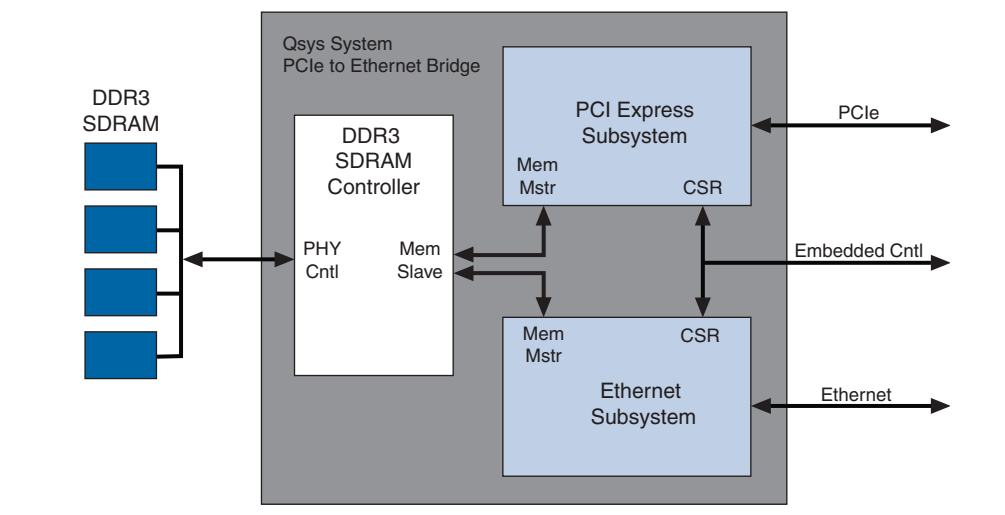
1. 添加和参数化所有额外组件。某些 IP 内核包括完整的 SOPC Builder 系统设计实例。
2. 使用 **System Contents** 标签上的 Connection 面板来连接器件。
3. 默认情况下不显示时钟名。要显示 **System Contents** 标签中 **Module Name** 列中时钟名和 **Clock** 列中的时钟，需要点击 **Filters** 以显示 **Filters** 对话框。在 **Filter** 列中，点击 **All**。
4. 点击 **Generate** 生成系统。SOPC Builder 生成系统并产生 *<system name>.qip* 文件，此文件包含了约束和 Quartus II Compiler 中处理 IP 内核或系统所需要的信息。
5. 在 Quartus II 软件中，点击 **Add/Remove Files in Project** 将 .qip 添加到工程中。
6. 在 Quartus II 软件中编译您的设计。

Qsys 系统集成工具设计流程

您可以使用 Qsys 系统集成工具来构建一个包含您定制 IP 内核的系统。您能够轻松添加其它器件并快速创建 Qsys 系统。Qsys 自动生成 HDL 文件，此文件包含所有指定器件和互联。在 Qsys 中指定所需的连接。HDL 文件可以通过 Quartus II 软件编译，以生成用于 Altera 器件编程的输出文件。Qsys 生成 Verilog HDL 仿真模型，用于包含您系统的 IP 内核。

图 9 - 4 显示了 Qsys 系统实例的高级结构图。

图 9 - 4. Qsys 系统实例




- 要了解关于 Qsys 系统互联的详细信息，请参考 *Quartus II Handbook* 卷 1 中的 *Qsys Interconnect* 章节和 *Avalon Interface Specifications*。
- 要了解关于 Qsys 工具和 Quartus II 软件的详细信息，请参考 *Quartus II Handbook* 卷 1 中的 *System Design with Qsys* 章节和 Quartus II Help。


指定参数

要使用 Qsys 流程指定您 IP 内核的参数，需执行下列步骤：

1. 使用 File 菜单上的 **New Project Wizard** 创建一个新的 Quartus II 工程。
2. 在 Tools 菜单上，点击 **Qsys**。
3. 在 **Component Library** 窗口中，双击您的 IP 内核名将其添加到您的系统中。出现相关参数编辑器。

- 在 Qsys 工具中的所有标签中指定所需参数。关于这些参数的详细说明，请参考第 9 - 39 页 “参数化基于 ALTMEMPHY IP 的存储控制器” 和第 9 - 54 页 “参数化基于 UniPHY IP 的存储控制器”。

 如果您的设计包括外部存储器接口 IP 内核，那么当参数化这些内核时必须开启 **Controller Settings** 标签上的 **Generate power-of-2 bus widths for SOPC Builder**。

 UniPHY IP 内核对特定应用提供预置参数。您可能需要根据操作频率来修改某些预置参数。需要修改的一系列典型参数是 — 存储器参数 (Memory Parameters): Memory CAS 延迟设置和 Memory CAS 写延迟设置。存储器时序 (Memory Timing): tWTR, tFAW, tRRD 和 tRTP。

4. 要使用预置参数，需要点击箭头展开 **Presets** 列表，选择所需的预置，然后点击 **Apply**。

 您也可以添加和修改您自己的预置。


5. 要创建，修改或者删除您自己定制的预置，需要点击 **Presets** 列表底部的 **New**, **Update** 或者 **Delete**。
6. 点击 **Finish** 来完成 IP 内核实例，并将其添加到系统中。

 在更正信息窗口中列出的全部参数化错误以前，**Finish** 按钮不可用。

完成 Qsys 系统

执行下列步骤来完成 Qsys 系统：

1. 添加和参数化所有额外组件。
2. 使用 **System Contents** 标签上的 **Connection** 面板来连接器件。
3. 在 **Export** 列中，输入是顶层 Qsys 系统端口的任何连接名称。
4. 如果要仿真您的 Qsys 系统，在 **Generation** 标签上，选择 **Create testbench Qsys system** 到 **Standard, BFM for standard Avalon interfaces** 以创建一个具有（附加到所有导出接口的）总线功能模型 (BFM) 的测试台，或者选择 **Simple, BFM for clocks and resets** 以创建一个具有只驱动时钟和复位接口的 BFM 的测试台。
5. 要同时生成测试台 Qsys 系统的仿真模型，需要将 **Create testbench simulation model** 设置成 **Verilog** 或者 **VHDL**。生成仿真模型前，设置此选项为 **None** 以查看或修改生成的测试台系统。
6. 如果您的系统不是 Quartus II 工程的一部分，但想生成综合寄存器传输语言 (RTL) 或者高级硬件描述语言 (HDL) 文件，则需要开启 **Create HDL design files for synthesis**。
7. 点击 **Generate** 生成系统。Qsys 生成系统并产生 *<system name>.qip* 文件，此文件包含了约束和 Quartus II Compiler 中处理 IP 内核或系统所需要的信息。
8. 在 Quartus II 中，点击 **Add/Remove Files in Project**，将 *.qip* 添加到工程中。
9. 在 Quartus II 中编译您的工程。

 要确保 **memory** 和 **oct** 接口被导出到顶层 RTL 文件，一定要小心，不要意外地重命名或删除 **System Contents** 标签的 **Export** 列中的这些接口。

Qsys 和 SOPC Builder 接口

表 9-1 和表 9-2 列出了用于 Qsys 和 SOPC Builder 中每个接口的带有 UniPHY 信号的 DDR2 和 DDR3 SDRAM，并提供了连接这些接口的说明和指南。

表 9-1. 具有 UniPHY 接口的 DDR2 SDRAM 控制器 (1/5)

接口中的信号	接口类型	说明 / 如何连接
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	对 PHY 中的 PLL 和全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，控制器将置位此接口。
afi_reset_in interface		
afi_reset_n	复位输入 (PLL slave)	当接口处于 PLL slave 模式中时，此接口是一个必须连接到 PLL master 模式中配置相同的存储器接口的 afi_reset_n 输出的复位输入。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 此接口在 PLL master 或者 no sharing 模式中时是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 当接口处于 PLL slave 模式时，您必须将此 afi_clk 输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI 半时钟，它是 afi_clk 频率的一半。 当此接口处于 PLL master 或者 no sharing 模式中时，此接口是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI 半时钟，它是 afi_clk 频率的一半。 当接口处于 PLL slave 模式时，这是一个时钟输入，您必须将此时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。

表 9 - 1. 具有 UniPHY 接口的 DDR2 SDRAM 控制器 (2/5)

接口中的信号	接口类型	说明 / 如何连接
memory interface (DDR2 SDRAM)		
mem_a	导线管 (conduit)	PHY 与存储器件之间的接口信号。
mem_ba		
mem_ck		
mem_ck_n		
mem_cke		
mem_cs_n		
mem_dm		
mem_ras_n		
mem_cas_n		
mem_we_n		
mem_dq		
mem_dqs		
mem_dqs_n		
mem_odt		
mem_ac_parity		
mem_err_out_n		
mem_parity_error_n		
memory interface (LPDDR2)		
mem_ca	导线管 (conduit)	PHY 与存储器件之间的接口信号。
mem_ck		
mem_ck_n		
mem_cke		
mem_cs_n		
mem_dm		
mem_dq		
mem_dqs		
mem_dqs_n		
avl interface		
avl_ready	Avalon-MM Slave	存储器接口与用户逻辑之间的 Avalon-MM 接口信号。
avl_burst_begin		
avl_addr		
avl_rdata_valid		
avl_rdata		
avl_wdata		
avl_be		
avl_read_req		
avl_write_req		
avl_size		

表 9 - 1. 具有 UniPHY 接口的 DDR2 SDRAM 控制器 (3/5)

接口中的信号	接口类型	说明 / 如何连接
status interface		
local_init_done	导线管 (conduit)	存储器接口状态信号。
local_cal_success		
local_cal_fail		
oct interface		
rup (Stratix® III/IV, Arria® II GZ)	导线管 (conduit)	rup/rdn 或 rzqin 的 OCT 参考电阻管脚。
rdn (Stratix III/IV, Arria II GZ)		
rzq (Stratix V, Arria V, Cyclone V)		
local_powerdown interface		
local_powerdn_ack	导线管 (conduit)	仅当开启 Enable Auto Powerdown 时，控制器的这一断电接口才被使能。
pll_sharing interface		
pll_mem_clk	导线管 (conduit)	PLL sharing 的接口信号，连接 PLL masters 与 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
pll_write_clk		
pll_addr_cmd_clk		
pll_locked		
pll_avl_clk		
pll_config_clk		
pll_hr_clk		
pll_p2c_read_clk		
pll_c2p_write_clk		
pll_dr_clk		
dll_sharing interface		
dll_delayctrl	导线管 (conduit)	DLL sharing 接口，用于连接 DLL masters 与 DLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
dll_pll_locked		
oct_sharing interface		
seriesterminationcontrol	导线管 (conduit)	OCT sharing 接口，用于连接 OCT masters 与 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol		
hcx_dll_reconfig interface		
dll_offset_ctrl_addnsub	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 DLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以使能 DLL 重配置。
dll_offset_ctrl_offset		
dll_offset_ctrl_addnsub ⁽¹⁾		
dll_offset_ctrl_offset ⁽¹⁾		
dll_offset_ctrl_offsetctrl out ⁽¹⁾		
dll_offset_ctrl_b_offsetctrlout ⁽¹⁾		

表 9 - 1. 具有 UniPHY 接口的 DDR2 SDRAM 控制器 (4/5)

接口中的信号	接口类型	说明 / 如何连接
hcx_pll_reconfig interface		
configupdate	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 PLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以使能 PLL 重配置。
phasecounterselect		
phasestep		
phaseupdown		
scanclk		
scanclkena		
scandata		
phasedone		
scandataout		
scandone		
hcx_rom_reconfig interface		
hc_rom_config_clock	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 ROM 加载器接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以控制定序器 ROM 的加载。
hc_rom_conig_datain		
hc_rom_config_rom_data_ready		
hc_rom_config_init		
hc_rom_config_init_busy		
hc_rom_config_rom_rden		
hc_rom_config_rom_address		
autoprecharge_req interface		
local_autopch_req	导线管 (conduit)	预加电接口，用于与定制控制模块的连接。仅当开启 Auto-precharge Control 时，此接口才被使能。
user_refresh interface		
local_refresh_req	导线管 (conduit)	用户刷新接口，用于与定制控制模块的连接。仅当开启 User Auto-Refresh Control 时，此接口才被使能。
local_refresh_chip		
local_refresh_ack		
self_refresh interface		
local_self_rfsh_req	导线管 (conduit)	自刷新接口，用于与定制控制模块的连接。仅当开启 Self-refresh Control 时，此接口才被使能。
local_self_rfsh_chip		
local_self_rfsh_ack		
ecc_interrupt interface		
ecc_interrupt	导线管 (conduit)	ECC 中断信号，用于与定制控制模块的连接。仅当开启 Error Detection and Correction Logic 时，此接口才被使能。

表 9 - 1. 具有 UniPHY 接口的 DDR2 SDRAM 控制器 (5/5)

接口中的信号	接口类型	说明 / 如何连接
csr interface		
csr_write_req	Avalon-MM Slave	配置和状态寄存器信号，用于存储器接口和与 Avalon-MM master 的连接。仅当开启 Configuration and Status Register 时，此接口才被使能。
csr_read_req		
csr_waitrequest		
csr_addr		
csr_be		
csr_wdata		
csr_rdata		
csr_rdata_valid		
Hard Memory Controller MPFE FIFO Clock Interface		
mp_cmd_clk	导线管 (conduit)	使能 Hard Memory Interface 时，三个 FIFO 缓存 (command, read data 和 write data) 在 MPFE 中被创建。每个 FIFO 缓存有其各自的时钟和复位端口。仅当开启 Enable Hard Memory Interface 时，此接口才被使能。
mp_rfifo_clk		
mp_wfifo_clk		
mp_cmd_reset		
mp_rfifo_reset		
mp_wfifo_reset		
Hard Memory Controller Bonding Interface		
bonding_in_1	导线管 (conduit)	绑定接口用于绑定两个控制器以扩展带宽。仅当开启 Export Bonding Interface 时，此接口才被使能。
bonding_in_2		
bonding_in_3		
bonding_out_1		
bonding_out_2		
bonding_out_3		
表 9-1 注释:		
(1) 信号仅在 DLL master 模式中可用。		

表 9 - 2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (1/6)

接口中的信号	接口类型	说明 / 如何连接
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	对 PHY 中的 PLL 和全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，此接口被置位。

表 9 - 2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (2/6)

接口中的信号	接口类型	说明 / 如何连接
afi_reset_in interface		
afi_reset_n	复位输入 (PLL slave)	当此接口处于 PLL slave 模式时，它是一个复位输入，您必须在 PLL master 模式中将该复位输入连接到相同配置的存储器接口的 afi_reset_n 输出。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 当此接口处于 PLL slave 模式时，这是一个时钟输入，您必须将该时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL slave 模式时，您必须将此 afi_half_clk 输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。

表 9 - 2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (3/6)

接口中的信号	接口类型	说明 / 如何连接
memory interface		
mem_a	导线管 (conduit)	PHY 与存储器件之间的接口信号。
mem_ba		
mem_ck		
mem_ck_n		
mem_cke		
mem_cs_n		
mem_dm		
mem_ras_n		
mem_cas_n		
mem_we_n		
mem_dq		
mem_dqs		
mem_dqs_n		
mem_odt		
mem_reset_n		
mem_ac_parity		
mem_err_out_n		
mem_parity_error_n		
avl interface		
avl_ready	Avalon-MM Slave	存储器接口与用户逻辑之间的 Avalon-MM 接口信号。
avl_burst_begin		
avl_addr		
avl_rdata_valid		
avl_rdata		
avl_wdata		
avl_be		
avl_read_req		
avl_write_req		
avl_size		
status interface		
local_init_done	导线管 (conduit)	存储器接口状态信号。
local_cal_success		
local_cal_fail		
oct interface		

表 9-2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (4/6)

接口中的信号	接口类型	说明 / 如何连接
rup (Stratix III/IV, Arria II GZ)	导线管 (conduit)	rup/rdn 或 rzqin 的 OCT 参考电阻管脚。
rdn (Stratix III/IV, Arria II GZ)		
rzq (Stratix V, Arria v, Cyclone V)		
local_powerdown interface		
local_powerdn_ack	导线管 (conduit)	仅当开启 Enable Auto Powerdown 时，控制器的这一断电接口才被使能。
pll_sharing interface		
pll_mem_clk	导线管 (conduit)	接口信号用于 PLL sharing 连接 PLL masters 到 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
pll_write_clk		
pll_addr_cmd_clk		
pll_locked		
pll_avl_clk		
pll_config_clk		
pll_hr_clk		
pll_p2c_read_clk		
pll_c2p_write_clk		
pll_dr_clk		
dll_sharing interface		
dll_delayctrl	导线管 (conduit)	DLL sharing 接口，用于连接 DLL masters 与 DLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
dll_pll_locked		
oct_sharing interface		
seriesterminationcontrol	导线管 (conduit)	OCT sharing 接口，用于连接 OCT masters 与 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol		
hcx_dll_reconfig interface		
dll_offset_ctrl_addnsub	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 DLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以使能 DLL 重配置。
dll_offset_ctrl_offset		
dll_offset_ctrl_addnsub ⁽¹⁾		
dll_offset_ctrl_offset ⁽¹⁾		
dll_offset_ctrl_offsetctrl out ⁽¹⁾		
dll_offset_ctrl_b_offsetctrlout ⁽¹⁾		

表 9 - 2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (5/6)

接口中的信号	接口类型	说明 / 如何连接
hcx_pll_reconfig interface		
configupdate	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 PLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以使能 PLL 重配置。
phasecounterselect		
phasestep		
phaseupdown		
scanclk		
scanclkena		
scandata		
phasedone		
scandataout		
scandone		
hcx_rom_reconfig interface		
hc_rom_config_clock	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 ROM 加载器接口才被使能。 您可以连接此接口到用户创建的定制逻辑，以控制定序器 ROM 的加载。
hc_rom_conig_datain		
hc_rom_config_rom_data_ready		
hc_rom_config_init		
hc_rom_config_init_busy		
hc_rom_config_rom_rden		
hc_rom_config_rom_address		
autoprecharge_req interface		
local_autopch_req	导线管 (conduit)	预加电接口，用于连接定制控制模块。仅当开启 Auto-precharge Control 时，此接口才被使能。
user_refresh interface		
local_refresh_req	导线管 (conduit)	用户刷新接口，用于连接定制控制模块。仅当开启 User Auto-Refresh Control 时，此接口才被使能。
local_refresh_chip		
local_refresh_ack		
self_refresh interface		
local_self_rfsh_req	导线管 (conduit)	自刷新接口，用于连接定制控制模块。仅当开启 Self-refresh Control 时，此接口才被使能。
local_self_rfsh_chip		
local_self_rfsh_ack		
ecc_interrupt interface		
ecc_interrupt	导线管 (conduit)	ECC 中断信号，用于连接定制控制模块。仅当开启 Error Detection and Correction Logic 时，此接口才被使能。

表 9 - 2. 具有 UniPHY 接口的 DDR3 SDRAM 控制器 (6/6)

接口中的信号	接口类型	说明 / 如何连接
csr interface		
csr_write_req	Avalon-MM Slave	配置和状态寄存器信号用于存储器接口连接 Avalon_MM master。仅当开启 Configuration and Status Register 时，此接口才被使能。
csr_read_req		
csr_waitrequest		
csr_addr		
csr_be		
csr_wdata		
csr_rdata		
csr_rdata_valid		
Hard Memory Controller MPFE FIFO Clock Interface		
mp_cmd_clk	导线管 (conduit)	使能 Hard Memory Interface 时，三个 FIFO 缓存 (command, read data 和 write data) 在 MPFE 中被创建。每个 FIFO 缓存有其各自的时钟和复位端口。仅当开启 Enable Hard Memory Interface 时，此接口才被使能。
mp_rfifo_clk		
mp_wfifo_clk		
mp_cmd_reset_n		
mp_rfifo_reset_n		
mp_wfifo_reset_n		
Hard Memory Controller Bonding Interface		
bonding_in_1	导线管 (conduit)	绑定接口用于绑定两个控制器以扩展带宽。仅当开启 Export Bonding Interface 时，此接口才被使能。
bonding_in_2		
bonding_in_3		
bonding_out_1		
bonding_out_2		
bonding_out_3		
表 9 - 2 注释:		
(1) 信号仅在 DLL master 模式中可用。		

表 9 - 3 列出了应用于 Qsys 和 SOPC Builder 中的每个接口的 LPDDR2 SDRAM 信号，并提供了连接这些接口的说明和指南。

表 9 - 3. 具有 UniPHY 接口的 LPDDR2 SDRAM 控制器 (1/5)

接口中的信号	接口类型	说明 / 如何连接
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	对 PLL 和 PHY 中的全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，控制器将置位此接口。

表 9 - 3. 具有 UniPHY 接口的 LPDDR2 SDRAM 控制器 (2/5)

接口中的信号	接口类型	说明 / 如何连接
afi_reset_in interface		
afi_reset_n	复位输入 (PLL slave)	当接口处于 PLL slave 模式中时，它是一个复位输入，您必须将该复位输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_reset 输出。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 当此接口处于 PLL slave 模式时，这是一个时钟输入，您必须将此 afi_clk 时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI half clock 是 afi_clk 频率的一半。当此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI half clock 是 afi_clk 频率的一半。当此接口处于 PLL slave 模式时，您必须连接此 afi_half_clk 输入到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。
memory interface		
mem_ca	导线管 (conduit)	PHY 与存储器之间的接口信号。
mem_ck		
mem_ck_n		
mem_cke		
mem_cs_n		
mem_dm		
mem_dq		
mem_dqs		
mem_dqs_n		
avl interface		

表 9-3. 具有 UniPHY 接口的 LPDDR2 SDRAM 控制器 (3/5)

接口中的信号	接口类型	说明 / 如何连接		
avl_ready	Avalon-MM Slave	存储器接口与用户逻辑之间的 Avalon-MM 接口信号。		
avl_burst_begin				
avl_addr				
avl_rdata_valid				
avl_rdata				
avl_wdata				
avl_be				
avl_read_req				
avl_write_req				
avl_size				
status interface				
local_init_done			导线管 (conduit)	存储器接口状态信号。
local_cal_success				
local_cal_fail				
oct interface				
rzq	导线管 (conduit)	rzqin 的 OCT 参考电阻管脚。		
local_powerdown interface				
local_powerdn_ack	导线管 (conduit)	仅当开启 Enable Auto Powerdown 时，控制器的这一断电接口才被使能。		
local_deep_powerdn interface				
local_deep_powerdn_ack	导线管 (conduit)	深度断电接口用于控制器使能深度断电。当开启 Enable Deep Power-Down Controls 时，此接口被使能。		
local_deep_powerdn_chip				
local_deep_powerdn_req				
pll_sharing interface				
pll_mem_clk	导线管 (conduit)	接口信号用于 PLL sharing 连接 PLL masters 到 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。		
pll_write_clk				
pll_addr_cmd_clk				
pll_locked				
pll_avl_clk				
pll_config_clk				
pll_mem_phy_clk				
afi_phy_clk				
pll_write_clk_pre_phy_clk				
dll_sharing interface				
dll_delayctrl	导线管 (conduit)	DLL sharing 接口用于连接 DLL masters 到 DLL slaves。仅当设置 DLL sharing mode 为 master 或 slave 时，此接口才被使能。		
dll_pll_locked				
oct_sharing interface				

表 9 - 3. 具有 UniPHY 接口的 LPDDR2 SDRAM 控制器 (4/5)

接口中的信号	接口类型	说明 / 如何连接
seriesterminationcontrol	导线管 (conduit)	OCT sharing 接口用于连接 OCT masters 到 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol		
autoprecharge_req interface		
local_autopch_req	导线管 (conduit)	预加电接口，用于连接定制控制模块。仅当开启 Auto-precharge Control 时，此接口才被使能。
user_refresh interface		
local_refresh_req	导线管 (conduit)	用户刷新接口，用于连接定制控制模块。仅当开启 User Auto-Refresh Control 时，此接口才被使能。
local_refresh_chip		
local_refresh_ack		
self_refresh interface		
local_self_rfsh_req	导线管 (conduit)	自刷新接口，用于连接定制控制模块。仅当开启 Self-refresh Control 时，此接口才被使能。
local_self_rfsh_chip		
local_self_rfsh_ack		
ecc_interrupt interface		
ecc_interrupt	导线管 (conduit)	ECC 中断信号，用于连接定制控制模块。仅当开启 Error Detection and Correction Logic 时，此接口才被使能。
csr interface		
csr_write_req	Avalon-MM Slave	配置和状态寄存器信号用于存储器接口连接 Avalon_MM master。仅当开启 Configuration and Status Register 时，此接口才被使能。
csr_read_req		
csr_waitrequest		
csr_addr		
csr_be		
csr_wdata		
csr_rdata		
csr_rdata_valid		
Local_rdata_error interface		
Local_rdata_error	导线管 (conduit)	Error Detection and Correction 逻辑使能时，此信号指示读数据错误。
Hard Memory Controller MPFE FIFO Clock Interface		
mp_cmd_clk	导线管 (conduit)	使能 Hard Memory Interface 时，三个 FIFO 缓存 (command, read data 和 write data) 在 MPFE 中被创建。每个 FIFO 缓存有其各自的时钟和复位端口。仅当开启 Enable Hard Memory Interface 时，此接口才被使能。
mp_rfifo_clk		
mp_wfifo_clk		
mp_cmd_reset_n		
mp_rfifo_reset_n		
mp_wfifo_reset_n		
Hard Memory Controller Bonding Interface		

表 9-3. 具有 UniPHY 接口的 LPDDR2 SDRAM 控制器 (5/5)

接口中的信号	接口类型	说明 / 如何连接
bonding_in_1	导线管 (conduit)	绑定接口用于绑定两个控制器以扩展带宽。仅当开启 Export Bonding Interface 时，此接口才被使能。
bonding_in_2		
bonding_in_3		
bonding_out_1		
bonding_out_2		
bonding_out_3		

表 9-4 列出了应用于 Qsys 和 SOPC Builder 中的每个接口的 QDR II 和 QDR II+ SRAM 信号，并提供了连接这些接口的说明和指南。

表 9-4. 具有 UniPHY 接口的 QDR II 和 QDR II+ SRAM 控制器 (1/4)

接口中的信号	接口类型	说明 / 如何连接
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	PLL 和 PHY 中的全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，控制器将置位此接口。
afi_reset_in interface		
afi_reset_n	复位输入 (PLL slave)	当接口处于 PLL slave 模式中时，它是一个复位输入，您必须将该复位输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_reset 输出。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 当此接口处于 PLL slave 模式时，这是一个时钟输入，您必须将此 afi_clk 时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		

表 9 - 4. 具有 UniPHY 接口的 QDR II 和 QDR II+ SRAM 控制器 (2/4)

接口中的信号	接口类型	说明 / 如何连接
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL slave 模式时，您必须将此 afi_half_clk 输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。
memory interface		
mem_a	导线管 (conduit)	PHY 与存储器之间的接口信号。
mem_cqn		
mem_bws_n		
mem_cq		
mem_d		
mem_k		
mem_k_n		
mem_q		
mem_wps_n		
mem_rps_n		
mem_doff_n		
avl_r interface		
avl_r_read_req	Avalon-MM Slave	读请求的存储器接口与用户逻辑之间的 Avalon-MM 接口信号。
avl_r_ready		
avl_r_addr		
avl_r_size		
avl_r_rdata_valid		
avl_r_rdata		
avl_w interface		
avl_w_write_req	Avalon-MM Slave	写请求的存储器接口与用户逻辑之间的 Avalon-MM 接口信号。
avl_w_ready		
avl_w_addr		
avl_w_size		
avl_w_wdata		
avl_w_be		
status interface		
local_init_done	导线管 (conduit)	存储器接口状态信号。
local_cal_success		
local_cal_fail		

表 9-4. 具有 UniPHY 接口的 QDR II 和 QDR II+ SRAM 控制器 (3/4)

接口中的信号	接口类型	说明 / 如何连接
oct interface		
rup (Stratix III/IV, Arria II GZ, Arria II GX)	导线管 (conduit)	rup/rdn 或 rzqin 的 OCT 参考电阻管脚。
rdn (Stratix III/IV, Arria II GZ, Arria II GX)		
rzq (Stratix V, Arria V, Cyclone V)		
pll_sharing interface		
pll_mem_clk	导线管 (conduit)	接口信号用于 PLL sharing 连接 PLL masters 到 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
pll_write_clk		
pll_addr_cmd_clk		
pll_locked		
pll_avl_clk		
pll_config_clk		
pll_hr_clk		
pll_p2c_read_clk		
pll_c2p_write_clk		
pll_dr_clk		
dll_sharing interface		
dll_delayctrl	导线管 (conduit)	DLL sharing 接口用于连接 DLL masters 到 DLL slaves。仅当设置 DLL sharing mode 为 master 或 slave 时，此接口才被使能。
dll_pll_locked		
oct_sharing interface		
seriesterminationcontrol (Stratix III/IV/V, Arria II GZ, Arria V, Cyclone V)	导线管 (conduit)	OCT sharing 接口用于连接 OCT masters 到 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol (Stratix III/IV/V, Arria II GZ, Arria V, Cyclone V)		
terminationcontrol (Arria II GX)		
hcx_dll_reconfig		
dll_offset_ctrl_addnsub	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 DLL 重配置接口才被使能。
dll_offset_ctrl_offset		
dll_offset_ctrl_addnsub ⁽¹⁾		
dll_offset_ctrl_offset ⁽¹⁾		
dll_offset_ctrl_offsetctrlout ⁽¹⁾		
dll_offset_ctrl_b_offsetctrlout ⁽¹⁾		

表 9 - 4. 具有 UniPHY 接口的 QDR II 和 QDR II+ SRAM 控制器 (4/4)

接口中的信号	接口类型	说明 / 如何连接
hcx_pll_reconfig		
configupdate	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 PLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑以使能 PLL 重配置。
phasecounterselect		
phasestep		
phaseupdown		
scanclk		
scanclkena		
scandata		
phasedone		
scandataout		
scandone		
hcx_rom_reconfig		
hc_rom_config_clock	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 ROM 加载器接口才被使能。 您可以连接此接口到用户创建的定制逻辑以控制定序器 ROM 的加载。
hc_rom_config_datain		
hc_rom_config_rom_data_read y		
hc_rom_config_init		
hc_rom_config_init_busy		
hc_rom_config_rom_rden		
hc_rom_config_rom_address		
表 9-4 注释:		
(1) 信号仅在 DLL master 模式中可用。		

表 9 - 5 列出了用于 Qsys 和 SOPC Builder 中的每个接口的 RLDRAM II 信号，并提供了连接这些接口的说明和指南。

表 9 - 5. 具有 UniPHY 接口的 RLDRAM II 控制器 (1/4)

接口名称	接口类型	说明
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	对 PLL 和 PHY 中的全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，控制器将置位此接口。
afi_reset_in interface		

表 9 - 5. 具有 UniPHY 接口的 RLDRAM II 控制器 (2/4)

接口名称	接口类型	说明
afi_reset_n	复位输入 (PLL slave)	当接口处于 PLL slave 模式中时，它是一个复位输入，您必须将该复位输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_reset 输出。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。 当此接口处于 PLL slave 模式时，这是一个时钟输入，您必须将此 afi_clk 时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI half clock 是 afi_clk 频率的一半。 当此接口处于 PLL slave 模式时，您必须将此 afi_half_clk 输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。
memory interface		
mem_a	导线管 (conduit)	PHY 与存储器之间的接口信号。
mem_ba		
mem_ck		
mem_ck_n		
mem_cs_n		
mem_dk		
mem_dk_n		
mem_dm		
mem_dq		
mem_qk		
mem_qk_n		
mem_ref_n		
mem_we_n		

表 9 - 5. 具有 UniPHY 接口的 RLDRAM II 控制器 (3/4)

接口名称	接口类型	说明
avl interface		
avl_size	Avalom-MM Slave	存储器接口与用户逻辑之间的 Avalon-MM 接口信号。
avl_wdata		
avl_rdata_valid		
avl_rdata		
avl_ready		
avl_write_req		
avl_read_req		
avl_addr		
status interface		
local_init_done	导线管 (conduit)	存储器接口状态信号。
local_cal_success		
local_cal_fail		
oct interface		
rup (Stratix III/IV, Arria II GZ)	导线管 (conduit)	rup/rdn 或 rzqin 的 OCT 参考电阻管脚。
rdn (Stratix III/IV, Arria II GZ)		
rzq (Stratix V)		
pll_sharing interface		
pll_mem_clk	导线管 (conduit)	接口信号用于 PLL sharing 连接 PLL masters 到 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
pll_write_clk		
pll_addr_cmd_clk		
pll_locked		
pll_avl_clk		
pll_config_clk		
pll_hr_clk		
pll_p2c_read_clk		
pll_c2p_write_clk		
pll_dr_clk		
dll_sharing interface		
dll_delayctrl	导线管 (conduit)	DLL sharing 接口用于连接 DLL masters 到 DLL slaves。仅当设置 DLL sharing mode 为 master 或 slave 时，此接口才被使能。
oct_sharing interface		
seriesterminationcontrol	导线管 (conduit)	OCT sharing 接口用于连接 OCT masters 到 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol		

表 9-5. 具有 UniPHY 接口的 RLDRAM II 控制器 (4/4)

接口名称	接口类型	说明
hcx_dll_reconfig interface		
dll_offset_ctrl_addnsub	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 DLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑以使能 DLL 重配置。
dll_offset_ctrl_offset		
dll_offset_ctrl_addnsub ⁽¹⁾		
dll_offset_ctrl_offset ⁽¹⁾		
dll_offset_ctrl_offsetctrlout ⁽¹⁾		
dll_offset_ctrl_b_offsetctrlout ⁽¹⁾		
hcx_pll_reconfig interface		
configupdate	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 PLL 重配置接口才被使能。 您可以连接此接口到用户创建的定制逻辑以使能 PLL 重配置。
phasecounterselect		
phasetstep		
phaseupdown		
scanclk		
scanclkena		
scandata		
phasedone		
scandataout		
scandone		
hcx_rom_reconfig interface		
hc_rom_config_clock	导线管 (conduit)	仅当开启 HardCopy Compatibility Mode 时，此 ROM 加载器接口才被使能。 您可以连接此接口到用户创建的定制逻辑以控制定序器 ROM 的加载。
hc_rom_config_datain		
hc_rom_config_rom_data_ready		
hc_rom_config_init		
hc_rom_config_init_busy		
hc_rom_config_rom_rden		
hc_rom_config_rom_adress		
parity_error_interrupt interface		
parity_error	导线管 (conduit)	奇偶校验错误中断管道用于连接到定制控制模块。只有开启 Enable Error Detection Parity 此接口才被使能。
user_refresh interface		
ref_req	导线管 (conduit)	用户刷新接口用于连接定制控制模块。只有开启 User Auto-Refresh Control 此接口才被使能。
ref_ba		
ref_ack		
reserved interface		
reserved	导线管 (conduit)	当选择基于 Nios® II 的定序器时某些管脚配置所需要的保留接口。
表 9-5 注释:		
(1) 信号仅在 DLL master 模式中可用。		

表 9-6 列出了用于 Qsys 和 SOPC Builder 中的每个接口的 RLD RAM 3 信号，并提供了连接这些接口的说明和指南。

表 9-6. RLD RAM 3 UniPHY 接口 (1/3)

接口中的信号	接口类型	说明 / 如何连接
pll_ref_clk interface		
pll_ref_clk	时钟输入	PLL 参考时钟输入。
global_reset interface		
global_reset_n	复位输入	对 PLL 和 PHY 中的全部逻辑的异步全局复位。
soft_reset interface		
soft_reset_n	复位输入	异步复位输入。复位 PHY，但不复位 PHY 使用的 PLL。
afi_reset interface		
afi_reset_n	复位输出 (PLL master/no sharing)	当接口处于 PLL master 或 no sharing 模式中时，此接口是 AFI 接口的一个异步复位输出。当 PLL 失锁或者 PHY 被复位时，控制器将置位此接口。
afi_reset_in interface		
afi_reset_n	复位输入 (PLL slave)	当接口处于 PLL slave 模式中时，它是一个复位输入，您必须在 PLL master 模式中将该复位输入连接到相同配置的存储器接口的 afi_reset 输出。
afi_clk interface		
afi_clk	时钟输出 (PLL master/no sharing)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_clk_in interface		
afi_clk	时钟输入 (PLL slave)	根据存储器接口参数，此 AFI 接口时钟可以是全速率或者半速率的存储器时钟频率。当此接口处于 PLL slave 模式时，这是一个时钟输入，您必须将此 afi_clk 时钟输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_clk 输出。
afi_half_clk interface		
afi_half_clk	时钟输出 (PLL master/no sharing)	AFI half clock 是 afi_clk 频率的一半。当此接口处于 PLL master 或者 no sharing 模式中时，它是一个时钟输出。
afi_half_clk_in interface		
afi_half_clk	时钟输入 (PLL slave)	AFI half clock 是 afi_clk 频率的一半。当此接口处于 PLL slave 模式时，您必须将此 afi_half_clk 输入连接到在 PLL master 模式中相同配置的存储器接口的 afi_half_clk 输出。
memory interface		

表 9 - 6. RLDRAM 3 UniPHY 接口 (2/3)


mem_a	管道 (Conduit)	PHY 与存储器件之间的接口信号。
mem_ba		
mem_ck		
mem_ck_n		
mem_cs_n		
mem_dk		
mem_dk_n		
mem_dm		
mem_dq		
mem_qk		
mem_qk_n		
mem_ref_n		
mem_we_n		
mem_reset_n		
afi interface		
afi_addr	Avalon-MM Slave	PHY 与控制器之间的 Altera PHY 接口 (AFI) 信号。
afi_ba		
afi_cs_n		
afi_we_n		
afi_ref_n		
afi_wdata_valid		
afi_wdata		
afi_dm		
afi_rdata		
afi_rdata_en		
afi_rdata_en_full		
afi_rdata_valid		
afi_rst_n		
afi_cal_success		
afi_cal_fail		
afi_wlat		
afi_rlat		
oct interface		
oct_rzqin	管道 (conduit)	rzqin 的 OCT 参考电阻管脚。
pll_sharing interface		

表 9 - 6. RLD RAM 3 UniPHY 接口 (3/3)

pll_mem_clk	管道 (conduit)	接口信号用于 PLL sharing 连接 PLL masters 到 PLL slaves。仅当设置 PLL sharing mode 为 master 或 slave 时，此接口才被使能。
pll_write_clk		
pll_addr_cmd_clk		
pll_locked		
pll_avl_clk		
pll_config_clk		
pll_mem_phy_clk		
afi_phy_clk		
pll_write_clk_pre_phy_clk		
pll_p2c_read_clk		
pll_c2p_write_clk		
dll_sharing interface		
dll_delayctrl	管道 (conduit)	DLL sharing 接口用于连接 DLL masters 到 DLL slaves。仅当设置 DLL sharing mode 为 master 或 slave 时，此接口才被使能。
dll_pll_locked		
oct_sharing interface		
seriesterminationcontrol	管道 (conduit)	OCT sharing 接口用于连接 OCT masters 到 OCT slaves。仅当设置 OCT sharing mode 为 master 或 slave 时，此接口才被使能。
parallelerminationcontrol		

生成的文件

完成 IP 生成流程后，在您的工程目录中会创建生成的文件。创建的目录结构在某种程度上会根据参数化和生成 IP 的工具而变化。

 在生成 IP 时间，PLL 参数静态定义在 `<variation_name>_parameters.tcl` 中。当编辑 PLL 参数时，需要将这些变化应用到该文件中的 PLL 参数，以确保时序约束和时序报告是正确的。

下面章节列出了 ALTMEMPHY 和 UniPHY IP 的生成文件。

基于 ALTMEMPHY IP 的存储控制器的生成文件

表 9 - 7 列出了使用 MegaWizard Plug-In Manager 生成的 ALTMEMPHY 目录和主要文件。

表 9 - 7. ALTMEMPHY 生成的文件 (1/3)

文件名	说明
alt_mem_phy_defines.v	包含接口中使用的常量。不管在 MegaWizard Plug-In Manager 中选择什么语言，此文件始终是 Verilog HDL。
<variation_name>.ppf	你的 ALTMEMPHY variation 的管脚布局文件
<variation_name>.qip	你的 ALTMEMPHY variation 的 Quartus II IP 文件，包含与此宏功能相关的文件。
<variation_name>.v/.vhd	你的 ALTMEMPHY variation 的顶层文件，根据在 MegaWizard Plug-In Manager 中选择的语言生成此文件。

表 9 - 7. ALTMEMPHY 生成的文件 (2/3)

文件名	说明
<code><variation_name>.vho</code>	仅包含 VHDL 的功能仿真模型。
<code><variation_name>_alt_mem_phy_seq_wrapper.vo/.vho</code>	仅用于仿真的包装文件 (wrapper file)，调用 sequencer 文件，根据在 MegaWizard Plug-In Manager 中选择的语言生成此文件。
<code><variation_name>.html</code>	列出了生成的顶层文件和在宏功能中使用的端口。
<code><variation_name>_alt_mem_phy_seq_wrapper.v/.vhd</code>	仅用于编译的包装文件 (wrapper file)，调用 sequencer 文件，根据在 MegaWizard Plug-In Manager 中选择的语言生成此文件。
<code><variation_name>_alt_mem_phy_seq.vhd</code>	包含校准期间使用的 sequencer。不管在 MegaWizard Plug-In Manager 中选择什么语言，此文件始终是 VHDL 语言。
<code><variation_name>_alt_mem_phy.v</code>	包含除 sequencer 之外的 ALTMEMPHY variation 的所有模块。不管在 MegaWizard Plug-In Manager 中选择什么语言，此文件始终是 VHDL 语言。 <code><variation_name>_alt_mem_phy_seq.vhd</code> 包括 DDR3 SDRAM sequencer。
<code><variation_name>_alt_mem_phy_pll_<device>.ppf</code>	此 XML 文件描述了 Quartus II Pin Planner 的 MegaCore 管脚属性。
<code><variation_name>_alt_mem_phy_pll.v/.vhd</code>	您的 ALTMEMPHY variation 的 PLL 宏功能文件，根据在 MegaWizard Plug-In Manager 中选择的语言生成此文件。
<code><variation_name>_alt_mem_phy_delay.vhd</code>	包括用于仿真的一个延迟模块。仅当选择 VHDL 作为您的 MegaWizard Plug-In Manager 输出文件的语言时才会生成此文件。
<code><variation_name>_alt_mem_phy_dq_dqs.vhd or .v</code>	包含 DQ/DQS I/O 组合电路单元互联和实例的生成文件。仅当使用 Arria II GX 目标器件时才会生成此文件。
<code><variation_name>_alt_mem_phy_dq_dqs_clearbox.txt</code>	规格文件，使用 clearbox 流程生成 <code><variation_name>_alt_mem_phy_dq_dqs</code> 文件。仅当使用 Arria II GX 目标器件时才会生成此文件。
<code><variation_name>_alt_mem_phy_pll.qip</code>	包含与此宏功能相关文件的 Quartus II IP 文件，用于您的 ALTMEMPHY variation 所使用的 PLL。
<code><variation_name>_alt_mem_phy_pll_bb.v/.cmp</code>	黑匣子 (black box) 文件，用于您的 ALTMEMPHY 中生成的 PLL。通常不使用。
<code><variation_name>_alt_mem_phy_reconfig.qip</code>	Quartus II IP 文件，用于 PLL 重配置模块。仅当使用 Arria GX, HardCopy® II, Stratix II 和 Stratix II GX 目标器件时才会生成此文件。
<code><variation_name>_alt_mem_phy_reconfig.v/.vhd</code>	PLL 重配置模块。仅当使用 Arria GX, HardCopy II, Stratix II 和 Stratix II GX 目标器件时才会生成此文件。
<code><variation_name>_alt_mem_phy_reconfig_bb.v/cmp</code>	黑匣子 (black box) 文件，用于 PLL 重配置模块。仅当使用 Arria GX, HardCopy II, Stratix II 和 Stratix II GX 目标器件时才会生成此文件。
<code><variation_name>_bb.v/.cmp</code>	黑匣子 (black box) 文件，用于您的 ALTMEMPHY variation，取决于是否使用 Verilog HDL 或 VHDL 语言。
<code><variation_name>_ddr_pins.tcl</code>	包含用于 <code><variation_name>_ddr_timing.sdc</code> 和 <code><variation_name>_report_timing.tcl</code> 文件中的流程 (procedure)。

表 9 - 7. ALTMEMPHY 生成的文件 (3/3)

文件名	说明
<code><variation_name>_pin_assignments.tcl</code>	包括您的 ALTMEMPHY variation 的 I/O 标准，驱动强度，输出使能组，DQ/DQS 组和匹配约束。如果您的顶层设计管脚名不匹配默认管脚名或者带前缀的管脚名，那么需要编辑此文件中的约束。
<code><variation_name>_ddr_timing.sdc</code>	包含您的 ALTMEMPHY variation 的时序约束。
<code><variation_name>_report_timing.tcl</code>	脚本文件，在编译期间报告您的 ALTMEMPHY variation 的时序。

表 9 - 8 列出了在 `<variation_name>_alt_mem_phy.v/.vhd` 文件中例化的模块。根据您的指定的存储器标准，一个特定的 ALTMEMPHY variation 可能使用，也可能不使这些模块。

表 9 - 8. `<variation_name>_alt_mem_phy.v` 文件中的模块

模块名	使用	说明
<code><variation_name>_alt_mem_phy_addr_cmd</code>	All ALTMEMPHY variations	创建地址和命令结构。
<code><variation_name>_alt_mem_phy_clk_reset</code>	All ALTMEMPHY variations	例化 PLL，DLL 和复位逻辑。
<code><variation_name>_alt_mem_phy_dp_io</code>	All ALTMEMPHY variations	生成 DQ，DQS，DM 和 QVLD I/O 管脚。
<code><variation_name>_alt_mem_phy_mimic</code>	DDR2/DDR SDRAM ALTMEMPHY variation	创建 DDR 和 DDR2 SDRAM PHY IP 的 VT 跟踪机制。
<code><variation_name>_alt_mem_phy_oct_delay</code>	DDR2/DDR SDRAM ALTMEMPHY variation，动态 OCT 使能时。	生成 OCT 信号的正确延迟和持续时间。
<code><variation_name>_alt_mem_phy_postamble</code>	DDR2/DDR SDRAM ALTMEMPHY variations	生成 DDR 和 DDR2 SDRAM PHY IP 的后同步信号使能和禁用方案。
<code><variation_name>_alt_mem_phy_read_dp</code>	All ALTMEMPHY variations（不用于 Stratix III 或 Stratix IV 器件）	通过读路径 FIFO 缓存得到 I/O 中的读数据，完成从重同步时钟到 PHY 时钟的传输。
<code><variation_name>_alt_mem_phy_read_dp_group</code>	DDR2/DDR SDRAM ALTMEMPHY variations（仅 Stratix III 和 Stratix IV 器件）	per DQS group 版本的 <code><variation_name>_alt_mem_phy_read_dp</code> 。
<code><variation_name>_alt_mem_phy_rdata_valid</code>	DDR2/DDR SDRAM ALTMEMPHY variations	生成 sequencer 和控制器的读数据有效信号。
<code><variation_name>_alt_mem_phy_seq_wrapper</code>	All ALTMEMPHY variations	生成 DDR 和 DDR2 SDRAM 的 sequencer。
<code><variation_name>_alt_mem_phy_write_dp</code>	All ALTMEMPHY variations	生成半速率到全速率 DDR 数据的解复用。
<code><variation_name>_alt_mem_phy_write_dp_fr</code>	DDR2/DDR SDRAM ALTMEMPHY variations	全速率版本的 <code><variation_name>_alt_mem_phy_write_dp</code> 。

表 9 - 9 列出了由 HPC II 生成的其它文件，位于您的工程目录中。

表 9 - 9. 控制器生成的文件

文件名	说明
<code>alt_mem_ddrx_addr_cmd.v</code>	将内部协议相关信号解码成存储器地址和命令信号。
<code>alt_mem_ddrx_addr_cmd_wrap.v</code>	例化 <code>alt_mem_ddrx_addr_cmd.v</code> 文件的 wrapper。
<code>alt_mem_ddrx_ddr2_odt_gen.v</code>	生成 DDR2 存储器接口的片上匹配 (ODT) 控制信号。
<code>alt_mem_ddrx_ddr3_odt_gen.v</code>	生成 DDR3 存储器接口的片上匹配 (ODT) 控制信号。
<code>alt_mem_ddrx_odt_gen.v</code>	例化 <code>alt_mem_ddrx_ddr2_odt_gen.v</code> 和 <code>alt_mem_ddrx_ddr3_odt_gen.v</code> 文件的 wrapper。此文件也控制 ODT 寻址方案。
<code>alt_mem_ddrx_rdwr_data_tmg.v</code>	将内部数据突发相关信号解码成存储器数据信号。
<code>alt_mem_ddrx_arbiter.v</code>	包含根据指定方案决定执行哪条命令的逻辑。
<code>alt_mem_ddrx_burst_gen.v</code>	将内部 DRAM-aware 命令转换成 AFI 信号。
<code>alt_mem_ddrx_cmd_gen.v</code>	将用户请求转换成 DRAM-aware 命令。
<code>alt_mem_ddrx_csr.v</code>	包含配置寄存器。
<code>alt_mem_ddrx_buffer.v</code>	包含本地数据的缓存。
<code>alt_mem_ddrx_buffer_manager.v</code>	管理缓存的分配。
<code>alt_mem_ddrx_burst_tracking.v</code>	跟踪每个本地突发命令的接收数据。
<code>alt_mem_ddrx_dataid_manager.v</code>	管理与存储在缓存中数据相关联的 ID。
<code>alt_mem_ddrx_fifo.v</code>	包含 FIFO 缓存，存储本地数据以生成链路；也用在 <code>rdata_path</code> 中存储读地址和错误地址。
<code>alt_mem_ddrx_list.v</code>	跟踪与内部存储的数据相关联的 DRAM 命令。
<code>alt_mem_ddrx_rdata_path.v</code>	包含读数据通路逻辑。
<code>alt_mem_ddrx_wdata_path.v</code>	包含写数据通路逻辑。
<code>alt_mem_ddrx_define.v</code>	定义在 RTL 文件中使用的通用参数。
<code>alt_mem_ddrx_ecc_decoder.v</code>	例化相应宽度的 ECC 解码器逻辑。
<code>alt_mem_ddrx_ecc_decoder_32_syn.v</code>	包含可综合的 32-bit ECC 解码器。
<code>alt_mem_ddrx_ecc_decoder_64_syn.v</code>	包含可综合的 64-bit ECC 解码器。
<code>alt_mem_ddrx_ecc_encoder.v</code>	例化相应宽度的 ECC 编码器逻辑。
<code>alt_mem_ddrx_ecc_encoder_32_syn.v</code>	包含可综合的 32-bit ECC 解码器。
<code>alt_mem_ddrx_ecc_encoder_64_syn.v</code>	包含可综合的 64-bit ECC 解码器。
<code>alt_mem_ddrx_ecc_encoder_decoder_wrapper.v</code>	例化所有 ECC 逻辑的 wrapper。
<code>alt_mem_ddrx_input_if.v</code>	包含本地输入接口逻辑。
<code>alt_mem_ddrx_mm_st_converter.v</code>	包含 Avalon-MM 接口的支持逻辑。
<code>alt_mem_ddrx_rank_timer.v</code>	包含一个关联等级时序的计时器。
<code>alt_mem_ddrx_sideband.v</code>	包含用户控制的刷新和预加电信号的支持逻辑。
<code>alt_mem_ddrx_tbp.v</code>	包含重排序 (reordering) 的命令队列和相关逻辑。
<code>alt_mem_ddrx_timing_param.v</code>	包含与 non-rank 时序关联的计时器逻辑。
<code>alt_mem_ddrx_controller_st_top.v</code>	例化所有子模块和配置寄存器的 wrapper。
<code>alt_mem_ddrx_controller_top.v</code>	包含具有 Avalon-MM 接口的存储控制器的 wrapper。
<code>alt_mem_ddrx_controller.v</code>	例化所有子模块的包装器 wrapper。

基于 UniPHY IP 的存储控制器生成的文件

表 9 - 10 列出了通过 MegaWizard Plug-In Manager, SOPC Builder 和 Qsys 生成的目录结构和主要文件。

表 9 - 10. 生成的目录结构和主要文件 (1/2)

目录	文件名	说明	
MegaWizard Plug-In Manager			
综合文件	<working_dir>/	<variation_name>.qip	Quartus II IP 文件 —— 在综合文件集中生成的全部文件。在您的 Quartus II 工程中包含此文件。
	<working_dir>/	<variation_name>.v 或 <variation_name>.vhd	顶层综合文件。 .v 是 IEEE Encrypted Verilog。 .vhd 是生成的 VHDL。
	<working_dir>/<variation_name>/	<variation_name>_0002.v	UniPHY 顶层文件。
	<working_dir>/<variation_name>/	*.v, *.sv, *.tcl, *.sdc, *.ppf	用于综合 RTL 和约束文件。
	<working_dir>/<variation_name>/	<variation_name>_p0_pin_assignments.tcl	综合后运行的管脚约束脚本。
仿真文件	<working_dir>/<variation_name>_sim/	<variation_name>.v	顶层仿真文件，用于 Verilog 以及 VHDL。
	<working_dir>/<variation_name>_sim/<subcomponent_module>/	*.v, *.sv, *.vhd, *.vho,*.hex, *.mif	RTL 和约束文件，用于仿真。 .v 和 .sv 文件是 IEEE Encrypted Verilog。 .vhd 和 .vho 是生成的 VHDL。
MegaWizard Plug-In Manager—Example Design Fileset			
综合文件	<variation_name>_example_design/example_project/	<variation_name>_example.qip	Quartus II IP 文件，是指在可综合工程中生成的全部文件。
	<variation_name>_example_design/example_project/	<variation_name>_example.qpf	Quartus II 工程，用于综合流程。
	<variation_name>_example_design/example_project/	<variation_name>_example.qsf	Quartus II 工程，用于综合流程。
	<variation_name>_example_design/example_project/ <variation_name>_example/	<variation_name>_example.v	顶层文件。
	<variation_name>_example_design/example_project/ <variation_name>_example/submodules/	*.v, *.sv, *.tcl, *.sdc, *.ppf	RTL 和约束文件。
	<variation_name>_example_design/example_project/ <variation_name>_example/submodules/	<variation_name>_example_if0_p0_pin_assignments.tcl	综合后运行的管脚约束脚本。 _if0 和 _p0 是实例名。关于更多信息，请参考第 9 - 39 页表 9 - 11。

表 9 - 10. 生成的目录结构和主要文件 (2/2)

目录		文件名	说明
仿真文件	<variation_name>_example_design/simulation/	generate_sim_verilog_example_design.tcl	运行此文件以生成 Verilog 仿真实例设计。
	<variation_name>_example_design/simulation/	generate_sim_vhdl_example_design.tcl	运行此文件以生成 VHDL 仿真实例设计。
	<variation_name>_example_design/simulation/	README.txt	文本文件，包括如何生成和运行仿真实例设计的说明。
	<variation_name>_example_design/simulation/verilog/mentor	run.do	ModelSim 脚本，仿真生成的 Verilog 实例设计。
	<variation_name>_example_design/simulation/vhdl/mentor	run.do	ModelSim 脚本，仿真生成的 VHDL 实例设计。
	<variation_name>_example_design/simulation/verilog/<variation_name>_sim/	<variation_name>_example_sim.v	顶层包装器（测试台），用于 Verilog。
	<variation_name>_example_design/simulation/vhdl/<variation_name>_sim/	<variation_name>_example_sim.vhd	顶层包装器（测试台），用于 VHDL。
	<variation_name>_example_design/simulation/<variation_name>_sim/verilog/submodules/	*.v, *.sv, *.hex, *.mif	RTL 和 ROM 数据，用于 Verilog。
	<variation_name>_example_design/simulation/<variation_name>_sim/vhdl/submodules/	*.vhd, *.vho, *.hex, *.mif	RTL 和 ROM 数据，用于 VHDL。
SOPC Builder			
<working_dir>/		<system_name>.qip	Quartus II IP 文件——在 SOPC Builder 工程中生成的全部文件。
<working_dir>/		<system_name>.v	系统顶层 RTL。
<working_dir>/		<module_name>.v	模块包装器 RTL。
<working_dir>/<module_name>/		*.v, *.sv, *.tcl, *.sdc, *.ppf	每个系统模块的 TL 和约束的子目录。
Qsys			
<working_dir>/<system_name>/synthesis/		<system_name>.qip	Quartus II IP 文件，是指在综合文件集中生成的全部文件。
<working_dir>/<system_name>/synthesis/		<system_name>.v	系统顶层 RTL，用于综合。
<working_dir>/<system_name>/simulation/		<system_name>.v or <variation_name>.vhd	系统顶层 RTL，用于仿真。 .v 文件是 IEEE Encrypted Verilog。 .vhd 文件是生成的 VHDL。
<working_dir>/<system_name>/synthesis/submodules/		*.v, *.sv, *.tcl, *.sdc, *.ppf	RTL 和约束文件，用于综合。
<working_dir>/<system_name>/simulation/submodules/		*.v, *.sv, *.hex, *.mif	RTL 和 ROM 数据，用于仿真。

表 9 - 11 列出了存储器接口 IP 中子模块文件的前缀或实例名。这些实例结合在一起形成唯一的综合和仿真文件名。

表 9 - 11. 子模块文件的前缀

前缀	说明
_c0	指定控制器。
_d0	指定驱动器或流量生成器。
_dll0	指定 DLL。
_e0	指定实例设计。
_if0	指定存储器接口。
_m0	指定 AFI 复用器。
_oct0	指定 OCT。
_p0	指定 PHY。
_pll0	指定 PLL。
_s0	指定定序器。
_t0	指定流量生成器状态检查器。

参数化基于 ALTMEMPHY IP 的存储控制器

本章节介绍了具有 ALTMEMPHY IP 的 DDR，DDR2 和 DDR3 SDRAM 存储控制器的可设置参数。

ALTMEMPHY 参数编辑器中的 **Parameter Settings** 页面使您能够参数化以下设置：

- 存储器设置
- PHY 设置
- 电路板设置

MegaWizard Plug-In Manager 底部的文本窗口显示关于存储器接口，警告和错误信息。窗口中显示的所有错误被纠正之前，**Finish** 按钮是禁用的。

下面章节详细介绍了 **Parameter Settings** 页面的四个标签。

存储器设置

通过使用此标签来应用您的存储器制造商的数据表中的存储器参数。


表 9 - 12 介绍了 ALTMEMPHY 参数编辑器的 **Memory Settings** 页面上的 **General Settings**。

表 9 - 12. General Settings (1/2)

参数名	说明
Device family	目标器件系列（例如，Arria II GX）。这里所选择的器件系列一定要匹配参数编辑器的 2a 页中选择的器件系列。要了解关于选择器件系列的详细信息，请参考 <i>External Memory Interface Handbook</i> 的 <i>Selecting your FPGA Device</i> 章节中的“Device Family Selection”部分。
Speed grade	选择特定的器件速度等级（例如：2，3 或 4，对于 Arria II GX 器件系列）。

表 9 - 12. General Settings (2/2)

参数名	说明
PLL reference clock frequency	确定 PLL 的外部输入时钟的时钟频率。如果频率不是一个约整数，那么要保留小数点后三位（例如：166.667 MHz 或者 100 MHz）以避免功能仿真或 PLL 锁定问题。
Memory clock frequency	确定存储器接口时钟频率。如果存储器件运行在其可达到的最大频率以下，那么要确保输入一个实际的操作频率，而不是存储器件可达到的最大频率。此外，如果频率不是一个约整数，那么要保留小数点后三位（例如：333.333 MHz 或者 400 MHz）以避免功能仿真或 PLL 锁定问题。
Controller data rate	选择存储控制器的数据速率。设置控制器频率等于存储器接口频率（全速率），或者等于存储器接口频率的一半（半速率）。全速率选项不适用于 DDR3 SDRAM 器件。
Enable half rate bridge	此选项仅用于 HPC II 全速控制器。 启用此选项，在允许用户端运行在半倍存储器时钟速度的同时，保持控制器处于存储器全时钟域，从而减少延迟。
Local interface clock frequency	取决于存储器时钟频率和控制器数据速率的值。
Local interface width	取决于存储器时钟频率和控制器数据速率的值。

 当采用高性能的 HardCopy 器件移植时，ALTMEMORY IP 应该使用中间速度等级以确保 PLL 与 PHY sequencer 设置匹配。设计的编译能够在更快的速度等级下执行。

在 ‘Memory Preset’ 列表中显示

表 9 - 13 描述了用于过滤显示的 Memory Presets 的选项。在这组选项中，您可以指示是否创建 DDR3 SDRAM 的路径。

表 9 - 13. 在 ‘Memory Presets’ 列表中显示

参数名	说明
Memory type (存储器类型)	您可以过滤要显示的存储器类型，例如：DDR3 SDRAM。
Memory vendor (存储器供应商)	您可以根据供应商过滤存储器类型。JEDEC 也是其中的一个选项，使您能够选择 JEDEC 规范。如果所选择的供应商没有列出来，那么可以选择 JEDEC 用于 DDR3 SDRAM 接口。然后，选择一个与所选器件规格类似的器件，并检查每个参数值。确保修改每个参数值以匹配您的器件规格。
Memory format (存储器格式)	您可以根据格式过滤存储器类型，例如：分立器件或 DIMM 封装。
Maximum frequency (最大频率)	您可以根据最大操作频率过滤存储器类型。

存储器预置 (Memory Preset)


在 Memory Presets 列表选择一个与您使用的实际存储器件最接近或者相同的器件。然后点击 **Modify Parameters** 按钮对 Preset Editor 对话框中的以下设计进行参数化：

- 存储器属性 — 这些设置决定您系统的 DQ, DQ 选通 (DQS), 地址和存储器时钟管脚数量。
- 存储器初始化选项 — 作为初始化进程的一部分，这些设置存储在存储器模式寄存器中。
- 存储器时序参数 — 这些是创建和时序约束 PHY 的参数。

 尽管您使用的器件已列在 **Memory Presets** 中，但也要保证 **Preset Editor** 对话框中的设置是准确的，因为在存储器件数据表中可能已经更新了某些参数。

您可以通过修改白色背景的参数来反映您的系统。您也可以修改灰色背景的参数，使器件参数与您使用的器件相匹配。这些灰色背景的参数是所选存储器件的特征，通过修改这些参数可以创建新的定制存储器预置。如果点击 **Save As**（页面的左下角）将新的设置保存在 `<quartus_install_dir>\quartus\common\ip\altera\altmemphy\lib\` 目录中，那么您能够在使用相同软件版本创建的其它 Quartus II 工程中使用这一新的存储器预置。

当点击 **Save**，新的存储器预置出现在 **Memory Settings** 标签中 **Memory Presets** 列表的底部。

 如果将新的设置存储在默认目录以外的目录中，则需要在点击 **Memory Settings** 标签中的 **Load Preset**，以加载设置到 **Memory Presets** 列表中。

Advanced 选项显示由 FPGA 校准的存储器规格的百分比。Altera 根据工艺变化估算百分比的值。

DDR 和 DDR2 SDRAM 的预置编辑器设置

表 9 - 14 到表 9 - 16 描述了 DDR SDRAM 参数，用于存储器属性，初始化选项和时序参数。DDR SDRAM 具有相同的参数，但参数值的范围不同于 DDR2 SDRAM。

表 9 - 14. DDR SDRAM 属性设置 (1/2)

参数名	范围 ⁽¹⁾	单位	说明
Output clock pairs from FPGA	1 - 6	pairs	定义从 FPGA 驱动到存储器的差分时钟对的数量。与多器件连接时，较多的时钟对可以降低每个输出的加载。存储器时钟管脚将 Arria II GX, Stratix III 和 Stratix IV 器件中的信号分离器功能用于差分信号。
Total Memory chip selects	1, 2, 4, or 8	bits	设置您存储器接口中的芯片选择的数量。芯片选择的数量定义您的存储器深度。用户端对芯片选择地址进行二进制编码的范围是受限的。如果此范围不满足您的规范，那么您可以设置此值为下一个更大值。然而，ALTMEMPHY 宏功能的最大地址空间没有映射到任何的 实际存储器地址。ALTMEMPHY 宏功能能够支持多片选，基于所有片选对 mem_cs_n 信号进行校准。
Memory interface DQ width	4 - 288	bits	定义存储器接口上的 DQ 管脚总数。如果与多个器件连接，DQ 管脚总数等于器件数乘以每个器件的 DQ 管脚数。尽管 GUI 支持选择 288-bit DQ 位宽，但接口数据位宽也受限于器件上的管脚数。要达到最佳性能，需要将整个接口布局在器件的一侧上。
Memory vendor	JEDEC, Micron, Qimonda, Samsung, Hynix, Elpida, Nanya, other	—	对于所有支持的存储器标准，列出了存储器供应商名。

表 9 - 14. DDR2 SDRAM 属性设置 (2/2)

参数名	范围 ⁽¹⁾	单位	说明
Memory format	Discrete Device, Unbuffered DIMM, Registered DIMM	—	指定和器件连接还是和模块连接。SODIMM 被未缓存的或寄存的 DIMM 支持。
Maximum memory frequency	参见存储器件数据表	MHz	设置存储器所支持的最大频率。
Column address width	9 - 11	bits	定义您的器件列地址位的数量。
Row address width	13 - 16	bits	定义您的器件行地址位的数量。
Bank address width	2 or 3	bits	定义您的器件块 (bank) 地址位的数量。
Chip selects per DIMM	1 or 2	bits	定义您的器件中每个 DIMM 上的芯片选择的数量。
DQ bits per DQS bit	4 or 8	bits	定义每个数据随路时钟 (DQS) 管脚的数据 (DQ) 位的数量。
Precharge address bit	8 or 10	bits	选择地址总线位，用作预加电地址位。
Drive DM pins from FPGA	Yes or No	—	指定是否将 DM 管脚用于写操作。Altera 器件不支持 ×4 模式的 DM 管脚。
Maximum memory frequency for CAS latency 3.0	80 - 533	MHz	对于给定的 CAS 延迟，定义存储器数据表中的频率界限。如果操作频率超过这个数，ALTMEMPHY parameter editor 会生成一个警告信息。
Maximum memory frequency for CAS latency 4.0			
Maximum memory frequency for CAS latency 5.0			
Maximum memory frequency for CAS latency 6.0			

表 9 - 14 注释：

(1) 范围值取决于所使用的实际存储器件。

表 9 - 15. DDR2 SDRAM 初始化选项

参数名	范围	单位	说明
Memory burst length	4 or 8	beats	设置每个传输读取或写入字的数量。 存储器突发长度为 4 等同于半速率设计中本地突发长度为 1，也等同于全速率设计中本地突发长度为 2。
Memory burst ordering	Sequential or Interleaved	—	控制读传输中数据在存储器与 FPGA 之间的传输顺序。关于详细信息，请参考存储器件数据表。
Enable the DLL in the memory devices	Yes or No	—	当设成 Yes 时，使能存储器件中的 DLL。您必须始终使能存储器件中的 DLL，因为 DLL 禁用时 Altera 不能保证任何的 ALTMEMPHY 操作。当 DLL 禁用时存储器件中的全部时序都无效。
Memory drive strength setting	Normal or Reduced	—	控制存储器件的输出缓存的驱动强度。不是所有的器件都支持降低的驱动强度。默认选项是 normal。
Memory ODT setting	Disabled, 50, 75, 150	W	设置存储器 ODT 值。在 DDR SDRAM 接口中不可用。
Memory CAS latency setting	3, 4, 5, 6	cycles	设置从读命令到存储器第一个输出数据的延迟，以时钟周期为单位。

表 9 - 16. DDR2 SDRAM 时序参数设置 ⁽¹⁾(1/2)

参数名	范围	单位	说明
t_{INIT}	0.001 - 1000	μs	最小的存储器初始化时间。复位后，控制器在此期间不发送任何命令到存储器。
t_{MRD}	2 - 39	ns	最小的加载模式命令持续时间。控制器在发送加载模式寄存器命令之后和发送其它任何命令之前等待的时间。 根据 Micron 的器件数据表中的 t_{CK} 周期， t_{MRD} 在 DDR2 SDRAM 高性能控制器中指定为单位 ns。通过将数据表中指定的周期数乘以 t_{CK} 来实现从 t_{MRD} 到 ns 的转换，其中 t_{CK} 是存储器操作频率，不是存储器件的 t_{CK} 。
t_{RAS}	8 - 200	ns	最小的 active to precharge 时间。控制器在发送一个 active 命令之后和发送一个 precharge 命令到同一 bank 之前等待的时间。
t_{RCD}	4 - 65	ns	最小的 active to read-write 时间。控制器在发送一个 active 命令后，在此期间不发送 read 或 write 命令到 bank。
t_{RP}	4 - 65	ns	最小的 precharge 命令持续时间。控制器在发送一个 precharge 命令后在此期间不访问 bank。
t_{REFI}	1 - 65534	μs	refresh 命令之间的最大间隔时间。除非启用户控制的刷新，否则控制器在此间隔执行常规刷新。
t_{RFC}	14 - 1651	ns	最短的 auto-refresh 命令持续时间。auto-refresh 命令发出后，控制器在运行前等待的时间。
t_{WR}	4 - 65	ns	最短写恢复时间。发出 precharge 命令前，写传输结束后控制器等待的时间。
t_{WTR}	1 - 3	t_{CK}	最短的 write-to-read 命令延时。在发出下一个 read 命令到同一 bank 中之前，write 命令结束后控制器等待的时间。该时序参数以时钟周期为单位，值取到下一个整数。
t_{AC}	300 - 750	ps	从 CK/CK# 信号的 DQ 输出存取时间。
t_{DQSCK}	100 - 750	ps	从 CK/CK# 信号的 DQS 输出存取时间。
t_{DQSQ}	100 - 500	ps	DQS 到 DQ 的最大 skew；DQS 到最后 DQ 有效，每组，每次存取。
t_{DQSS}	0 - 0.3	t_{CK}	正 DQS 锁存边沿到关联时钟边沿。
t_{DS}	10 - 600	ps	DQ 和 DM 相对于 DQS 的输入建立时间，它有一个降额值，取决于 DQS 的斜率（对于 DDR 以及 DDR2 SDRAM 接口）和 DQS 是单端还是差分（对于 DDR2 SDRAM 接口。一定要使用正确数值，输入的值要关系到 $V_{\text{REF}}(\text{dc})$ ，而不是 $V_{\text{IH}}(\text{ac}) \text{ min}$ 或 $V_{\text{IL}}(\text{ac}) \text{ max}$ 关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{DH}	10 - 600	ps	DQ 和 DM 相对于 DQS 的输入保持时间，它有一个降额值，取决于 DQS 的斜率（对于 DDR 以及 DDR2 SDRAM 接口）和 DQS 是单端还是差分（对于 DDR2 SDRAM 接口。一定要使用正确数值，输入的值要关系到 $V_{\text{REF}}(\text{dc})$ ，而不是 $V_{\text{IH}}(\text{ac}) \text{ min}$ 或 $V_{\text{IL}}(\text{ac}) \text{ max}$ 关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{DSH}	0.1 - 0.5	t_{CK}	从 CK 的 DQS 下降沿保持时间。
t_{DSS}	0.1 - 0.5	t_{CK}	到 CK 的 DQS 下降沿建立时间。

表 9 - 16. DDR2 SDRAM 时序参数设置 ⁽¹⁾ (2/2)

参数名	范围	单位	说明
t_{IH}	100 - 1000	ps	地址和控制输入保持时间，它具有一个降额值，取决于 CK 和 CK# 时钟的斜率和地址和命令信号。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{IS}	100 - 1000	ps	地址和控制输入建立时间，它具有一个降额值，取决于 CK 和 CK# 时钟的斜率和地址和命令信号。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{QHS}	100 - 700	ps	最大数据保持 skew 因子。
t_{RRD}	2.06 - 64	ns	activate 到 activate 时间，每个器件，RAS 到 RAS 延迟时序参数。
t_{FAW}	7.69 - 256	ns	four-activate window 时间，每个器件。
t_{RTP}	2.06 - 64	ns	Read 到 precharge 时间。

表 9 - 16 注释：

- (1) 关于参数范围的相关信息，请参考存储器数据表。一些参数以时钟周期 (t_{CK}) 为单位。如果 MegaWizard Plug-In Manager 要求以时间 (ps 或 ns) 为单位输入值，那么需要将此值与您接口的时钟周期相乘（不是存储器数据表中列出的最大时钟周期）。

DDR3 SDRAM 的预置编辑器设置

表 9 - 17 到表 9 - 19 描述了用于存储器属性、初始化选项和时序参数的 DDR3 SDRAM 参数。

表 9 - 17. DDR3 SDRAM 属性设置 (1/2)

参数名	范围 ⁽¹⁾	单位	说明
Output clock pairs from FPGA	1 - 6	pairs	定义从 FPGA 到存储器驱动的差分时钟对的数量。存储器时钟管脚对差分信号使用 Arria II GX 器件中的信号分离 (signal splitter) 功能。 ALTMEMPHY 参数编辑器在窗口底部显示一个错误，如果对 DDR3 SDRAM 选择多个参数，则 ALTMEMPHY 参数编辑器在窗口底部显示一个错误。
Total Memory chip selects	1, 2, 4, or 8	bits	设置存储器接口中芯片选择的数量。芯片选择的数量定义了您存储器的深度。用户端对芯片选择地址进行二进制编码的范围是受限的。
Memory interface DQ width	4 - 288	bits	定义存储器接口上的 DQ 管脚总数。如果连接多个器件，那么需要将器件的数量乘以每个器件的 DQ 管脚数。尽管 GUI 支持选择 288-bit DQ 位宽，但由于电路板布局的限制（影响较高数据位宽上的时序），DDR3 SDRAM variation 仅支持到 80-bit 位宽。此外，接口数据位宽被器件上的管脚数限制。要获得最佳性能，需要将整个接口布置在器件的一侧上。
Mirror addressing	—	—	在多 rank (multiple rank) DDR3 SDRAM DIMM 上，地址信号被分别布线到每个 rank 参考 JEDEC 规范中的地址镜像。 在此区域输入地址镜像的 rank 片选。每个片选对应一个 bit。例如，对于四个芯片选择，输入 1011 来镜像芯片选择 #3, #1 和 #0 上的地址。

表 9 - 17. DDR3 SDRAM 属性设置 (2/2)

参数名	范围 (1)	单位	说明
Memory vendor	Elpida, JEDEC, Micron, Samsung, Hynix, Nanya, other	—	列出了所有支持的存储器标准的存储器供应商。
Memory format	Discrete Device	—	Arria II GX 器件仅支持不带整平的 DDR3 SDRAM 器件，例如： Discrete Device 存储器格式。
Maximum memory frequency	参见存储器件 数据表	MHz	设置存储器所支持的最大频率。
Column address width	10 - 12	bits	定义您接口的列地址比特数量。
Row address width	12 - 16	bits	定义您接口的行地址比特的数量。如果你 DDR3 SDRAM 器件的行地址总线是 12-bit 宽，那么在顶层文件中设置行地址位宽为 13 ，并将第 13 个 bit 设为逻辑电平低（或者不要使第 13 个 bit 与存储器件连接）。
Bank address width	3	bits	定义您接口 bank 地址比特的数量。
Chip selects per device	1 or 2	bits	定义您接口中每个器件上芯片选择的数量。当前所有 rank 可以完成校准，但只能使用一个 rank 执行时序分析。
DQ bits per DQS bit	4 or 8	bits	定义每个数据选通 (DQS) 管脚的数据 (DQ) 比特的数量。
Drive DM pins from FPGA	Yes or No	—	指定是否将 DM 管脚用于写操作。Altera 器件不支持 ×4 模式的 DM 管脚。
Maximum memory frequency for CAS latency 5.0	80 - 700	MHz	对给定 CAS 延迟指定存储器数据表中的频率限制。如果你选择的 CAS 延迟的操作频率超过该值，那么 ALTMEMPHY MegaWizard Plug-In Manager 会生成一个警告信息。DDR3 SDRAM 器件所支持的最低频率为 300 MHz。
Maximum memory frequency for CAS latency 6.0			
Maximum memory frequency for CAS latency 7.0			
Maximum memory frequency for CAS latency 8.0			
Maximum memory frequency for CAS latency 9.0			
Maximum memory frequency for CAS latency 10.0			

表 9 - 17 注释：

(1) 范围值取决于所使用的实际存储器件。

表 9 - 18. DDR3 SDRAM 初始化选项 (1/3)

参数名	范围	单位	说明
Memory burst length	4, 8, on-the-fly	beats	设置每次传输读写的字数。
Memory burst ordering	Sequential or Interleaved	—	控制读传输期间存储器与 FPGA 之间数据传输的顺序。关于详细信息，请参考存储器件数据表。
DLL precharge power down	Fast exit or Slow exit	—	当 CKE 禁用时，设置模式寄存器设置以禁用 (Slow exit) 或使能 (Fast exit) 存储器 DLL。

表 9 - 18. DDR3 SDRAM 初始化选项 (2/3)

参数名	范围	单位	说明
Enable the DLL in the memory devices	Yes or No	—	设为 Yes 时，使能存储器件中的 DLL。因为当 DLL 关闭时，Altera 不能保证任何的 ALTMEMPHY 操作，因此您必须始终使能存储器件中的 DLL。DLL 关闭时存储器件的所有时序都是无效的。
ODT Rtt nominal value	ODT disable, RZQ/4, RZQ/2, RZQ/6	W	DDR3 SDRAM 接口中的 RZQ 被设成 240 Ω 。将片上匹配 (ODT) 值设为 60 Ω (RZQ/4), 120 Ω (RZQ/2), 或者 40 Ω (RZQ/6)。如果不打算使用 ODT，那么要将此参数值设为 ODT disable 。对于 single-ranked DIMM，将此值设置成 RZQ/4。
Dynamic ODT (Rtt_WR) value	Dynamic ODT off, RZQ/4, RZQ/2	W	DDR3 SDRAM 接口中的 RZQ 被设成 240 Ω 。将写操作期间的存储器 ODT 值设为 60 Ω (RZQ/4) 或者 120 Ω (RZQ/2)。由于 ALTMEMPHY 仅支持 single rank DIMM，因此不需要此选项 (设成 Dynamic ODT off)。
Output driver impedance	RZQ/6 (Reserved) or RZQ/7	W	DDR3 SDRAM 接口中的 RZQ 被设成 240 Ω 。设置存储器件的输出驱动器阻抗。某些器件可能没有 RZQ/6 选项。在选择此选项前，一定要查看存储器件数据表。
Memory CAS latency setting	5.0, 6.0, 7.0, 8.0, 9.0, 10.0	cycles	设置从读命令到存储器件的第一个输出数据的延迟，以时钟周期表示。
Memory additive CAS latency setting	Disable, CL - 1, CL - 2	cycles	使您能够在 CAS 延迟设置的基础上添加额外延迟。
Memory write CAS latency setting (CWL)	5.0, 6.0, 7.0, 8.0	cycles	设置从写命令到存储器的第一个预期数据的延迟，以时钟周期表示。
Memory partial array self refresh	Full array, Half array {BA[2:0]=000, 0 01, 010, 011}, Quarter array {BA[2:0]=000, 0 01}, Eighth array {BA[2:0]=000}, Three Quarters array {BA[2:0]=010, 0 11, 100, 101, 110 , 111}, Half array {BA[2:0]=100, 1 01, 110, 111}, Quarter array {BA[2:0]=110, 111}, Eighth array {BA[2:0]=111}	—	决定是否只自刷新某些数组，而不是全部数组。根据 DDR3 SDRAM 规范，如果使用此功能时输入 self refresh ，位于数组中超出指定地址范围的数据会丢失。DDR3 SDRAM Controller with ALTMEMPHY IP 不支持此选项，所以如果使用 Altera 控制器，需要设成 Full Array 。

表 9 - 18. DDR3 SDRAM 初始化选项 (3/3)

参数名	范围	单位	说明
Memory auto self refresh method	Manual SR reference (SRT) or ASR enable (Optional)	—	设置存储器件的自动自刷新方法。DDR3 SDRAM Controller with ALTMEMPHY IP 当前不支持用于扩展温度存储器自刷新的 ASR 选项。
Memory self refresh range	Normal or Extended	—	确定自刷新的温度范围。使用此选项的同时也需要使用可选的自动自刷新选项。Altera 控制器当前不支持扩展温度自刷新操作。

表 9 - 19. DDR3 SDRAM 时序参数设置 (1/2) (1)

参数名	范围	单位	说明
Time to hold memory reset before beginning calibration	0 - 1000000	μs	DDR3 SDRAM 器件初始化进程中，MRS 命令发出前，一个上电循环后保持复位的最短时间。
t_{INIT}	0.001 - 1000	μs	最短存储器初始化时间。复位后，在此期间控制器不发出任何命令到存储器。
t_{MRD}	2 - 39	ns	最短的加载模式寄存器命令周期。加载模式寄存器命令发出后，任何其它命令发出前控制器等待的时间。 在 DDR3 SDRAM 高性能控制器中并根据 Micron 器件数据表中的 t_{CK} 周期指定 t_{MRD} 以 ns 为单位。通过将数据表中指定的周期数乘以 t_{CK} 可以将 t_{MRD} 转换成 ns，其中 t_{CK} 是存储器操作频率，而不是存储器的 t_{CK} 。
t_{RAS}	8 - 200	ns	最短的 active 到 precharge 时间。active 命令发出后，precharge 命令发送到同一 bank 前控制器等待的时间。
t_{RCD}	4 - 65	ns	最短的 active 到 read-write 时间。active 命令发出后，在此期间内控制器不发出 read 或 write 命令到 bank。
t_{RP}	4 - 65	ns	最短的 precharge 命令时间。precharge 命令发出后，在此期间控制器不访问 bank。
t_{REFI}	1 - 65534	μs	refresh 命令之间的最大间隔时间。除非开启用户控制的刷新，否则控制器在此间隔执行常规刷新操作。
t_{RFC}	14 - 1651	ns	最短的 auto-refresh 命令持续时间。auto-refresh 命令发出后，控制器在运行前等待的时间。
t_{WR}	4 - 65	ns	最短写恢复时间。发出 precharge 命令前，写传输结束后控制器等待的时间。
t_{WTR}	1 - 6	t_{CK}	最短的 write-to-read 命令延时。在发出下一个 read 命令到同一 bank 中之前，write 命令结束后控制器等待的时间。该时序参数以时钟周期为单位，值取到下一个整数。
t_{AC}	0 - 750	ps	DQ 输出存取时间。
t_{DQSCK}	50 - 750	ps	从 CK/CK# 信号的 DQS 输出存取时间。
t_{DQSQ}	50 - 500	ps	DQS 到 DQ 的最大 skew；DQS 到最后 DQ 有效，每组，每次存取。
t_{DQSS}	0 - 0.3	t_{CK}	正 DQS 锁存边沿到关联时钟边沿。

表 9 - 19. DDR3 SDRAM 时序参数设置 (2/2) (1)

参数名	范围	单位	说明
t_{DH}	10 - 600	ps	DQ 和 DM 相对于 DQS 的输入保持时间，它有一个降额值，取决于 DQS 的斜率（对于 DDR 以及 DDR2 SDRAM 接口）和 DQS 是单端还是差分（对于 DDR2 SDRAM 接口。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{DS}	10 - 600	ps	DQ 和 DM 相对于 DQS 的输入建立时间，它有一个降额值，取决于 DQS 的斜率（对于 DDR 以及 DDR2 SDRAM 接口）和 DQS 是单端还是差分（对于 DDR2 SDRAM 接口。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{DSH}	0.1 - 0.5	t_{CK}	从 CK 的 DQS 下降沿保持时间。
t_{DSS}	0.1 - 0.5	t_{CK}	到 CK 的 DQS 下降沿建立时间。
t_{IH}	50 - 1000	ps	地址和控制输入保持时间，它具有一个降额值，取决于 CK 和 CK# 时钟的斜率和地址和命令信号。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{IS}	65 - 1000	ps	地址和控制输入建立时间，它具有一个降额值，取决于 CK 和 CK# 时钟的斜率和地址和命令信号。一定要使用正确数值，输入的值要关系到 $V_{REF}(dc)$ ，而不是 $V_{IH}(ac)$ min 或 $V_{IL}(ac)$ max。关于如何对规格降额的详细信息，请参考第 9 - 48 页 “降额存储器建立和保持时序”。
t_{QHS}	0 - 700	ps	最大数据保持 skew 因子。
t_{QH}	0.1 - 0.6	t_{CK}	DQ 输出保持时间。
t_{RRD}	2.06 - 64	ns	activate 到 activate 时间，每个器件，RAS 到 RAS 延迟时序参数。
t_{FAW}	7.69 - 256	ns	four-activate window 时间，每个器件。
t_{RTP}	2.06 - 64	ns	read 到 precharge 时间。


表 9 - 19 注释：

- (1) 关于参数范围的相关信息，请参考存储器数据表。一些参数以时钟周期 (t_{CK}) 为单位。如果 MegaWizard Plug-In Manager 要求以时间 (ps 或 ns) 为单位输入值，那么需要将此值与您接口的时钟周期相乘（不是存储器数据表中列出的最大时钟周期）。

降额存储器建立和保持时序

由于存储器数据表中的基本建立和保持时间规范假设的输入斜率对于 Altera 器件可能不准，因此需要在 **Preset Editor** 对话框中降额或更新以下存储器规范：

- t_{DS}
- t_{DH}
- t_{IH}
- t_{IS}

 对于 Arria II GX 和 Stratix IV 器件 (不包括 DDR SDRAM)，不需要使用 **Preset Editor** 进行降额。您只需要输入 V_{REF} 参数，在 **Board Settings** 标签中输入斜率信息时会自动完成降额。


降额值后，需要对该值进行标准化，因为 Altera 输入和输出时序规范与 V_{REF} 关联。然而，JEDEC 基本建立时间规格关联到 V_{IH}/V_{IL} AC 电平；JEDEC 基本保持时间规范关联到 V_{IH}/V_{IL} DC 电平。

当存储器建立和保持时间数被降额并标准化到 V_{REF} 时，需要在 **Preset Editor** 对话框中更新这些值以确保您的时序约束是正确的。

例 9 - 1. 降额 DDR2 SDRAM

例如，根据 JEDEC，400-MHz DDR2 SDRAM 有以下规格，假设 1V/ns DQ 斜率上升信号和 2V/ns 差分斜率：

- Base $t_{DS} = 50$
- Base $t_{DH} = 125$
- $V_{IH}(ac) = V_{REF} + 0.2 \text{ V}$
- $V_{IH}(dc) = V_{REF} + 0.125 \text{ V}$
- $V_{IL}(ac) = V_{REF} - 0.2 \text{ V}$
- $V_{IL}(dc) = V_{REF} - 0.125 \text{ V}$

 JEDEC 列出 t_{DS} 和 t_{DH} 规范的两组不同的基值和降额值，对具有 267 MHz 最大频率的 DDR2 SDRAM 器件使用单端还是差分 DQS 信号。此外，对于这些器件， $V_{IL}(ac)$ 和 $V_{IH}(ac)$ 值也可能不同。

一个上升沿的 V_{REF} 参考的建立和保持信号是：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 50 + 0 + 200 = 250 \text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 125 + 0 + 67.5 = 192.5 \text{ ps}$$

如果写数据的输出斜率不同于 1V/ns，那么您必须先降额 t_{DS} 和 t_{DH} 值，然后将这些 AC/DC 电平规范转换成 V_{REF} 规范。

对于 2V/ns DQ 斜率上升信号和 2V/ns DQS-DQSn 斜率：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 25 + 100 + 100 = 225 \text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 100 + 45 + 62.5 = 207.5 \text{ ps}$$

对于 0.5V/ns DQ 斜率上升信号和 1V/ns DQS-DQSn 斜率：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 25 + 0 + 400 = 425 \text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 100 - 65 + 250 = 285 \text{ ps}$$

对于 address/command 斜率降额可以采用一个类似的方法。对于 t_{IS}/t_{IH} ，降额公式中使用的斜率是 address/command 斜率；对于 t_{DS}/t_{DH} ，使用 DQ 斜率。

例 9-2. 降额 DDR3 SDRAM

例如，根据 JEDEC，533-MHz DDR3 SDRAM 具有下面规范，假设 1V/ns DQ 斜率上升信号和 2V/ns DQS-DQSn 斜率：

- Base t_{DS} = 25
- Base t_{DH} = 100
- $V_{IH}(ac) = V_{REF} + 0.175\text{ V}$
- $V_{IH}(dc) = V_{REF} + 0.100\text{ V}$
- $V_{IL}(ac) = V_{REF} - 0.175\text{ V}$
- $V_{IL}(dc) = V_{REF} - 0.100\text{ V}$

一个上升沿的 V_{REF} 参考的建立和保持信号是：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 25 + 0 + 175 = 200\text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 100 + 0 + 100 = 200\text{ ps}$$

如果写数据的输出斜率不同于 1V/ns，那么您必须先降额 t_{DS} 和 t_{DH} 值，然后将这些 AC/DC 电平规范转换成 V_{REF} 规范。

对于 2V/ns DQ 斜率上升信号和 2V/ns DQS-DQSn 斜率：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 25 + 88 + 87.5 = 200.5\text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 100 + 50 + 50 = 200\text{ ps}$$

对于 0.5V/ns DQ 斜率上升信号和 1V/ns DQS-DQSn 斜率：

$$t_{DS}(V_{REF}) = \text{Base } t_{DS} + \text{delta } t_{DS} + (V_{IH}(ac) - V_{REF})/\text{slew_rate} = 25 + 5 + 350 = 380\text{ ps}$$

$$t_{DH}(V_{REF}) = \text{Base } t_{DH} + \text{delta } t_{DH} + (V_{IH}(dc) - V_{REF})/\text{slew_rate} = 100 + 10 + 200 = 310\text{ ps}$$

PHY 设置

点击 **Next** 或者 **PHY Settings** 标签来设置表 9 - 20 中描述的选项。这些选项只适用于 Altera 目标器件。

表 9 - 20. ALTMEMPHY PHY 设置 (1/2)

参数名	适用的器件系列		说明
	DDR/DDR2 SDRAM	DDR3 SDRAM	
Use dedicated PLL outputs to drive memory clocks	HardCopy II and Stratix II (prototyping for HardCopy II)	不支持	<p>开启以使用专用 PLL 输出来生成外部存储器时钟，这是 HardCopy II ASIC 及其 Stratix II FPGA 原型所要求的。关闭后，DDIO 输出寄存器生成时钟输出。</p> <p>当您将 DDIO 输出寄存器用于存储器时钟时，存储器时钟以及 DQS 信号都会很好地对齐并很容易满足 t_{DQSS} 规范。然而，当专用时钟输出用于存储器时钟时，存储器时钟以及 DQS 信号不会正确地对齐，需要 PLL 的正相位偏移以对齐信号。</p>
Dedicated memory clock phase	HardCopy II and Stratix II (prototyping for HardCopy II)	不支持	<p>当使用专用 PLL 输出驱动存储器时钟时，对齐 CK/CK# 信号与 DQS/DQS# 信号所需要的相移。</p>
Use differential DQS	Arria II GX, Stratix III, and Stratix IV	不支持	<p>使能此功能以实现更高的信号完整性。建议用于 333 MHz 或更高频率的操作。仅用于 DDR2 SDRAM 的选项，因为 DDR SDRAM 不支持差分 DQSS。</p>
Enable external access to reconfigure PLL prior to calibration	HardCopy II, Stratix II, Stratix III, and Stratix IV (prototyping for HardCopy II)	HardCopy II	<p>当对 HardCopy II、Stratix II、Stratix III 和 Stratix IV 器件使能此选项时，ALTPLL_RECONFIG megafunction 的输入达到顶层以用于调试。</p> <p>此选项使您能够在校准前重配置 PLL，在开始校准读侧上的重同步时钟前调整存储器时钟 (mem_clk_2x) 的相位。读侧上的重同步时钟校准取决于写侧上的存储器时钟的相位。</p>
Instantiate DLL externally	All supported device families, except for Cyclone® III devices	所有支持的器件系列	<p>如果想应用非标准的相移到 DQS 捕捉时钟，那么对 Stratix III、Stratix IV、HardCopy III 或 HardCopy IV 器件使用此选项。ALTMEMPHY DLL offsetting I/O 然后能够连接到外部 DLL 和 Offset Control Block。</p> <p>由于 Cyclone III 器件没有 DLL，因此不支持该功能。</p>
Enable dynamic parallel on-chip termination	Stratix III and Stratix IV	不支持	<p>此选项提供 I/O 阻抗匹配和匹配性能。勾选此选项后，ALTMEMPHY megafunction 在读期间使能并行匹配，在写期间使能串行匹配。仅适用于 DDR 和 DDR2 SDRAM 接口，其中 DQ 和 DQS 是双向的。使用动态匹配要求使用 OCT 校准模块，这样可能会根据您的 R_{UP}/R_{DN} 管脚位置来限制 R_{UP}/R_{DN} 管脚布局。</p> <p>尽管 DDR SDRAM 不支持 ODT，但在 Altera FPGA 中仍然支持动态 OCT。</p> <p>关于详细信息，请参考 <i>Stratix III Device Handbook</i> 卷 1 的 <i>External Memory Interfaces in Stratix III Devices</i> 章节，或者 <i>Stratix IV Device Handbook</i> 卷 1 的 <i>External Memory Interfaces in Stratix IV Devices</i> 章节。</p>

表 9 - 20. ALTMEMPHY PHY 设置 (2/2)

参数名	适用的器件系列		说明
	DDR/DDR2 SDRAM	DDR3 SDRAM	
Clock phase	Arria II GX, Arria GX, Cyclone III, HardCopy II, Stratix II, and Stratix II GX	Arria II GX	调整地址和命令相位能够改善存储器上的地址和命令建立和保持裕量，从而对传播延迟（根据不同的负载而不同）进行补偿。您可以根据 phy_clk 和 write_clk 时钟的上升沿和下降沿选择 0°、90°、180° 或者 270°。在 Stratix IV 和 Stratix III 器件中，时钟相位被设置成 dedicated 。
Dedicated clock phase	Stratix III and Stratix IV	不支持	将专用 PLL 输出用于地址和命令时，您可以选择任何合法的 PLL 相移来改善地址和命令信号的建立和保持。此值的设置范围在 180° 到 359° 之间，默认值为 240°。然而，对于半速率设计而言，通常 PHY 时序要求一个大于 240° 的值，对于全速率设计，要求一个大于 270° 的值。
Board skew	All supported device families except Arria II GX and Stratix IV devices	不支持	对于 FPGA 到存储器（分布式存储器或 DIMM）的整个接口，任意两个存储器接口之间的最大偏斜。此参数包括所有类型的信号（数据、选通、时钟、地址和命令信号）。您需要输入最坏情况偏斜，无论是在 DQS/DQ 组中、跨越所有组还是地址和命令和时钟信号。此参数生成 .sdc 时序约束。
Autocalibration simulation options	所有支持的器件系列		可选择 Full Calibration （仿真时间较长）， Quick Calibration 或 Skip Calibration 。 关于详细信息，请参考 <i>Simulating Memory IP</i> 章节中的“Simulation Options”部分。

电路板设置

点击 **Next** 或者 **Board Settings** 标签来设置表 9 - 21 中描述的选项。通过设置电路板设置参数可以在时序分析中模拟板级效应。如果选择 Arria II GX 或者 Stratix IV 器件用于您的接口，那么这些选项是可用的，否则是禁用的。对于使用 DDR SDRAM 的所有器件，这些选项也是禁用的。

表 9 - 21. ALTMEMPHY 电路板设置 (1/2)

参数名	单位	说明
Number of slots/discrete devices	—	设置 single-rank 或者 multi-rank 配置。
CK/CK# slew rate (differential)	V/ns	设置 CK 和 CK# 信号的差分斜率。
Addr/command slew rate	V/ns	设置地址和命令信号的斜率。
DQ/DQS# slew rate (differential)	V/ns	设置 DQ 和 DQS# 信号的差分斜率。
DQ slew rate	V/ns	设置 DQ 信号的斜率。
Addr/command eye reduction (setup)	ns	设置由于地址和命令信号的 ISI 导致的 setup 一侧上的眼图缩减。
Addr/command eye reduction (hold)	ns	设置由于地址和命令信号的 ISI 导致的 hold 一侧上的眼图缩减。
DQ eye reduction	ns	设置由于 DQ 信号的 ISI 导致的 setup 一侧上的眼图总缩减。
Delta DQS arrival time	ns	设置由于 ISI 导致的到达时间范围变化的增加。
Max skew between DIMMs/devices	ns	设置 rank 之间 DQ 信号上的最大偏斜或传播延时，特别是不同 slot 的 DIMM。 对于 DIMM 以及器件，此值影响 multi-rank 配置的 DDR2 接口的 Resynchronization 裕量。

表 9 - 21. ALTMEMPHY 电路板设置 (2/2)

参数名	单位	说明
Max skew within DQS group	ns	设置 DQS 组中 DQ 管脚之间的最大偏斜。此值影响所有配置 (single- 或 multi-rank, DIMM 或器件) 的 DDR2 接口的 Read Capture 和 Write 裕量。
Max skew between DQS groups	ns	设置不同 DQS 组中 DQS 信号之间的最大偏斜。此值影响 single- 或 multi-rank 配置 DDR2 接口的 Resynchronization 裕量。
Addr/command to CK skew	ns	设置 CK 信号与地址和命令信号之间的偏斜或传播延时。正值代表长于 CK 信号的地址和命令信号。负值代表短于 CK 信号的地址和命令信号。Quartus II 使用此偏斜来优化 address/command 信号的延迟以具有正确的建立和保持裕量用于 DDR2 接口。

控制器设置



 本章节对 11.0 版本引入的 High Performance Controller II (HPC II) 高级特性的参数作了介绍。在更早版本中创建的设计如果在 11.0 版本中重新生成，那么将不会继承这些新的高级特性；要了解关于不具有 11.0 版本高级特性的 HPC II 参数的详细信息，请参考 *External Memory Interface Handbook for Quartus II version 10.1*，可从 Altera 网站上的 [Literature: External Memory Interfaces](#) 页面上获得。

表 9 - 22 列出了 Controller Settings 标签中提供的选项。

表 9 - 22. Controller Settings (1/2)

参数	说明
Controller architecture	指定控制器体系结构。
Enable self-refresh controls	开启以使控制器控制什么时候将外部存储器置于 self-refresh 模式，请参考 <i>External Memory Interface Handbook</i> 的 Functional Description—HPC II Controller 章节中的 “User-Controlled Self-Refresh” 部分。
Enable power down controls	开启以使控制器控制什么时候将外部存储器置于 power-down 模式。
Enable auto power down	在控制器中观测到指定数量的空闲控制器时钟周期后，控制器自动将外部存储器置于 power-down 模式。您可以在 Auto Power Down Cycles 域指定空闲周期数，在此空闲周期后，控制器使存储器掉电，请参考 <i>External Memory Interface Handbook</i> 的 Functional Description—HPC II Controller 章节中的 “Automatic Power-Down with Programmable Time-Out” 部分。
Auto power down cycles	确定控制器将外部存储器置于 power-down 模式前所需的空闲控制器时钟周期数。该值的合法范围从 1 到 65,535。 如果该值设为 0 时钟周期，那么 auto power-down 模式被禁用。
Enable user auto-refresh controls	开启使控制器发出一个单一刷新。
Enable auto-precharge control	开启以使能控制器顶层的 auto-precharge 控制。请求读或写突发的同时置位 auto-precharge 控制信号使您能够指定控制器在读或写突发的结束是否应该关闭 (auto-precharge) 当前打开页。
Enable reordering	开启以使控制器命令和数据重排序，从而达到最高效率。
Starvation limit for each command	指定执行一个等待命令前能够执行的命令数量。该值的合法范围从 1 到 63。

表 9 - 22. Controller Settings (2/2)

参数	说明
Local-to-memory address mapping	<p>使您能够控制 Avalon 接口上 address bit 之间的映射，和存储器件上 chip, row, bank 和 column bit 之间的映射。</p> <p>如果您的应用发出大于存储器列大小的突发，那么选择 Chip-Row-Bank-Column 选项。此选项使控制器能够在突发达到列的末端时使用其 look-ahead bank management 特性来隐藏改变当前打开页所产生的影响。</p> <p>另一方面，如果您的应用包括几个主端口 (master)，每个主端口使用存储器的不同区域，那么选择 Chip-Bank-Row-Column 选项。此选项使您能够使用最高地址位来分配存储器中的物理 bank 以达到每一个主端口。物理 bank 的分配避免了不同的主端口访问可能导致低效的同一个 bank，因为控制器然后必须打开并关闭同一 bank 中的行。</p>
Command queue look-ahead depth	指定 command queue look-ahead depth 值来控制 look-ahead bank management 逻辑检验的读或写请求数。
Local maximum burst count	指定突发数来配置控制器从端口接受的最大 Avalon 突发数。
Reduce controller latency by	指定减少控制器延迟的控制器时钟周期。默认值为 0，但可选择 1 以时序收敛为代价来提高您设计的延迟性能。
Enable configuration and status register interface	开启以使能存储控制器的运行时配置和状态检索。使能此选项添加一个额外的 Avalon-MM 从端口到存储控制器顶层，存储控制器顶层支持运行时重配置和状态检索，以实现存储器时序参数，存储器地址大小和模式寄存器设置，以及控制器功能。如果 Error Detection and Correction Logic 选项被使能，那么同一从端口也使您能够控制和检索该逻辑的状态。关于详细信息，请参考 <i>External Memory Interface Handbook</i> 的 <i>Functional Description—HPC II Controller</i> 章节中的“Configuration and Status Register (CSR) Interface”部分。
Enable error detection and correction logic	开启以使能纠错编码 (ECC) 功能，用于单比特纠错和双比特纠错。
Enable auto error correction	开启此选项后，当 ECC 逻辑检测到单比特错误时，控制器能够执行自动纠错。或者，关闭此选项，在特定的时间进行纠错，以实现更高的系统效率。
Multiple controller clock sharing	<p>此选项仅在 SOPC Builder Flow 中可用。开启此选项使一个控制器能够使用系统（具有兼容 PLL）中另一个控制器的 Avalon 时钟。此选项 SOPC Builder 系统能够包括两个或更多个同步于您的主逻辑的存储控制器。</p> <p> 此选项不适用于 Cyclone III 或 Cyclone IV 器件系列。</p>
Local interface protocol	<p>指定用户逻辑与存储控制器之间的本地接口。Avalon-MM 接口使您能够轻松连接到其它 Avalon-MM 外设。</p> <p>HPC II 体系结构仅支持 Avalon-MM 接口。</p>

参数化基于 UniPHY IP 的存储控制器

本章节介绍具有 UniPHY IP 的 LPDDR2, DDR2, DDR3 SDRAM, QDR II, QDR II+ SRAM, RLDram II 和 RLDram 3 存储控制器的设置参数。

您可以在 UniPHY 参数编辑器中的 **Parameter Settings** 页面中参数化以下设置：

- PHY Settings (PHY 设置)
- Memory Parameters (存储器参数)
- Memory Timing (存储器时序)
- Board Settings (电路板设置)

- Controller Settings(控制器设置)
- Diagnostics(诊断)

如果您尝试创建不支持的配置，那么 MegaWizard Plug-In Manager 底部的文本框会显示关于存储器接口，警告和错误信息。窗口中显示的全部错误被纠正之前，**Finish** 按钮是禁用的。

下面章节详细介绍了 **Parameter Settings** 页面上的各个标签。

PHY 设置

表 9 - 23 列出了 PHY 参数。

表 9 - 23. 时钟参数 (1/3)

参数	说明
General Settings	
Speed Grade	指定 FPGA 目标器件的速度等级，该速度等级影响生成的时序约束和时序报告。
Generate PHY only	开启此选项以生成一个没有存储控制器的 UniPHY core。开启此选项时，AFI 接口被导出，您便能够轻松连接到您自己的存储控制器。 不适用于 RLDRAM 3 UniPHY，因为没有用于 RLDRAM 3 UniPHY 的控制器支持。
Clocks	
Memory clock frequency	驱动存储器件的时钟频率，精确到小数点后 4 位。 要获得您目标存储器配置的最高支持频率，请参考 Altera 主页上的 External Memory Interface Spec Estimator 页面。
Achieved memory clock frequency	PLL 生成的实际频率来驱动外部存储器接口（存储器时钟）。
PLL reference clock frequency	驱动 PLL 的输入时钟频率，精确到小数点后 4 位。
Rate on Avalon-MM interface	Avalon-MM 接口上数据总线的位宽。 Full 产生一个 2× 存储器数据位宽。 Half 产生一个 4× 存储器数据位宽。 Quarter 产生一个 8× 存储器数据位宽。将 Quarter 用于 533 MHz 及更高的存储器频率。 要对其它存储器选择 Avalon-MM 接口速率，请参考 Altera 网站的 External Memory Interface Spec Estimator 页面中您目标器件的本地接口时钟速率。
Achieved local clock frequency	PLL 生成的实际频率来驱动存储控制器的本地接口（AFI 时钟）。
Enable AFI half rate clock	导出 afi_half_rate 时钟，此时钟运行一半 AFI 时钟速率到顶层。
Advanced PHY Settings	
Advanced clock phase control	使能对时钟相位的访问。默认值应该满足大多数 DIMM 和电路板布局的需要，如果需要可以修改默认值以对更大的地址和命令对时钟偏斜进行补偿。 此选项仅用于 DDR, DDR2 和 DDR3 SDRAM。
Additional address and command clock phase	使您能够增加或减少地址和命令时钟上相移的数量。基相移中心对齐存储器上的地址和命令时钟，这在任何情况下都不是最优的设置。增加或减少相移的数量能改善时序。默认值是 0 度。 在 DDR, DDR2, DDR3 SDRAM 和 LPDDR2 SDRAM 中，此值可设在 -360 度和 360 度之间。在 QDRII/II+ SRAM 和 RLDRAM II 中，可用设置是 -45, -22.5, 22.5 和 45。 要实现最佳设置，需要根据地址和命令时序分析结果对该值进行调整。

表 9 - 23. 时钟参数 (2/3)

参数	说明
Additional phase for core-to-periphery transfer	使您能够相移 core-to-periphery 传输的锁存时钟。通过延迟锁存时钟，正相移值分别对内核中寄存器之间的传输和外设中半速率 DDIO_OUT 模块之间的传输改善建立时序。根据内核时序分析来调整此设置。 默认值是 0 度。此值可设在 -179 度和 179 度之间。
Additional CK/CK# phase	使您能够增加或减少 CK/CK# 时钟上相移的数量。基相移中心对齐存储器上的地址和命令时钟，这在任何情况下都不是最优的设置。增加或减少相移的数量能改善时序。默认值是 0 度。在 CK/CK# 上增加或减少相移也影响读、写和整平传输，但在地址和命令时钟上不会影响。 要实现最佳设置，需要根据地址和命令时序分析结果对该值进行调整。要确保时钟相移调整后满足读、写和写整平时序。当调整 Additional address and command clock phase 后出现内核时序失败，需要调整此值。 默认值是 0 度。此值可设在 -360 度和 360 度之间。 此选项仅适用于 LPDDR2, DDR, DDR2 和 DDR3 SDRAM。
Supply voltage	电源电压和子系列类型存储器。 此选项仅适用于 DDR3 SDRAM。
I/O standard	I/O 标准电压。根据您设计的存储器标准来设置 I/O 标准。
PLL sharing mode	选择 No sharing 时，参数编辑器例化 PLL 模块，而不导出 PLL 信号。选择 Master 时，参数编辑器例化 PLL 模块并导出 PLL 信号。选择 Slave 时，参数编辑器显示一个 PLL 接口，您必须连接一个外部 PLL 主端口以驱动 PLL 从接口信号。 若不共享 PLL，则选择 No sharing 。否则选择 Master 或 Slave 。 关于资源共享的详细信息，请参考 <i>External Memory Interface Handbook</i> 的 Functional Description—UniPHY 章节中的 “The DLL and PLL Sharing Interface” 部分。
Number of PLL sharing interfaces	此选项用于指定需要创建的 PLL 共享接口数，方便 Qsys 流程中多个一对一连接的创建。在 Megawizard 中，您可以选择一个共享接口，手动将主端口连接到所有从端口。 当 DLL sharing mode 设置成 Master 时，使能此选项。
DLL sharing mode	当选择 No sharing 时，参数编辑器例化 DLL 模块，而不导出 DLL 信号。选择 Master 时，参数编辑器例化 DLL 模块并导出 DLL 信号。选择 Slave 时，参数编辑器显示一个 DLL 接口，您必须连接一个外部 DLL 主端口以驱动 DLL 从接口信号。 若不共享 DLL，则选择 No sharing 。否则选择 Master 或 Slave 。 关于资源共享的详细信息，请参考 <i>External Memory Interface Handbook</i> 的 Functional Description—UniPHY 章节中的 “The DLL and PLL Sharing Interface” 部分。
Number of DLL sharing interfaces	此选项用于指定需要创建的 PLL 共享接口数，方便 Qsys 流程中多个一对一连接的创建。在 Megawizard 中，您可以选择一个共享接口，手动将主端口连接到所有从端口。 当 PLL sharing mode 设置成 Master 时，使能此选项。
OCT sharing mode	当选择 No sharing 时，参数编辑器例化 OCT 模块，而不导出 OCT 信号。选择 Master 时，参数编辑器例化 OCT 模块并导出 OCT 信号。选择 Slave 时，参数编辑器显示一个 OCT 接口，您必须连接一个外部 OCT 控制模块以驱动 OCT 从接口信号。 若不共享 OCT，则选择 No sharing 。否则选择 Master 或 Slave 。 关于资源共享的详细信息，请参考 <i>External Memory Interface Handbook</i> 的 Functional Description—UniPHY 章节中的 “The OCT Sharing Interface” 部分。
Number of OCT sharing interfaces	此选项用于指定需要创建的 OCT 共享接口数，方便 Qsys 流程中多个一对一连接的创建。在 Megawizard 中，您可以选择一个共享接口，手动将主端口连接到所有从端口。 当 DLL sharing mode 设置成 Master 时，使能此选项。

表 9 - 23. 时钟参数 (3/3)

参数	说明
HardCopy compatibility	对生成的 IP 内核使能所有需要的 HardCopy 兼容性选项。对于某些参数化，在写数据通路添加一个流水线阶段，以帮助采用 HardCopy 器件的设计实现更大挑战性的时序收敛；流水线阶段不影响总体读写延迟。 如果您将设计移植到 HardCopy 器件，则需要启用此选项。关于详细信息，请参考 <i>HardCopy Design Migration Guidelines</i> 章节。
Reconfigurable PLL location	当您将存储器接口中使用的 PLL 设置成运行时可重配置，你必须指定 PLL 的位置。此约束生成一个只能布局在指定侧上的 PLL。 开启 HardCopy compatibility 来使能此选项。在 HardCopy 设计中，你必须根据接口位置指定 PLL 位置。
Sequencer optimization	选择 Performance 以使能基于 Nios II 的 sequencer，或者选择 Area 以使能基于 RTL 的 sequencer。 Altera 建议对高于 400 MHz 的存储器时钟频率使能基于 Nios II 的 sequencer，如果想减少资源使用，则使能基于 RTL 的 sequencer。 此选项仅适用于 QDRII 和 QDR II+ SRAM，和 RLDRAM II。

存储器参数

使用此标签来应用存储器厂商的数据表中的存储器参数。

LPDDR2, DDR2 和 DDR3 SDRAM

表 9 - 24 列出了 LPDDR2, DDR2 和 DDR3 SDRAM 的存储器参数。

表 9 - 24. 存储器参数 (1/3)

参数	说明
Memory vendor	存储器的供应商。根据使用选择存储器供应商。对于未列出的存储器供应商，选择具有最近存储器参数的 JEDEC，并根据所使用的存储器供应商的值来编辑参数值。然而，如果从存储器预置列表选择一个配置，那么该预置设置的默认存储器供应商将被自动选择。
Memory format	存储器件的格式。 如果只使用存储器件，则选择 Discrete 。选择 Unbuffered 或 Registered 用于 DIMM 格式。使用 DIMM 格式以启用整平电路，仅用于 LPDDR2 支持器件。DDR2 也支持 DIMM。
Memory device speed grade	存储器件能够运行的最大频率。
Total interface width	存储器件的 DQ 管脚总数。对于 DDR2 和 DDR3 SDRAM (带或不带整平)，限制成 144 bit。 由于最大的 Avalon 数据位宽是 1024，因此总接口位宽取决于 Avalon-MM 接口上的速率。如果您选择 144 bit 作为 Quarter-rate 的总接口位宽，那么 avalon 数据位宽为 1152，超过了最大的 avalon 数据位宽。
DQ/DQS group size	每个 DQS 组的 DQ bit 数量。
Number of DQS groups	根据总接口位宽和 DQ/DQS 组的容量参数自动计算出的 DQS 组数。
Number of chip selects (DDR2 and DDR3 SDRAM device only)	chip-selects 的数量，IP core 将此数量的 chip-selects 用于当前器件配置。 根据存储器件的数量指定 chip-selects 的总数。
Depth expansion (LPDDR2 SDRAM)	指定以深度扩展的器件数。仅支持单一 chip select。
Number of clocks	存储器接口上时钟总线的位宽。

表 9 - 24. 存储器参数 (2/3)

参数		说明
Row address width		存储器接口上行地址的位宽。
Column address width		存储器接口上列地址的位宽。
Bank-address width		存储器接口上 bank 地址总线的位宽。
Enable DM pins		指定存储器件的 DM 管脚是否由 FPGA 驱动。关闭此选项可以避免在使用 x4 模式存储器件时过度使用 FPGA 器件管脚。 使用 x4 模式存储器件时，对 DDR3 SDRAM 关闭此选项。 若使用 Avalon 字节使能，则必须开启此选项。
DQS# Enable (DDR2)		开启差分 DQS 信号以改善信号完整性和提高系统性能。 此选项仅适用于 DDR2 SDRAM。
Memory Initialization Options—DDR2		
Address and command parity		使能 address/command 奇偶校验检测。这是 Registered DIMM 所要求的。
Mode Register 0	Burst length	指定突发长度。
	Read burst type	指定给定突发访问顺序是顺序存取还是交叉存取。 对 Altera 存储控制器使用顺序访问。当在 PHY Settings 标签上使能 Generate PHY only 参数并且是用具有交叉访问的用户控制器时，可以使用交叉访问。
	DLL precharge power down	确定 precharge 掉电期间存储器中 DLL 是处于 slow exit 模式还是 fast exit 模式。关于详细信息，请参考存储器供应商数据表。
	Memory CAS latency setting	确定 READ 命令与存储器件上输出数据的第一个可用 bit 之间的时钟周期数。关于详细信息，请参考存储器供应商数据表中的 speed bin 表。 根据目标器件速度等级和存储器时钟频率设置该参数。
Mode Register 1	Output drive strength setting	确定存储器件上输出驱动器阻抗设置。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。
	Memory additive CAS latency setting	确定存储器件的前置 CAS 附加延迟。 使能此功能以提高命令和总线效率，并提升系统带宽。关于详细信息，请参考 <i>Optimizing the Controller</i> 章节。
	Memory on-die termination (ODT) setting	确定存储器件上的片上匹配阻值。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。
Mode Register 2	SRT Enable	确定自刷新温度 (SRT)。对于常温 (0-85C) 选择 1x refresh rate ，对于高温 (>85C) 选择 2x refresh rate 。
Memory Initialization Options—DDR3		
Mirror Addressing: 1 per chip select		指定 multiple rank DIMM 的镜像寻址。请参考存储器供应商数据表来获得详细信息。在此区域输入镜像地址的 rank。例如，对于四个 chip select，输入 1101 来镜像 chip select #3, #2 和 #0 上的地址。
Address and command parity		使能 address/command 奇偶校验检查以检测数据传输中的错误。这是 Registered DIMM (RDIMM) 所要求的。

表 9 - 24. 存储器参数 (3/3)

参数		说明
Mode Register 0	Read burst type	指定给定突发访问顺序是顺序存取还是交叉存取。 对 Altera 存储控制器使用顺序访问。当在 PHY Settings 标签上使能 Generate PHY only 参数并且是用具有交叉访问的用户控制器时，可以使用交叉访问。
	DLL precharge power down	指定 precharge 掉电期间存储器件中的 DLL 是开启还是关闭。
	Memory CAS latency setting	确定 READ 命令与存储器件上输出数据的第一个可用 bit 之间的时钟周期数以及接口频率。关于详细信息，请参考存储器供应商数据表中的 speed bin 表。 根据目标器件速度等级和存储器时钟频率设置该参数。
Mode Register 1	Output drive strength setting	存储器件上输出驱动器阻抗设置。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。
	Memory additive CAS latency setting	确定存储器件的前置 CAS 附加延迟。 使能此功能以提高命令和总线效率，并提升系统带宽。关于详细信息，请参考 <i>Optimizing the Controller</i> 章节。
	ODT Rtt nominal value	确定存储器件上的片上匹配阻值。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。
Mode Register 2	Auto selfrefresh method	禁止或使能 auto selfrefresh。
	Selfrefresh temperature	指定自刷新温度为 Normal 或 Extended。
	Memory write CAS latency setting	从内部写命令的发出到第一个输入数据的锁存的时钟周期数。关于详细信息，请参考存储器供应商数据表中的 speed bin 表。根据目标器件速度等级和存储器时钟频率设置该参数。
	Dynamic ODT (Rtt_WR) value	存储器件的动态 ODT 功能的模式，用于 multi-rank 配置。请参考 <i>DDR2 and DDR3 SDRAM Board Layout Guidelines</i> 。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。
Memory Initialization Options— LPDDR2		
Mode Register 1	Burst Length	指定突发长度。
	Read Burst Type	指定给定突发访问顺序是顺序存取还是交叉存取。 对 Altera 存储控制器使用顺序访问。当在 PHY Settings 标签上使能 Generate PHY only 参数并且是用具有交叉访问的用户控制器时，可以使用交叉访问。
Mode Register 2	Memory CAS latency setting	确定 READ 命令与存储器件上输出数据的第一个可用 bit 之间的时钟周期数。 根据目标存储器接口频率设置该参数。请参考存储器数据表和目标存储器速度等级。
Mode Register 3	Output drive strength settings	确定存储器件上输出驱动器阻抗设置。 要实现最佳的信号完整性，需要根据电路板仿真结果选择最佳设置。

QDR II 和 QDR II+ SRAM

表 9 - 25 介绍了 QDR II 和 QDR II+ SRAM 的存储器参数。

表 9 - 25. 存储器参数 (1/2)

参数	说明
Address width	存储器件上地址总线的位宽。
Data width	存储器件上数据总线的位宽。

表 9 - 25. 存储器参数 (2/2)

参数	说明
Data-mask width	存储器件上 data-mask 的位宽。
CQ width	存储器件上 CQ（读选通）总线的位宽。
K width	存储器件上 K（写选通）总线的位宽。
Burst length	存储器件所支持的突发长度。关于详细信息，请参考存储器供应商数据表。
Topology	
x36 emulated mode	在 FPGA 上使用较小的存储器宽接口仿真较大的存储器宽接口。 当 FPGA 不支持 x36 DQ/DQS group 时，开启此选项。此选项使两个 x18 DQ/DQS group 能够仿真一个 x36 read data group。
Emulated write groups	write group 的数量，用于在 FPGA 上形成 x36 存储器接口。选择 2 以使用 2 个 x18 DQ/DQS group 形成 x36 write data group。选择 4 以使用 4 个 x9 DQ/DQS group 形成 x36 write data group。
Device width	指定用于位宽扩展的存储器件的数量。

RLDRAM II

表 9 - 26 介绍了 RLDRAM II 的存储器参数。

表 9 - 26. 存储器参数

参数	说明
Address width	存储器件上地址总线的位宽。
Data width	存储器件上数据总线的位宽。
Bank-address width	存储器件上 bank-address 总线的位宽。
Data-mask width	存储器件上 data-mask 的位宽。
QK width	存储器件上 QK（读选通）总线的位宽。 数据位宽设为 9 时，选择 1。数据位宽设为 18 或 36 时，选择 2。
DK width	存储器件上 DK（写选通）总线的位宽。 数据位宽设为 9 或 18 时，选择 1。数据位宽设为 36 时，选择 2。
Burst length	存储器件所支持的突发长度。关于详细信息，请参考存储器供应商数据表。
Memory mode register configuration	用于设置存储器模式的配置位。根据接口频率选择此选项。
Device impedance	选择 External (ZQ) 来使用外部阻抗电阻 (RQ) 调整驱动器阻抗。输出阻抗范围是 25-60Ω。您必须在 ZQ 管脚与 GND 之间连接 RQ 电阻。RQ 值必须是 5 倍输出阻抗。例如，60Ω 输出阻抗需要 300Ω RQ。 管脚电路板仿真设置此值。
On-Die Termination	开启此选项以使能存储器中的 ODT，匹配 DQs 和 DM 管脚到 V _{tt} 。读操作期间动态关闭此选项，写操作期间动态开启此选项。关于详细信息，请参考存储器供应商数据表。
Topology	
Device width	指定用于位宽扩展的存储器件的数量。

存储器时序

使用此标签来应用存储器厂商的数据表中的存储器时序。表 9-27 列出了存储器时序参数。

表 9-27. 存储器时序参数 (1/3)

参数	说明
DDR2/DDR3 SDRAM	
tIS (base)	到 CK 时钟上升的地址和控制建立时间。根据存储器速度等级和参考存储器供应商数据表进行设置。
tIH (base)	CK 时钟上升后的地址和控制保持时间。根据存储器速度等级和参考存储器供应商数据表进行设置。
tDS (base)	到时钟 (DQS) 上升的数据建立时间。根据存储器速度等级和参考存储器供应商数据表进行设置。
tDH (base)	时钟 (DQS) 上升后的数据保持时间。根据存储器速度等级和参考存储器供应商数据表进行设置。
tDQSQ	DQS, DQS# 到 DQ skew, 每次存取。根据存储器速度等级和参考存储器供应商数据表进行设置。
tQHS (DDR2) tQH (DDR3)	来自 DQS, DQS# 的 DQ 输出保持时间 (绝对时间值) 来自 DQS, DQS# 的 DQ 输出保持时间 (tCK 的百分比)。根据存储器速度等级和参考存储器供应商数据表进行设置。
tDQSCK	来自 CK/CK# 的 DQS 输出访问时间。根据存储器速度等级和参考存储器供应商数据表进行设置。
tDQSS	DQS 的第一个锁存沿到相关时钟沿 (tCK 的百分比)。根据存储器速度等级并参考存储器供应商数据表进行设置。
tQSH (DDR3) tDQSH (DDR2)	DQS Differential High Pulse Width (tCK 的百分比)。指定存储器接收到的 DQS 信号的最短 high 时间。根据存储器速度等级并参考存储器供应商数据表进行设置。
tDSH	来自 CK 的 DQS 下降沿保持时间 (tCK 的百分比)。根据存储器速度等级并参考存储器供应商数据表进行设置。
tDSS	DQS 下降沿到 CK 建立时间 (tCK 的百分比)。根据存储器速度等级并参考存储器供应商数据表进行设置。
tINIT	上电时存储器初始化时间。根据存储器速度等级并参考存储器供应商数据表进行设置。
tMRD	加载模式寄存器命令周期。根据存储器速度等级并参考存储器供应商数据表进行设置。
tRAS	active 到 precharge 时间。根据存储器速度等级并参考存储器供应商数据表进行设置。
tRCD	active 到 read 或 write 时间。根据存储器速度等级并参考存储器供应商数据表进行设置。
tRP	precharge 命令周期。根据存储器速度等级并参考存储器供应商数据表进行设置。
tREFI	refresh 命令间隔。根据存储器速度等级并参考存储器供应商数据表进行设置。
tRFC	auto-refresh 命令间隔。根据存储器速度等级并参考存储器供应商数据表进行设置。

表 9 - 27. 存储器时序参数 (2/3)

参数	说明
tWR	写恢复时间。根据存储器速度等级并参考存储器供应商数据表进行设置。
tWTR	Write 到 read 周期。根据存储器速度等级和存储器时钟频率进行设置。请参考存储器供应商数据表。
tFAW	四活动窗口时间。根据存储器速度等级和页面大小进行设置。请参考存储器供应商数据表。
tRRD	RAS 到 RAS 延迟时间。根据存储器速度等级，页面大小和存储器时钟速率进行设置。请参考存储器供应商数据表。
tRTP	read 到 precharge 时间。根据存储器速度等级进行设置。请参考存储器供应商数据表。 根据存储器接口频率和存储器时钟频率计算该值。
QDR II and QDR II+ SRAM	
tWL (cycles)	写延迟。对于为 2 的突发长度，设置写延迟为 0，对于为 4 的突发长度设置写延迟为 1。
tRL (cycles)	读延迟。根据存储器协议进行设置。请参考存储器数据表。
tSA	到 K 时钟上升的地址和控制建立时间。根据存储器协议进行设置。请参考存储器数据表。
tHA	K 时钟上升后的地址和控制保持时间。根据存储器协议进行设置。请参考存储器数据表。
tSD	到时钟 (K/K#) 上升的数据建立时间。根据存储器协议进行设置。请参考存储器数据表。
tHD	时钟 (K/K#) 上升后的数据保持时间。根据存储器协议进行设置。请参考存储器数据表。
tCQD	回波时钟高电平到数据有效时间。根据存储器协议进行设置。请参考存储器数据表。
tCQDOH	回波时钟高电平到数据无效时间。根据存储器协议进行设置。请参考存储器数据表。
Internal jitter	QDR II/II+ 内部抖动。请参考存储器数据表。
TCQHCQnH	CQ 时钟上升到 CQn 时钟上升时间（上升沿到上升沿）。根据存储器速度等级进行设置。请参考存储器数据表。
TKHKnH	K 时钟上升到 Kn 时钟上升时间（上升沿到上升沿）。根据存储器速度等级进行设置。请参考存储器数据表。
RLDRAM II	
Maximum memory clock frequency	存储器件能够运行的最大频率。根据存储器速度等级进行设置。请参考存储器数据表。
Refresh interval	刷新闻隔。根据存储器速度等级进行设置。请参考存储器数据表。
tCKH (%)	输入时钟 (CK/CK#) 高电平，以整个时钟周的百分比表示。根据存储器速度等级进行设置。请参考存储器数据表。
tQKH (%)	读时钟 (QK/QK#) 高电平，以 tCKH 的百分比表示。根据存储器速度等级进行设置。请参考存储器数据表。
tAS	到 CK 时钟上升的地址和控制建立时间。根据存储器速度等级进行设置。请参考存储器数据表。

表 9 - 27. 存储器时序参数 (3/3)

参数	说明
tAH	CK 时钟上升后的地址和控制保持时间。根据存储器速度等级进行设置。请参考存储器数据表。
tDS	到时钟 (CK/CK#) 上升的数据建立时间。根据存储器速度等级进行设置。请参考存储器数据表。
tDH	时钟 (CK/CK#) 上升后的数据保持时间。根据存储器速度等级进行设置。请参考存储器数据表。
tQKQ_max	QK 时钟沿到 DQ 数据沿时间（在同一组中）。根据存储器速度等级进行设置。请参考存储器数据表。
tQKQ_min	QK 时钟沿到 DQ 数据沿时间（在同一组中）。根据存储器速度等级进行设置。请参考存储器数据表。
tCKDK_max	到输入数据时钟的时钟 (max)。根据存储器速度等级进行设置。请参考存储器数据表。
tCKDK_min	到输入数据时钟的时钟 (min)。根据存储器速度等级进行设置。请参考存储器数据表。

表 9 - 28. RLD RAM 3 存储器参数

参数	说明
Enable data-mask pins	指定存储器的 DM 管脚是否由 FPGA 驱动。
Data-mask width	存储器件上 data-mask 的位宽。
Data width	存储器件上数据总线的位宽。
QK width	存储器件上 QK（读选通）总线的位宽。 数据位宽设为 18 时，选择 1。数据位宽设为 36 时，选择 4。
DK width	存储器件上 DK（写选通）总线的位宽。对于 x36 器件，DQ[8:0] 和 DQ[26:18] 参照 DK0/DK0#，DQ[17:9] 和 DQ[35:27] 参照 DK1/DK1#。
Address width	存储器件上地址总线的位宽。
Bank-address width	存储器件上 bank-address 总线的位宽。
Burst length	存储器件所支持的突发长度。关于详细信息，请参考存储器供应商数据表。
tRC	设置 tRC 的模式寄存器位。根据存储器速度等级和数据延迟设置 tRC。请参考存储器供应商数据表中的 tRC 表。
Data Latency	设置延迟的模式寄存器位。根据接口频率和存储器速度等级设置延迟。请参考存储器数据表中的 speed bin 表。
Output Drive	设置输出驱动阻抗设置的模式寄存器位。根据电路板仿真设置此值。
ODT	设置 ODT 设置的模式寄存器位。根据电路板仿真设置此值。
AREF Protocol	模式寄存器设置用于刷新 bank 的存储数据。选择 Multibank 同时刷新 4 个 bank。选择 Bank Address Control 通过设置 bank 地址刷新特定 bank。
Write Protocol	模式寄存器设置用于写协议。当选择多个 bank (dual bank 或 quad bank) 时，相同的数据写入到多个 bank 中。
Topology	
Device width	指定用于位宽扩展的存储器件的数量。

表 9 - 29. RLDRAM 3 存储器时序

参数	说明
Memory Device Timing	
Maximum memory clock frequency	存储器件能够运行的最大频率。
tDS (base)	到 DK/DK# 的数据建立时间的基本规范。根据存储器速度等级进行设置。请参考存储器数据表。
tDH (base)	来自 DK/DK# 的数据保持时间的基本规范。根据存储器速度等级进行设置。请参考存储器数据表。
tQKQ_max	QK/QK# 时钟沿到 DQ 数据沿（在同一组中）。根据存储器速度等级进行设置。请参考存储器数据表。
tQH (% of CK)	来自 QK/QK# 的 DQ 输出保持时间。根据存储器速度等级进行设置。请参考存储器数据表。
tCKDK_max(% of CK)	到输入数据时钟的时钟 (max)。根据存储器速度等级进行设置。请参考存储器数据表。
tCKDK_min (% of CK)	到输入数据时钟的时钟 (min)。根据存储器速度等级进行设置。请参考存储器数据表。
tCKQK_max	QK 沿到时钟沿偏斜 (max)。根据存储器速度等级进行设置。请参考存储器数据表。
tIS (base)	到 CK 的地址和控制建立时间的基本规范。根据存储器速度等级进行设置。请参考存储器数据表。
tIH (base)	来自 CK 的地址和控制保持时间的基本规范。根据存储器速度等级进行设置。请参考存储器数据表。
Controller Timing	
Read-to-Write NOP commands (min)	在 read 命令之后和在 write 命令之前的最少 no operation 命令数。此值必须不小于 $((\text{Burst Length}/2) + \text{RL} - \text{WL} + 2)$ 。“Bus Turnaround” 时序分析使用此值以及其它 delay/skew 参数来确定是否有总线竞争问题。 根据控制器规格进行设置。
Write-to-Read NOP commands (min)	在 write 命令之后和在 read 命令之前的最少 no operation 命令数。此值必须不小于 $((\text{Burst Length}/2) + \text{WL} - \text{RL} + 1)$ 。“Bus Turnaround” 时序分析使用此值以及其它 delay/skew 参数来确定是否有总线竞争问题。 根据控制器规格进行设置。
RLDRAM 3 Board Derate	
CK/CK# slew rate (differential)	CK/CK# 斜率（差分）。
Address/Command slew rate	地址和命令斜率。
DK/DK# slew rate (Differential)	DK/DK# 斜率（差分）。
DQ slew rate	DQ 斜率。
tIS	到 CK 的 Address/command 建立时间。

电路板设置

使用 **Board Settings** 标签输入时序分析中的板级影响。**Board Settings** 标签用于以下设置：

- Setup and hold derating（建立和保持降额）（仅适用于 LPDDR2/DDR2/DDR3 SDRAM, RLDRAM 3 和 RLDRAM II）
- Intersymbol interference（码间干扰）
- Board skews（电路板偏斜）



要获得精确的时序结果，您必须对您的 PCB 输入正确的电路板设置参数。

IP core 支持一个以及多个 chip-select 配置。对于某些 Altera 电路板，Altera 已经确定了对这些配置的输出信号的效应值，并在 wizard 中存储了对输出斜率和码间干扰 (ISI) 的效应值。



这些存储值代表特定的 Altera 电路板。您必须修改这些值以满足您电路板的板级效应。您可以使用 HyperLynx 或类似的仿真工具来获得代表您电路板的值。



要了解关于如何在 Quartus II 软件中包含您的电路板仿真结果和如何使用 pin planner 分配管脚的详细信息，请参考 Altera Wiki 网站的 [List of designs using Altera External Memory IP](#) 页面的设计流程教程和设计实例。



要了解关于时序降额方法的详细信息，请参考 [Analyzing Timing of Memory IP](#) 章节的“Timing Deration Methodology for Multiple Chip Select DDR2 and DDR3 SDRAM Designs”部分。

建立和保持降额

输出信号的斜率影响存储器件的建立和保持时间。您可以指定输出信号的斜率来了解它们对地址和命令信号以及 DQ 信号的影响，或者直接指定建立和保持时间。



您应该输入预布局 (line) 和后布局 (board) 仿真的 PCB 开发过程中产生的信息。

表 9 - 30 列出了建立和保持降额参数。

表 9 - 30. 建立和保持降额参数 (1/2)

参数	说明
LPDDR2/DDR2/DDR3 SDRAM/RLDRAM 3	
Derating method	降额方法。默认设置基于 Altera 内部电路板仿真数据。要根据您的电路板状况获得精确的时序分析结果，Altera 建议执行电路板仿真并在 Quartus II 中输入斜率来自动计算降额的建立和保持时间，或者直接输入降额的建立和保持时间。 关于详细信息，请参考 <i>Analyzing Timing of Memory IP</i> 章节的 “Timing Deration Methodology for Multiple Chip Select DDR2 and DDR3 SDRAM Designs” 部分。
CK/CK# slew rate (differential)	CK/CK# 斜率（差分）。
Address/Command slew rate	地址和命令斜率。
DQS/DQS# slew rate (Differential)	DQS 和 DQS# 斜率（差分）。
DQ slew rate	DQ 斜率。
tIS	到 CK 的 Address/command 建立时间。
tIH	来自 CK 的 Address/command 保持时间。
tDS	到 DQS 的数据建立时间。
tDH	来自 DQS 的数据保持时间。
RLDRAM II	
tAS Vref to CK/CK# Crossing	对于给定的 address/command 和 CK/CK# 斜率，存储器件数据表提供相应的 “tAS Vref to CK/CK# Crossing” 值，该值用于确定降额的 address/command 建立时间。
tAS VIH MIN to CK/CK# Crossing	对于给定的 address/command 和 CK/CK# 斜率，存储器件数据表提供相应的 “tAS VIH MIN to CK/CK# Crossing” 值，该值用于确定降额的 address/command 建立时间。
tAH CK/CK# Crossing to Vref	对于给定的 address/command 和 CK/CK# 斜率，存储器件数据表提供相应的 “tAH CK/CK# Crossing to Vref” 值，该值用于确定降额的 address/command 保持时间。
tAH CK/CK# Crossing to VIH MIN	对于给定的 address/command 和 CK/CK# 斜率，存储器件数据表提供相应的 “tAH CK/CK# Crossing to VIH MIN” 值，该值用于确定降额的 address/command 保持时间。
tDS Vref to CK/CK# Crossing	对于给定的数据和 DK/DK# 斜率，存储器件数据表提供相应的 “tDS Vref to CK/CK# Crossing” 值，该值用于确定降额的数据建立时间。
tDS VIH MIN to CK/CK# Crossing	对于给定的数据和 DK/DK# 斜率，存储器件数据表提供相应的 “tDS VIH MIN to CK/CK# Crossing” 值，该值用于确定降额的数据建立时间。

表 9 - 30. 建立和保持降级参数 (2/2)

参数	说明
tDH CK/CK# Crossing to Vref	对于给定的数据和 DK/DK# 斜率，存储器件数据表提供相应的“tDH CK/CK# Crossing to Vref”值，该值用于确定降额的数据保持时间。
tDH CK/CK# Crossing to VIH MIN	对于给定的数据和 DK/DK# 斜率，存储器件数据表提供相应的“tDH CK/CK# Crossing to VIH MIN”值，该值用于确定降额的数据保持时间。
Derated tAS	降额的 address/command 建立时间从“tAS”，“tAS Vref to CK/CK# Crossing”和“tAS VIH MIN to CK/CK# Crossing”参数自动计算得出。
Derated tAH	降额的 address/command 保持时间从“tAH”，“tAH CK/CK# Crossing to Vref”和“tAH CK/CK# Crossing to VIH MIN”参数自动计算得出。
Derated tDS	降额的数据建立时间从“tDS”，“tDS Vref to CK/CK# Crossing”和“tDS VIH MIN to CK/CK# Crossing”参数自动计算得出。
Derated tDH	降额的数据保持时间从“tDH”，“tDH CK/CK# Crossing to Vref”和“tDH CK/CK# Crossing to VIH MIN”参数自动计算得出。

码间干扰

码间干扰是指信号失真，其中一个符号干扰随后的符号。通常情况下，当从一个单 chip-select 配置到一个多 chip-select 配置时，由于有多个短截线导致反射，因此码间干扰会随之增加。表 9 - 31 列出了码间干扰参数。

表 9 - 31. ISI 参数 (1/2)

参数	说明
Derating method	选择默认的 Altera 设置（特定的 Altera 电路板），或者手动输入来自板级仿真的参数。 仅在 LPDDR2/DDR2/DDR3 SDRAM 中支持此选项。
Address and command eye reduction (setup)	与没有 ISI 的情况相比，由于地址和命令信号上的 ISI 导致的 setup 侧上（或眼左侧）眼图的缩小。（在 single rank 设计中，ISI 能够为零；在 multirank 设计中，ISI 对于精确的时序分析是必要的。） 要了解关于如何测量地址和命令信号的 ISI 值的详细信息，请参考 <i>Analyzing Timing of Memory IP</i> 章节中的“Measuring Eye Reduction for Address/Command, DQ, and DQS Setup and Hold Time”部分。
Address and command eye reduction (hold)	与没有 ISI 的情况相比，由于地址和命令信号上的 ISI 导致的 hold 侧上（或眼右侧）眼图的缩小。（在 single rank 设计中，ISI 能够为零；在 multirank 设计中，ISI 对于精确的时序分析是必要的。） 要了解关于如何测量地址和命令信号的 ISI 值的详细信息，请参考 <i>Analyzing Timing of Memory IP</i> 章节中的“Measuring Eye Reduction for Address/Command, DQ, and DQS Setup and Hold Time”部分。

表 9 - 31. ISI 参数 (2/2)

参数	说明
DQ/ D eye reduction	与没有 ISI 的情况相比，由于 DQ 信号上的 ISI 导致的眼图缩小。Altera 假设 ISI 在眼左右两侧对称地降低眼宽。 要了解关于如何测量地址和命令信号的 ISI 值的详细信息，请参考 <i>Analyzing Timing of Memory IP</i> 章节中的 “Measuring Eye Reduction for Address/Command, DQ, and DQS Setup and Hold Time” 部分。
Delta DQS/Delta K/ Delta DK arrival time	与没有 ISI 的情况相比，DQS 到达时间范围变化的增加。Altera 假设 ISI 导致 DQS 左右两侧对称变化。 要了解关于如何测量地址和命令信号的 ISI 值的详细信息，请参考 <i>Analyzing Timing of Memory IP</i> 章节中的 “Measuring Eye Reduction for Address/Command, DQ, and DQS Setup and Hold Time” 部分。

电路板偏斜

PCB 走线之间的偏斜会降低时序裕量。而且，在多个 chip-select 拓扑结构中，不同 chip select 之间的偏斜能够进一步降低时序裕量。参数编辑器的 **Board Skews** 使您能够通过输入相关参数来对这些变化进行补偿。



当 PCB 的电路板偏斜参数是正确时，您必须确保 TimeQuest Report DDR 中报告的时序裕量是正值。

表 9 - 32 列出了电路板偏斜参数。对于包含延迟值的参数公式，应该按照以下方法测量延迟：

- Non-fly-by 拓扑结构（平衡时钟树）
 - 分立器件 - 所有的延迟 (CK, Addr/Cmd, DQ 和 DQS) 都是从 FPGA 到每个存储器件。
 - UDIMM - 对于 UDIMM 上的每个存储器件，所有延迟 (CK, Addr/Cmd, DQ 和 DQS) 都是从 FPGA 到 UDIMM 连接器。
 - RDIMM - Addr/Cmd 和 CK 延迟是从 FPGA 到 RDIMM 上的寄存器。对于 RDIMM 上的每一个存储器件，DQ 和 DQS 延迟是从 FPGA 到 RDIMM 连接器。
 - LRDIMM - 从 FPGA 到 LRDIMM 上寄存器的延迟。
- Fly-by 拓扑结构
 - 分立器件 - Addr/Cmd 和 CK 延迟是从 FPGA 到第一个存储器件。DQ 和 DQS 延迟是从 FPGA 到每个存储器件。
 - UDIMM - Addr/Cmd 和 CK 延迟是从 FPGA 到 UDIMM 上第一个存储器件的连接器。对于 UDIMM 上的每一个存储器件，DQ 和 DQS 延迟是从 FPGA 到 UDIMM 连接器。
 - RDIMM - Addr/Cmd 和 CK 延迟是从 FPGA 到 RDIMM 上的寄存器。对于 RDIMM 上的每一个存储器件，DQ 和 DQS 延迟是从 FPGA 到 RDIMM 连接器。
 - LRDIMM - 从 FPGA 到 LRDIMM 上寄存器的延迟。

这些公式可用于任何给定存储器件，但由电路板或组限定符 (*_b* 或 *_g*) 标识的公式除外，它们只用于特定器件或被迭代的组。


 使用下面的电路板偏斜参数来帮助您计算电路板偏斜，请参考 [Board skew parameter tool](#)。

表 9 - 32. 电路板偏斜参数 (1/7)

参数	说明
LPDDR2/DDR2/DDR3 SDRAM	
DQ/DQS Package Deskew	<p>若要使用 DQ 和 DQS 管脚上的电路板走线对 FPGA 封装进行偏斜去除，则要使能此参数。此选项增加读采样和写裕量。当存储器时钟频率大于 533 MHz 时使能此选项。使能此选项可以改善读采样和写时序裕量。您也可以利用时序报告中的读采样和写时序裕量来使能此选项。</p> <p>此选项使能时，Pin-Out File (.pin) 中的 DQ 和 DQS 管脚会显示封装偏斜，同时时序分析不再包含封装偏斜。与 DQ 或 DQS 相关的所有其它电路板延迟和偏斜参数必须将封装和电路板一起考虑。关于详细信息，请参考 DDR2 and DDR3 Board Layout Guidelines。</p>
Address/Command Package Deskew	<p>若要使用 DQ 和 DQS 管脚上的电路板走线对 FPGA 封装进行偏斜去除，则要使能此参数。Enable this parameter if you will deskew the FPGA package with your board traces on the address and command pins. 此选项增加地址和命令裕量。当存储器时钟频率大于 533 MHz 时使能此选项。使能此选项可以改善地址和命令时序裕量。您也可以利用时序报告中的地址和命令裕量来使能此选项。</p> <p>此选项使能时，封装偏斜被输出在 Pin-Out File (.pin) 中的地址和命令管脚上，封装偏斜不包括在时序分析中。与 Address 和 Command 相关的所有其它电路板延迟和偏斜参数必须将封装和电路板一起考虑。关于详细信息，请参考 DDR2 and DDR3 Board Layout Guidelines。</p>
Maximum CK delay to DIMM/device	<p>从 FPGA 到存储器件的最长 CK 走线的延迟，不论是在 DIMM 上还是在与 FPGA 相同的 PCB 上，都用以下公式计算：</p> $\max_r[\max_n(CK_{n,r}PathDelay)]$ <p>公式中，n 是存储器时钟数，r 是 DIMM/ 器件的 rank 数。例如，在 dual-rank DIMM 实现中，如果每个 rank DIMM 中有 2 对存储器时钟，那么最大 CK 延迟由下面公式表示：</p> $\max(CK_1PathDelay\ rank\ 1, CK_2Path\ Delay\ rank\ 1, CK_1Path\ Delay\ rank\ 2, CK_2Path\ Delay\ rank\ 2)$
Maximum DQS delay to DIMM/device	<p>从 FPGA 到存储器件的最长 DQS 走线的延迟，不论是在 DIMM 上还是在与 FPGA 相同的 PCB 上，都用以下公式计算：</p> $\max_r[\max_n(DQS_{n,r}Path\ Delay)]$ <p>公式中，n 是 DQS 的数量，r 是 DIMM/ 器件的 rank 数。例如，在 dual-rank DIMM 实现中，如果每个 rank DIMM 中有 2 个 DQS，那么最大 DQS 延迟由下面公式表示：</p> $\max(DQS_1PathDelay\ rank\ 1, DQS_2Path\ Delay\ rank\ 1, DQS_1Path\ Delay\ rank\ 2, DQS_2Path\ Delay\ rank\ 2)$

表 9 - 32. 电路板偏斜参数 (2/7)

参数	说明
Minimum delay difference between CK and DQS	<p>到达所有 DIMM/ 器件中的同一个 DIMM/ 器件时, CK 信号与任意 DQS 信号之间的最小偏斜或最小正偏斜 (或者最大负偏斜) 由下面公式表示:</p> $\min_r \left[\min_{n,m} \{ (CK_{n,r} \text{Delay} - DQS_{m,r} \text{Delay}) \} \right]$ <p>公式中, n 是存储器时钟数, m 是 DQS 的数量, r 是 DIMM/ 器件的 rank 数量。例如, 在 dual-rank DIMM 实现中, 如果每个 rank DIMM 中有 2 对存储器时钟和 4 个 DQS 信号 (每个时钟两个), 那么 CK 与 DQS 之间的最小延迟差由下面公式表示:</p> $\min \left\{ \begin{array}{l} (CK_{1_1} \text{Delay} - DQS_{1_1} \text{Delay}), (CK_{1_1} \text{Delay} - DQS_{2_1} \text{Delay}), (CK_{2_1} \text{Delay} - DQS_{3_1} \text{Delay}), (CK_{2_1} \text{Delay} - DQS_{4_1} \text{Delay}) \\ (CK_{1_2} \text{Delay} - DQS_{1_2} \text{Delay}), (CK_{1_2} \text{Delay} - DQS_{2_2} \text{Delay}), (CK_{2_2} \text{Delay} - DQS_{3_2} \text{Delay}), (CK_{2_2} \text{Delay} - DQS_{4_2} \text{Delay}) \end{array} \right\}$ <p>此参数值影响 multi-rank 中带整平的 DDR3 接口写整平裕量。此参数值也可应用于任意 rank 数的非整平配置, 要求是 DQS 在 Timequest Report DDR 中必须由正裕量。</p> <p>对于多个电路板而言, 如果想对不同的电路板使用相同的设计, 当到达所有 DIMM 中的同一个 DIMM 时, CK 信号与任意 DQS 信号之间的最小偏斜由下面公式表示:</p> $\underset{b}{\overset{boards}{Min}} \left[\underset{g}{\overset{groups}{Min}} [CK_{g-b} - DQS_{g-b}] \right]$
Maximum delay difference between CK and DQS	<p>到达所有 DIMM/ 器件中的同一个 DIMM/ 器件时, CK 信号与任意 DQS 信号之间的最大偏斜或最小负偏斜 (或者最大正偏斜) 由下面公式表示:</p> $\max_r \left[\max_{n,m} \{ (CK_{n,r} \text{Delay} - DQS_{m,r} \text{Delay}) \} \right]$ <p>公式中, n 是存储器时钟数, m 是 DQS 的数量, r 是 DIMM/ 器件的 rank 数量。例如, 在 dual-rank DIMM 实现中, 如果每个 rank DIMM 中有 2 对存储器时钟和 4 个 DQS 信号 (每个时钟两个), 那么 CK 与 DQS 之间的最大延迟差由下面公式表示:</p> $\max \left\{ \begin{array}{l} (CK_{1_1} \text{Delay} - DQS_{1_1} \text{Delay}), (CK_{1_1} \text{Delay} - DQS_{2_1} \text{Delay}), (CK_{2_1} \text{Delay} - DQS_{3_1} \text{Delay}), (CK_{2_1} \text{Delay} - DQS_{4_1} \text{Delay}) \\ (CK_{1_2} \text{Delay} - DQS_{1_2} \text{Delay}), (CK_{1_2} \text{Delay} - DQS_{2_2} \text{Delay}), (CK_{2_2} \text{Delay} - DQS_{3_2} \text{Delay}), (CK_{2_2} \text{Delay} - DQS_{4_2} \text{Delay}) \end{array} \right\}$ <p>此参数值影响 multi-rank 中带整平的 DDR3 接口写整平裕量。此参数值也可应用于任意 rank 数的非整平配置, 要求是 DQS 在 Timequest Report DDR 中必须由正裕量。</p> <p>对于多个电路板而言, 如果想对不同的电路板使用相同的设计, 当到达所有 DIMM 中的同一个 DIMM 时, CK 信号与任意 DQS 信号之间的最大偏斜 (或者最大正偏斜) 由下面公式表示:</p> $\underset{b}{\overset{boards}{Max}} \left[\underset{g}{\overset{groups}{Max}} [CK_{g-b} - DQS_{g-b}] \right]$
Maximum skew within DQS group	<p>DQS 组中的 DQ 与 DM 信号之间的最大偏斜。该值影响所有配置的 DDR2 和 DDR3 SDRAM 接口的读采样和写裕量 (一个或多个 chip-select, DIMM 或器件)。</p> <p>对于多个电路板, DQS 组中的 DQ 与 DM 信号之间的最大偏斜由下面公式表示:</p> $\underset{b}{\overset{boards}{Max}} \left[\underset{g}{\overset{groups}{Max}} [maxDQ_{g-b} - minDQ_{g-b}] \right]$

表 9 - 32. 电路板偏斜参数 (3/7)

参数	说明
Maximum skew between DQS groups	<p>不同 DQS 组中的 DQS 信号间的最大偏斜。该值影响没有整平的存储器接口（例如单一或多个 chip-select 配置的 DDR2 SDRAM 和分布式器件 DDR3 SDRAM）中的重同步裕量。</p> <p>对于多个电路板，如果想对不同的电路板使用相同的设计，不同 DQS 组中的 DQS 信号间的最大偏斜由下面公式表示：</p> $\frac{\text{boards}}{\text{Max}_b} \left[\frac{\text{groups}}{\text{Max}_g} [DQS_{g-b}] - \frac{\text{boards}}{\text{Min}_b} \left[\frac{\text{groups}}{\text{Min}_g} [DQS_{g-b}] \right] \right]$
Average delay difference between DQ and DQS	<p>每个 DQ 信号与 DQS 信号之间的平均延迟差，由最长和最短 DQ 信号延迟值的平均数减去 DQS 的延迟计算得出。DQ 与 DQS 之间的平均延迟差由下面公式表示：</p> $\frac{\sum_{n=1}^n \left[\left(\frac{\text{Longest DQ Path Delay in DQS}_n \text{ group} + \text{Shortest DQ Path Delay in DQS}_n \text{ group}}{2} \right) - \text{DQS}_n \text{ Path Delay} \right]}{n}$ <p>公式中，n 是 DQS 组数。对于 multi-rank 或多个 CS 配置，公式是：</p> $\frac{\sum_{r=1}^r [\text{Average delay differnt between DQ and DQS in rank r}]}{r}$
Maximum skew within address and command bus	<p>对于单一电路板，地址和命令信号间的最大偏斜由下面公式表示：</p> $0.5[(\text{MaxACdelay} - \text{MinCKdelay}) - (\text{MinACdelay} - \text{MaxCKdelay})]$ <p>对于多个电路板，如果想对不同的电路板使用相同的设计，地址和命令信号间的最大偏斜由下面公式表示：</p> $\frac{\frac{\text{boards}}{\text{Max}_b} [(\text{MaxAC}_b - \text{MinCK}_b)] - \frac{\text{boards}}{\text{Min}_b} (\text{MaxAC}_b - \text{MinCK}_b)}{2}$

表 9 - 32. 电路板偏斜参数 (4/7)

参数	说明
Average delay difference between address and command and CK	<p>一个等于最长和最短地址和命令信号延迟值的平均数减去 CK 信号的延迟的值。此值可以是正值也可以是负值。正值表示长于 CK 信号的地址和命令信号；负值表示短于 CK 信号的地址和命令信号。地址和命令信号与 CK 信号之间的平均延迟差由下面公式表示：</p> $\frac{\sum_{n=1}^n \left[\left(\frac{Longest\ AC\ Path\ Delay + Shortest\ AC\ Path\ Delay}{2} \right) - CK_n\ Path\ Delay \right]}{n}$ <p>公式中，n 是存储器时钟数。对于 multi-rank 或多个 CS 配置，公式是：</p> $\frac{\sum_{r=1}^r [Average\ delay\ differnt\ between\ AC\ and\ CK\ in\ rank\ r]}{r}$ <p>Quartus II 软件使用此偏斜来优化地址和命令信号的延迟，从而使 DDR2 和 DDR3 SDRAM 接口具有正确的建立和保持裕量。您应该从电路板仿真中得到该值。</p> <p>对于多个电路板，如果想对不同的电路板使用相同的设计，地址和命令信号与 CK 信号之间的平均延迟差由下面公式表示：</p> $Avg_b \left[\left(\frac{MaxAC_b + MinAC_b}{2} \right) - \left(\frac{MaxCK_b + MinCK_b}{2} \right) \right]$
QDR II and QDR II+	
Maximum delay difference between devices	<p>器件之间数据信号的最大延迟差由下面公式表示：</p> $Abs \left[\left(\frac{Longest\ device\ 1\ delay - Shortest\ device\ 2\ delay}{2} \right) - \left(\frac{Longest\ device\ 2\ delay - Shortest\ device\ 1\ delay}{2} \right) \right]$ <p>例如，在二器件 (two-device) 配置中有一个来自或去往最远器件（相对于最近器件）的数据信号的更大传播延迟。该参数适用于深度扩展。对于非深度扩展设计，将该值设为 0。</p>
Maximum skew within write data group (ie, K group)	参考相同 K 信号的 D 与 BWS 信号之间的最大偏斜。
Maximum skew within read data group (ie, CQ group)	参考相同 CQ 信号的 Q 信号之间的最大偏斜。
Maximum skew between CQ groups	不同读数据组的 CQ 信号之间的最大偏斜。

表 9 - 32. 电路板偏斜参数 (5/7)

参数	说明
Maximum skew within address/command bus	地址 / 命令信号之间的最大偏斜。 $0.5[(MaxACdelay - MinCKdelay) - (MinACdelay - MaxCKdelay)]$
Average delay difference between address/command and K	一个等于最长和最短地址 / 命令信号延迟值的平均数减去 K 信号的延迟的值。此值可以是正值也可以是负值。 地址和命令信号与 K 信号之间的平均延迟差由下面公式表示： $\frac{\sum_{n=1}^n \left[\left(\frac{Longest\ AC\ Path\ Delay + Shortest\ AC\ Path\ Delay}{2} \right) - K_n PathDelay \right]}{n}$ 公式中， n 代表 K 时钟数。
Average delay difference between write data signals and K	一个等于最长和最短写数据信号延迟值的平均数减去 K 信号的延迟的值。写数据信号包括 D 和 BWS 信号。此值可以是正值也可以是负值。 D 与 K 信号之间的平均延迟差由下面公式表示： $\frac{\sum_{n=1}^n \left[\left(\frac{Longest\ D\ Path\ Delay\ in\ K_n\ group + Shortest\ D\ Path\ Delay\ in\ K_n\ group}{2} \right) - K_n PathDelay \right]}{n}$ 公式中， n 代表 DQS 组数。
Average delay difference between read data signals and CQ	一个等于最长和最短读数据信号延迟值的平均数减去 CQ 信号的延迟的值。此值可以是正值也可以是负值。 Q 与 CQ 信号之间的平均延迟差由下面公式表示： $\frac{\sum_{n=1}^n \left[\left(\frac{Longest\ Q\ Path\ Delay\ in\ CQ_n\ group + Shortest\ Q\ Path\ Delay\ in\ CQ_n\ group}{2} \right) - CQ_n PathDelay \right]}{n}$ 公式中， n 代表 CQ 组数。
RLDRAM II//RLDRAM 3	
Maximum CK delay to device	从 FPGA 到任何器件 /DIMM 的最长 CK 走线的延迟由下面的公式表示： $\max_n(CK_n PathDelay)$ 公式中， n 代表存储器时钟数。例如，两对存储器时钟的最大 CK 延迟由下面的公式表示： $\max_2(CK_1 PathDelay, CK_2 PathDelay)$
Maximum DK delay to device	从 FPGA 到任何器件 /DIMM 的最长 DK 走线的延迟由下面的公式表示： $\max_n(DK_n PathDelay)$ 公式中， n 代表 DK 数。例如，两个 DK 的最大 DK 延迟由下面的公式表示： $\max_2(DK_1 PathDelay, DK_2 PathDelay)$

表 9 - 32. 电路板偏斜参数 (6/7)

参数	说明
Minimum delay difference between CK and DK	<p>到达存储器件时，CK 信号与任意 DK 信号之间的最小延迟差。该值等于 CK 信号的最小延迟减去 DK 信号的最大延迟。该值可以是正值也可以是负值。</p> <p>CK 与 DK 信号之间的最小延迟差由下面的公式表示：</p> $\min_{n,m}(CK_n PathDelay - DK_m PathDelay)$ <p>公式中，n 是存储器时钟数，m 是 DK 的数量。例如，对于 2 对存储器时钟和 4 个 DK 信号（每个时钟两个 DK 信号），CK 与 DK 之间的最小延迟差由下面公式表示：</p> $\min_{2,2} \{(CK_1 Delay - DK_1 Delay), (CK_1 Delay - DK_2 Delay), (CK_2 Delay - DK_3 Delay), (CK_2 Delay - DK_4 Delay)\}$
Maximum delay difference between CK and DK	<p>到达存储器件时，CK 信号与任意 DK 信号之间的最大延迟差。该值等于 CK 信号的最大延迟减去 DK 信号的最小延迟。该值可以是正值也可以是负值。</p> <p>CK 与 DK 信号之间的最大延迟差由下面的公式表示：</p> $\max_{n,m}(CK_n PathDelay - DK_m PathDelay)$ <p>公式中，n 是存储器时钟数，m 是 DK 的数量。例如，对于 2 对存储器时钟和 4 个 DK 信号（每个时钟两个 DK 信号），CK 与 DK 之间的最大延迟差由下面公式表示：</p> $\max_{2,2} \{(CK_1 Delay - DK_1 Delay), (CK_1 Delay - DK_2 Delay), (CK_2 Delay - DK_3 Delay), (CK_2 Delay - DK_4 Delay)\}$
Maximum delay difference between devices	<p>器件之间数据信号的最大延迟差由下面公式表示：</p> $Abs\left[\left(\frac{Longest\ device\ 1\ delay - Shortest\ device\ 1\ delay}{2}\right) - \left(\frac{Longest\ device\ 2\ delay - Shortest\ device\ 2\ delay}{2}\right)\right]$ <p>例如，在二器件（two-device）配置中有一个来自或去往最远器件（相对于最近器件）的数据信号的更大传播延迟。该参数适用于深度扩展。对于非深度扩展设计，将该值设为 0。</p>
Maximum skew within QK group	参考相同 QK 信号的 DQ 信号之间的最大偏斜。
Maximum skew between QK groups	不同数据组的 QK 信号之间的最大偏斜。
Maximum skew within address/command bus	<p>地址 / 命令信号之间的最大偏斜。</p> $0.5[(MaxACdelay - MinCKdelay) - (MinACdelay - MaxCKdelay)]$

表 9 - 32. 电路板偏斜参数 (7/7)

参数	说明
Average delay difference between address/command and CK	<p>一个等于最长和最短地址 / 命令信号延迟值的平均数减去 CK 信号的延迟的值。此值可以是正值也可以是负值。地址和命令信号与 CK 信号之间的平均延迟差由下面公式表示：</p> $\sum_{n=1}^n \left[\left(\frac{Longest\ AC\ Path\ Delay + Shortest\ AC\ Path\ Delay}{2} \right) - CK_n PathDelay \right]$ <p>公式中，n 代表 K 时钟数。</p>
Average delay difference between write data signals and DK	<p>一个等于最长和最短写数据信号延迟值的平均数减去 DK 信号的延迟的值。写数据信号包括 DQ 和 DM 信号。此值可以是正值也可以是负值。DQ 与 DK 信号之间的平均延迟差由下面公式表示：</p> $\sum_{n=1}^n \left[\left(\frac{Longest\ DQ\ Path\ Delay\ in\ DK_n\ group + Shortest\ DQ\ Path\ Delay\ in\ DK_n\ group}{2} \right) - DK_n PathDelay \right]$ <p>公式中，n 代表 DK 组数。</p>
Average delay difference between read data signals and QK	<p>一个等于最长和最短读数据信号延迟值的平均数减去 QK 信号的延迟的值。此值可以是正值也可以是负值。DQ 与 QK 信号之间的平均延迟差由下面公式表示：</p> $\sum_{n=1}^n \left[\left(\frac{Longest\ DQ\ Path\ Delay\ in\ QK_n\ group + Shortest\ DQ\ Path\ Delay\ in\ QK_n\ group}{2} \right) - QK_n PathDelay \right]$ <p>公式中，n 代表 QK 组数。</p>

控制器设置

使用此标签来应用适合您设计的控制器设置。



本章节介绍了具有高级特性的 High Performance Controller II (HPC II) 的参数设置，这些高级特性首次在 11.0 版本中出现并应用在版本 11.0 及后续版本中生成的设计中。在更早版本中创建并在版本 11.0 或后续版本中重新生成的设计不继承这些新的高级特性；关于不具有高级特的 HPC II 的参数设置的详细信息，请参考 Altera 网站上关于 Quartus II 10.1 的 [Literature: External Memory Interfaces](#)。

表 9-33 列出了控制器设置。

表 9-33. 控制器设置 (1/3)

参数		说明
LPDDR2/DDR2/DDR3 SDRAM		
Avalon Interface	Generate power-of-2 bus widths for SOPC Builder	将 Avalon-MM 侧数据总线向下取整到最接近的 2 次幂。您必须对 Qsys 以及 SOPC Builder 系统使能此选项。 此选项使能时，Avalon 数据总线被截短成 256 bit 宽。一个 256 bit 宽的 Avalon read-write 传输映射到四个存储器差拍 (beat) 传输，每个传输 72 bit (8 MSB bit 为零，64 LSB bit 为数据)。四个存储器差拍可包括一个完整的突发长度为 4 的传输，或者突发长度为 8 的传输的一部分。
	Generate SOPC Builder compatible resets	如果 IP core 要在 SOPC Builder 系统中使用，那么必须使能此选项。使能后，复位输入关联到 PLL 参考时钟，路径必须被截断。使用 SOPC Builder 时必须使能此选项，但使用 MegaWizard Plug-in Manager 或 Qsys 时不必。
	Maximum Avalon-MM burst length	指定 Avalon-MM 总线上的最大突发长度。影响 AVL_SIZE_WIDTH 参数。
	Enable Avalon-MM byte-enable signal	使能此选项时，控制器会添加字节使能信号 (avl_be) 使 Avalon-MM 总线控制连接存储器接口的数据屏蔽 (mem_dm) 管脚。如果使能此选项，那么也必须同时使能 Enable DM pins 。 关闭此选项后，字节使能信号 (avl_be) 对 Avalon-MM 总线是禁用的，默认情况下所有的字节都被使能。然而，如果此选项关闭时开启 Enable DM pins ，那么全部 write word 会被写入。
	Avalon interface address width	Avalon-MM 接口上的地址位宽。
	Avalon interface data width	Avalon-MM 接口上的数据位宽。
Low Power Mode	Enable self-refresh controls	在控制器顶层设计上使能自刷新 (self-refresh) 信号。这些控制使您能够控制何时将存储器置于 self-refresh 模式。
	Enable auto-power down	指定数量的空闲周期过后，允许控制器自动将存储器置于 power-down 模式。在 auto-power down cycles 参数中指定控制器对存储器断电前的空闲周期数。
	Auto power-down cycles	控制器对存储器自动断电前的空闲控制器时钟周期数。合法范围是 1 到 65,535 控制器时钟周期。

表 9 - 33. 控制器设置 (2/3)

参数		说明
Efficiency	Enable user auto-refresh controls	在控制器顶层使用户自动刷新 (auto-refresh) 控制信号。这些控制器信号使您能够控制控制器何时发出存储器自动刷新命令。
	Enable auto-precharge control	在控制器顶层使用户自动预充电 (autoprecharge) 控制信号。请求读或写突发时，置位 autoprecharge 控制信号使您能够指定控制器是否应该在读或写突发的结束时关闭 (autoprecharge) 当前打开页。
	Local-to-memory address mapping	使您能够控制 Avalon-MM 接口上 address bit 之间的映射，和存储器上 chip、row、bank 和 column bit 之间的映射。 选择 Chip-Row-Bank-Col 以提高有序传输 (sequential traffic) 的效率。 选择 Chip-Bank-Row-Col 以提高随机传输 (random traffic) 的效率。 选择 Row-Chip-Bank-Col 以提高多个芯片选择和有序传输的效率。
	Command queue look-ahead depth	选择先行深度 (look-ahead depth) 值以控制 look-ahead bank 管理逻辑检验的读或写请求的数量。较大值可能会提高 bank 管理的效率，但会使用更多的资源。较小值可能会效率较低，但同时也使用更少的资源。该值的有效范围为 1 到 16。
	Enable reordering	使控制器能够执行命令和数据重排序，降低总线周转时间和 row/bank 切换时间以提高控制器效率。
	Starvation limit for each command	指定等待命令执行前能够执行的命令数。该值的有效范围为 1 到 63。
Configuration, Status, and Error Handling	Enable Configuration and Status Register Interface	使能存储控制器实时配置和状态读取接口。此选项添加一个额外的 Avalon-MM 从端口到存储控制器顶层，您可以使用此端口修改或读取存储器时序参数，存储器地址大小，模式寄存器设置和控制器状态。如果使能 Error Detection and Correction Logic，同一从端口使您能够控制并检索该逻辑的状态。
	CSR port host interface	指定到 CSR 端口的连接类型。此端口能够被导出和内部连接到 JTAG Avalon Master。 选择 Internal (JTAG) ，导出 CSR 端口。 选择 Avalon-MM Slave ，连接 CSR 端口到 JTAG Avalon Master。 选择 Shared ，导出并连接 CSR 端口到 JTAG Avalon Master。
	Enable error detection and correction logic	使能 ECC 执行 single-bit 错误纠正和 double-bit 错误检测。要使用 ECC，您的存储器接口位宽必须是 40 或 72 的倍数。
	Enable auto error correction	当 ECC 逻辑检测到 single-bit 错误时，允许控制器执行自动纠正。
Advanced Controller Features	Enable half rate bridge	开启此选项，使能半速率桥接模块。
	Enable hard memory controller	开启此选项，使能硬核存储控制器。

表 9 - 33. 控制器设置 (3/3)

参数		说明
Multiple Port Front End	Export bonding port	开启此选项来导出绑定接口，用两个控制器获得更大 avalon 数据位宽。绑定端口被导出至顶层。
	Number of ports	指定要导出的 Avalon-MM 从端口数。端口数取决于所选择的端口宽度和类型。在多端口前端 (MPFE) 器件中有四个 64-bit read FIFO 和四个 64-bit write FIFO。例如，如果选择 256 bit 宽双向从端口，那么将充分使用所有的 FIFO，因此您可以只选一个端口。
	Width	对每个 Avalon-MM 从端口指定本地数据位宽。位宽取决于从端口的类型和所选端口的数量。这是由于 MPFE 中 FIFO 数量的限制。MPFE 中有四个 64-bit read FIFO 和四个 64-bit write FIFO。例如，如果选择一个双向从端口，那么您可以选择最高 256 bit，以使用所有的读和写 FIFO。 作为对您半速率或四分之一速率的设计选择一个最优端口宽度的一般指南，请应用以下公式： $\text{port width} = 2 \times \text{DQ width} \times \text{Interface width multiplier}$ 其中接口宽度乘数 (interface width multiplier) 是 2 (半速率接口) 和 4 (四分之一速率接口)。
	Priority	对 Avalon-MM 从端口指定绝对优先级。任何具有较高优先级的端口传输都将在具有较低优先级的端口传输之前进行。
	Weight	对 Avalon-MM 从端口指定相对优先级。当两个或更多端口具有相同绝对优先级时，首先执行具有较高 (较大值) 相对权重的端口传输。可设置的权重范围从 0 到 32。
	Type	指定 Avalon MM 从端口的类型为双向端口，只读端口或只写端口。
QDR II/QDR II+ SRAM and RLDRAM II		
Generate power-of-2 data bus widths for SOPC Builder		将 Avalon-MM 侧数据总线向下取整到最接近的 2 次幂。您必须对 Qsys 以及 SOPC Builder 系统使能此选项。
Generate SOPC Builder compatible resets		如果此内核要用于 SOPC Builder 系统中，则必须使能此选项。
Maximum Avalon-MM burst length		指定 Avalon-MM 总线上的最大突发长度。
Enable Avalon-MM byte-enable signal		开启此选项时，控制器对 Avalon-MM 总线添加一个字节使能信号 (avl_be_w)，在写操作期间控制存储器上的 bws_n 信号屏蔽字节。 关闭此选项时，avl_be_w 信号不可用，控制器将始终驱动存储器 bws_n 信号，以便在写操作期间不会屏蔽任何字节。
Avalon interface address width		指定 Avalon-MM 接口上的地址位宽。
Avalon interface data width		指定 Avalon-MM 接口上的数据位宽。
Reduce controller latency by		指定时钟周期数，以此时钟周期数来减少控制器延迟。 较低的控制延迟导致较低的资源使用和 f_{MAX} ，而较高的延迟导致较高的资源使用和 f_{MAX} 。
Enable user refresh		使用户控制的刷新。刷新信号优先于读 / 写请求。 此选项仅适用于 RLDRAM II。
Enable error detection parity		使能每字节奇偶校验保护。 此选项仅适用于 RLDRAM II。

诊断

Diagnostics 标签用于设置某些诊断功能的参数。

表 9 - 34 描述了用于仿真的参数。

表 9 - 34. 仿真选项

参数	说明
Simulation Options	
Auto-calibration mode	<p>指定是否通过减少校准来提高仿真性能。生成的 RTL 没有变化。可使用下面的自动校准模式：</p> <ul style="list-style-type: none">■ Skip calibration— 提供最快仿真。加载从存储器配置计算的设置，并进入用户模式。■ Quick calibration— 进入用户模式前校准（没有集中）每组一个比特。■ Full calibration— 与硬件中的校准相同，包括全部相位，延迟扫描和集中于每个数据比特。你可以使用带时序注释的存储器模型。要知道完全校准需要几个小时甚至几天来完成。 <p>要执行正确 PHY 仿真，需要选择 Quick calibration 或 Full calibration。关于详细信息，请参考 <i>Simulating Memory IP</i> 章节中的“Simulation Options”部分。</p> <p>对于 QDR II, QDR II+ SRAM 和 RLDRAM II, 必须选择基于 Nios II 的 sequencer 以使能自动校准模式。</p>
Skip memory initialization delays	当开启此选项，特定的存储器初始化命令之间的所需延迟被跳过以加速仿真。
Enable verbose memory model output	开启此选项以显示关于仿真期间每个存储器访问的详细信息。
Enable support for Nios II ModelSim® flow in Eclipse	<p>初始化存储器接口以使用 Eclipse 中的 Run as Nios II ModelSim 流程。</p> <p>此选项不适用于 QDR II 和 QDR II+ SRAM。</p>
Debug Options	
Debug level	指定存储器接口的调试级别。
Efficiency Monitor and Protocol Checker Settings	
Enable the Efficiency Monitor and Protocol Checker on the Controller Avalon Interface	<p>使能控制器 Avalon 接口上的效率监督模块和协议检查模块。</p> <p>此选项不适用于 QDR II 和 QDR II+ SRAM。</p>

文档修订历史

表 9 - 35 列出了本文档的修订历史。

表 9 - 35. 文档修订历史

日期	版本	修订内容
2012 年 11 月	6.0	<ul style="list-style-type: none"> ■ 添加了 RLD RAM 3 信息。 ■ 添加了 LPDDR2 信息。 ■ 从第 8 章改成第 9 章。
2012 年 6 月	5.0	<ul style="list-style-type: none"> ■ 在时钟参数表中添加了共享接口参数的数量。 ■ 在电路板偏斜参数表中添加了关于 DQ/DQS Package Deskew 和 Address/Command Package Deskew 的说明。 ■ 在电路板偏斜参数表中的几个参数说明中添加了多电路板的公式。 ■ 添加了“反馈”图标。
2011 年 11 月	4.0	<ul style="list-style-type: none"> ■ 更新了安装和许可授权部分。 ■ 合并了 Qsys 和 SOPC Builder 接口部分。 ■ 对于 ALTMEMPHY 以及 UniPHY IP，合并了 DDR, DDR2, DDR3 SDRAM, QDRII SRAM 和 RLD RAM II 的参数设置。 ■ 在参数化基于 UniPHY IP 的存储控制器章节添加了参数使用详情。 ■ 将 DDR, DDR2, DDR3 SDRAM, QDRII SRAM 和 RLD RAM II 的“Functional Description”章节移到 <i>External Memory Interface Handbook</i> 的卷 3 中。
2011 年 6 月	3.0	<ul style="list-style-type: none"> ■ 删除了 High-Performance Controller 参考。 ■ 更新了 High-Performance Controller II 信息。 ■ 删除了 HardCopy III, HardCopy IV E, HardCopy IV GX, Stratix III 和 Stratix IV 支持。 ■ 更新了生成的文件列表。 ■ 添加了 Qsys 和 SOPC Builder 接口部分。
2010 年 12 月	2.1	<p>针对 10.1 更新了以下内容：</p> <ul style="list-style-type: none"> ■ 更新了设计流程和生成的文件信息。 ■ 更新了参数化基于 UniPHY IP 的存储控制器章节。
2010 年 7 月	2.0	<ul style="list-style-type: none"> ■ 添加了关于新的 GUI 参数信息：Controller latency, Enable reduced bank tracking for area optimization 和 Number of banks to track。 ■ 删除了关于 IP Advisor 的信息。该特性从 DDR/DDR2 SDRAM IP support for version 10.0 中删除。
2010 年 2 月	1.3	纠正错别字
2010 年 2 月	1.2	<ul style="list-style-type: none"> ■ 对 Stratix IV 器件的全面支持。 ■ 添加了 HPC 初始化和校准阶段的时序图。
2009 年 11 月	1.1	少量修改
2009 年 11 月	1.0	首次发布