

LogiCORE™ IP Initiator/Target v3.1 for PCI™

入门指南

UG157 2007 年 10 月 10 日





Xilinx is disclosing this Document and Intellectual Property (hereinafter "the Design") to you for use in the development of designs to operate on, or interface with Xilinx FPGAs. Except as stated herein, none of the Design may be copied, reproduced, distributed, republished, downloaded, displayed, posted, or transmitted in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Xilinx. Any unauthorized use of the Design may violate copyright laws, trademark laws, the laws of privacy and publicity, and communications regulations and statutes.

Xilinx does not assume any liability arising out of the application or use of the Design; nor does Xilinx convey any license under its patents, copyrights, or any rights of others. You are responsible for obtaining any rights you may require for your use or implementation of the Design. Xilinx reserves the right to make changes, at any time, to the Design as deemed desirable in the sole discretion of Xilinx. Xilinx assumes no obligation to correct any errors contained herein or to advise you of any correction if such be made. Xilinx will not assume any liability for the accuracy or correctness of any engineering or technical support or assistance provided to you in connection with the Design.

THE DESIGN IS PROVIDED "AS IS" WITH ALL FAULTS, AND THE ENTIRE RISK AS TO ITS FUNCTION AND IMPLEMENTATION IS WITH YOU. YOU ACKNOWLEDGE AND AGREE THAT YOU HAVE NOT RELIED ON ANY ORAL OR WRITTEN INFORMATION OR ADVICE, WHETHER GIVEN BY XILINX, OR ITS AGENTS OR EMPLOYEES. XILINX MAKES NO OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE DESIGN, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT OF THIRD-PARTY RIGHTS.

IN NO EVENT WILL XILINX BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOST DATA AND LOST PROFITS, ARISING FROM OR RELATING TO YOUR USE OF THE DESIGN, EVEN IF YOU HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE TOTAL CUMULATIVE LIABILITY OF XILINX IN CONNECTION WITH YOUR USE OF THE DESIGN, WHETHER IN CONTRACT OR TORT OR OTHERWISE, WILL IN NO EVENT EXCEED THE AMOUNT OF FEES PAID BY YOU TO XILINX HEREUNDER FOR USE OF THE DESIGN. YOU ACKNOWLEDGE THAT THE FEES, IF ANY, REFLECT THE ALLOCATION OF RISK SET FORTH IN THIS AGREEMENT AND THAT XILINX WOULD NOT MAKE AVAILABLE THE DESIGN TO YOU WITHOUT THESE LIMITATIONS OF LIABILITY.

The Design is not designed or intended for use in the development of on-line control equipment in hazardous environments requiring fail-safe controls, such as in the operation of nuclear facilities, aircraft navigation or communications systems, air traffic control, life support, or weapons systems ("High-Risk Applications"). Xilinx specifically disclaims any express or implied warranties of fitness for such High-Risk Applications. You represent that use of the Design in such High-Risk Applications is fully at your risk.

© 2000–2007 Xilinx, Inc. All rights reserved. XILINX, the Xilinx logo, and other designated brands included herein are trademarks of Xilinx, Inc. PowerPC is a trademark of IBM, Inc. All other trademarks are the property of their respective owners.

修订历史

下表说明此技术文档的修订历史。

日期	版本	修订
2000 年 6 月 1 日	1.0	Xilinx 最初版本。
2001 年 4 月 11 日	2.0	修订文档格式，以使用 FrameMaker 6.0 功能。
2002 年 6 月 24 日	3.0	全公司共用模板集的 Xilinx 最初版本。
2004 年 11 月 11 日	3.5	增加有关安装和许可的章节。
2004 年 12 月 1 日	3.6	更新以包括 Virtex-4 支持。
2005 年 3 月 7 日	3.7	更新到 ISE 7.1i 和内部版本号 3.0.145。

日期	版本	修订
2005 年 5 月 13 日	4.0	更新到 ISE 7.1i SP 2，内部版本号 3.0.150。
2005 年 8 月 31 日	5.0	更新到 ISE 7.1i SP3 和内部版本号 3.0.151。
2005 年 9 月 12 日	6.0	更新到内部版本 3.0.152，删除验证目录结构的指令。
2006 年 1 月 18 日	7.0	更新到 ISE 8.1i，内部版本 3.0.155。
2006 年 2 月 14 日	8.0	更新以增加 ISE 8.1i 的 SP2，内部版本更新到 158，从第 3 章的“支持器件表”中删除 2S100。
2006 年 7 月 13 日	9.0	更新到 ISE v8.21，内部版本更新到 160。
2007 年 2 月 15 日	10.0	工具 ISE 版本更新到 9.2i，内部版本更新到 161。
2007 年 5 月 17 日	10.5	修改所有 PCI 参考，以符合 PCI-SIG 商标指导原则。将对 Cadence IUS 的支持提高到版本 5.7。
2007 年 8 月 8 日	11.0	为发布 IP1 Jade Minor 而更新，将对器件的支持提高到版本 3.2。更新 Spartan-3 FPGA 系列的 VCCO 要求。
2007 年 10 月 10 日	11.5	为发布 IP2 Jade Minor 而更新，为 LogiCORE 参考增加 IP，将法律免责声明更换为现用版本。

前言：关于本指南

指南内容	9
附加资源	10
惯例	10
排版	10
联机技术文档	11

第 1 章：入门

系统要求	13
关于示例设计	13
其他技术文档	14
技术支持	14
反馈	14
PCI 接口核	14
技术文档	14

第 2 章：核许可

开始前须知	15
许可选项	15
评估版	15
完全版	15
直接下载	16
安装许可文件	16

第 3 章：针对系列的考虑事项

设计支持	17
器件初始化	30
配置引脚	30
总线宽度检测	30
数据通路输出时钟使能	31
输入延迟缓冲器	32
区域时钟用法	33
总线时钟用法	35
电气兼容性	36
生成比特流	39

第 4 章：功能仿真

Cadence IUS	41
Mentor Graphics ModelSim	42
Verilog	42
VHDL	43

第 5 章：综合设计

Synplicity Synplify	45
---------------------	----

Verilog	45
VHDL	49
Exemplar LeonardoSpectrum	53
Xilinx XST	54

第 6 章：实现设计

ISE Foundation	55
使用引导文件	56

插图一览表

第 1 章：入门

第 2 章：核许可

第 3 章：针对系列的考虑事项

图 3-1: SLOT64 样本信号的生成	31
图 3-2: 区域时钟控制图解	34
图 3-3: 3.3V 输入缓冲器的兼容性关系	37
图 3-4: 3.0V 输出驱动器的 VCC0 生成	38
图 3-5: Spartan-3 和 Spartan-3E 的 3.3V 输出驱动器的 VCC0 生成	38

第 4 章：功能仿真

第 5 章：综合设计

图 5-1: 创建新文档	45
图 5-2: 主项目窗口	46
图 5-3: 选择要添加的文件（库）.....	46
图 5-4: 要添加的文件（LogiCORE IP 文件）.....	47
图 5-5: 选择要添加的文件（用户应用）.....	47
图 5-6: 显示源文件的主项目窗口.....	48
图 5-7: 实现选项：器件	49
图 5-8: 创建新文档	49
图 5-9: 项目窗口	50
图 5-10: 选择要添加的文件（库）.....	51
图 5-11: 选择要添加的文件（LogiCORE IP 文件）.....	51
图 5-12: 显示源文件的项目窗口	52
图 5-13: 实现选项	53

第 6 章：实现设计

关于本指南

《LogiCORE™ IP Initiator/Target v3.1 for PCI 入门指南》提供经过全面验证的 32 位和 64 位预实现 PCI 总线接口。本指南讲述支持的基于 Virtex™ 和 Spartan™ 架构的 32 位和 64 位核的设计流程，并且介绍 Cadence® IUS v5.8 中的示例设计。

指南内容

本手册包含以下各章：

- 第 1 章 “入门” 讲述核及有关信息，包括附加资源、技术支持以及向 Xilinx 提交反馈。
- 第 2 章 “核许可” 提供有关核安装和许可的信息。
- 第 3 章 “针对系列的考虑事项” 讲述针对指向 Virtex 和 Spartan 器件的核的设计考虑事项。
- 第 4 章 “功能仿真” 讲述如何使用支持的功能仿真工具（包括 Cadence® IUS 和 Mentor Graphics® ModelSim®）进行示例设计仿真。
- 第 5 章 “综合设计” 讲述如何使用支持的综合工具（包括 Synplicity® Synplify®、Exemplar Leonardo Spectrum 和 Xilinx XST）综合示例设计。
- 第 6 章 “实现设计” 讲述如何使用 ISE™ 软件附带的支持的 FPGA 实现工具实现示例设计。

附加资源

有关进一步信息，请访问 www.xilinx.com/cn/support。下表列出了可以从此网站访问的部分资源。也可以使用提供的网址直接访问这些资源。

资源	描述 / 网址
辅导资料	涵盖 Xilinx 设计流程（从设计输入到验证和调试）的辅导资料 www.xilinx.com/cn/support/techsup/tutorials/index.htm
问答浏览器	Xilinx 解决方案记录数据库 http://www.xilinx.com/cn/support/answers/index.htm
应用指南	针对器件的设计技巧与方法描述 http://www.xilinx.com/cn/support/documentation/application_notes.htm
数据手册	针对器件的关于 Xilinx 器件特性（包括读回、边界扫描、配置、长度计数和调试）的信息 http://www.xilinx.com/cn/support/documentation/data_sheets.htm
设计疑难解答	解答设计问题的互动工具 www.xilinx.com/cn/support/troubleshoot/psolvers.htm

惯例

排版

本技术文档使用以下排版惯例：

惯例	含义或用途	示例
Courier 字体	系统显示的信息、提示和程序文件	速度等级：- 100
Courier 粗体	在语法语句中输入的文字命令	ngdbuild design_name
<i>斜体</i>	需要赋值的语法语句中的变量	ngdbuild design_name
	参考其他手册	有关更多信息，请见《开发系统参考手册》。
	强调文字	如果拉一条导线跨接某符号表示的引脚，则这两个网不连接。
方括号 []	可选字条或参数。但是，在总线技术指标（如 bus [7:0] ）中，方括号内必须输入内容。	ngdbuild [option_name] design_name
大括号 { }	需要从中选择一项或多项的项目列表	lowpwr = {on off}
竖线	分隔选择列表中的项目	lowpwr = {on off}

惯例	含义或用途	示例
纵向省略符 . . .	省略的重复资料	IOB #1: Name = QOUT' IOB #2: Name = CLKIN' . . .
横向省略符 ...	省略的重复资料	允许模块 <i>block_name loc1 loc2</i> ... <i>locn</i> ;

联机技术文档

本技术文档使用以下惯例：

惯例	含义或用途	示例
蓝色文字	当前技术文档中某位置的前后参照链接	有关详情，请见“附加资源”。 有关详情，请见第 1 章中的“标题格式”。
<u>蓝色下划线文字</u>	网站超级链接（网址）	请转到 www.xilinx.com/cn 获取最新速度文件。

入门

Initiator/Target core for PCI 提供经过全面验证的 32 位和 64 位预实现 PCI 总线接口，支持 33 MHz 和 66 MHz 工作频率。本指南详述支持的指向 Virtex 和 Spartan 架构器件的 32 位和 64 位接口的设计流程。Cadence IUS 中提供的示例设计供您用来对接口进行仿真、综合与实现，以理解 PCI 设计流程。

系统要求

Windows

- Windows® 2000 Professional，带服务包 2–4
- Windows XP Professional，带服务包 1

Solaris/Linux

- Sun Solaris® 9/10
- Red Hat Enterprise Linux 3.0（32 位和 64 位）

软件

- ISE™ 9.2i，带适用的服务包

有关所需服务包，请查阅版本说明；可以从 <http://www.xilinx.com/cn/support/download/index.htm> 下载 ISE 服务包。

关于示例设计

示例设计是一个简单的用户应用，用作培训工具并用于设计流程测试。示例设计由用户应用 *Ping* 以及仿真和实现的支持文件组成。*ping32* 设计附带 32 位核，*ping64* 设计附带 64 位核。本技术文档中的示例涉及 *ping64*。如果使用 32 位核，请用 *ping32* 替换 *ping64*。

Ping 设计包括一个可以生成简单读写事务的测试平台。这一激励生成功能用来建立设计的配置空间，然后执行一些简单事务。另外还提供了一个专用的配置文件，测试平台假设所用基址寄存器的大小和数量。

用户可以改变与实现有关的核选项（即与所选 FPGA 架构有关的选项）。但是，用户不得改变影响 Initiator/Target core for PCI 功能行为的核选项，否则会在示例设计的仿真中产生不可预见的结果。对于定制设计，用户可以按《LogiCORE IP Initiator/Target v3.1 for PCI 用户指南》中所述更改核配置。

请按提供的循序渐进说明使用支持的设计工具对 *Ping* 示例设计进行仿真、综合与实现。

其他技术文档

有关 Initiator/Target core for PCI 的更多信息，请见 zip 文件中的以下技术文档：

- 《Initiator/Target v3.1 for PCI 版本说明》
- 《Initiator/Target v3.1 for PCI 用户指南》

有关进一步信息，请参阅《[Mindshare PCI 系统架构](#)》文档和 [PCI 特别利益集团](#) 网站提供的《PCI 局域总线规范》。

技术支持

有关技术支持，请访问 www.xilinx.com/cn/support。您的问题将提交给具有使用 Initiator/Target core for PCI 专业知识的工程师小组。

Xilinx 为按照《Initiator/Target v3.1 for PCI 用户指南》和《Initiator/Target v3.1 for PCI 入门指南》中所述使用此产品提供技术支持。对于不遵循这些指导原则的设计，Xilinx 无法保证此产品的时序、功能和可支持性。

反馈

Xilinx 欢迎有关 Initiator/Target core for PCI 及其所附技术文档的意见和建议。

PCI 接口核

对于有关 Initiator/Target core for PCI 的意见和建议，请在 www.xilinx.com/cn/support/clearexpress/websupport.htm 提交 WebCase。请务必包括以下信息：

- 产品名称
- 核版本号
- 意见说明

技术文档

对于有关本技术文档的意见或建议，请在 www.xilinx.com/cn/support/clearexpress/websupport.htm 提交 WebCase。请务必包括以下信息：

- 技术文档标题和编号
- 意见所涉及的页码
- 意见说明

核许可

本章提供有关安装和获得 Initiator/Target core for PCI 许可的说明，必须获得并安装许可才能将核用于设计。此核是依据 [Xilinx LogiCORE IP 现场许可协议](#) 或 [Xilinx LogiCORE IP 项目许可协议](#) 之条款提供的，这些协议符合公共许可证联盟制定的 [SignOnce](#) IP 许可 / 项目标准的规定。通过购买此核，您将在一年期内享有技术支持和更新的权利。

重要提示：请访问下载区，以确认您所使用的核是最新版本。您可以随时从产品下载区下载核的最新版本。

开始前须知

本章假定您已经使用 **CORE Generator™** 软件更新安装程序或通过手动安装了核。有关安装核的信息，请见 [Initiator/Target core for PCI](#) 产品下载区。

许可选项

评估版

获得评估版许可的方法取决于您所选核的版本。

- 对于 32/33 位 Virtex 和 Spartan 核，请访问 www.xilinx.com/cn/ipcenter/ipevaluation，在 Xilinx IP 评估版页上注册。可以从此位置获得访问权，然后生成您自己的许可。
- 对于 64/66 位核，请与您当地的现场应用工程师联系，向其申请“完全系统硬件评估版”许可密钥。

完全版

完全版许可在您购买核时提供，授予您在仿真和硬件中使用所有核功能的完全使用权，其中包括：

- 逻辑门级功能仿真支持
- 逆向注解逻辑门级功能仿真支持
- 包括布局、布线和比特流生成在内的完全实现支持
- 已编程器件内的无限时完全功能

获得完全版许可

购买后，可以从产品下载区下载 Initiator/Target core for PCI 完全版许可。要创建并下载用于 CORE Generator 软件的许可文件，请执行以下步骤：

1. 购买后，您会收到包含序列号的函，用该序列号注册即可获得 下载区（产品页的安全区）访问权。请转到 www.xilinx.com/cn/products/logiccore/lounge/lounge.htm，然后选择下载核的相应连接。
2. 在产品页上，单击“注册”进行注册并申请下载区访问权。Xilinx 会审查您的访问申请，通常会在 48 小时内授予下载区访问权。

注：如果您希望缩短处理周期，请与 Xilinx 客户服务部门联系 (www.xilinx.com/cn/support/services/contact_info.htm)。

3. 收到下载区访问权确认信息后，请在产品页上单击“访问下载区”进行登录。
4. 下载区起始页上端有一个链接，您可以用此链接生成许可证。单击此链接，按照说明填写许可申请表，然后单击“提交”生成许可证。您立刻会收到一封电子邮件，其中包含许可和安装说明。

直接下载

直接下载 Initiator/Target core for PCI 时不需要 CORE Generator 软件许可。

安装许可文件

选择许可选项之后，您会收到一封电子邮件，其中包括安装许可文件的说明。另外，该电子邮件还提供有关于高级许可选项和技术支持的信息。

针对系列的考虑事项

本章提供针对以 Virtex 和 Spartan 系列器件为目标的 Initiator/Target core for PCI 的重要设计信息。

设计支持

表 3-1 提供了所支持器件和接口组合的列表。该表中的每行包括一种器件、一种总线接口类型和两三个特定实现文件。

表 3-1: 器件和接口选择表

支持的器件	总线类型	封装文件	约束文件 / 引导文件
Spartan-II			
2S50-PQ208-5C	33 MHz 5.0V 32 位	pcim_lc_33_5_s	2s050pq208_32_33.ucf 无引导文件
2S50-PQ208-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s050pq208_32_33.ucf 无引导文件
2S100-PQ208-5C	33 MHz 5.0V 32 位	pcim_lc_33_5_s	2s100pq208_32_33.ucf 无引导文件
2S100-PQ208-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s100pq208_32_33.ucf 无引导文件
2S150-PQ208-5C	33 MHz 5.0V 32 位	pcim_lc_33_5_s	2s150pq208_32_33.ucf 无引导文件
2S150-PQ208-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s200pq208_32_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
2S150-FG456-6C	33 MHz 5.0V 64 位	pcim_lc_33_5_s	2s150fg456_64_33.ucf 无引导文件
2S150-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s150fg456_64_33.ucf 无引导文件
2S150-FG456-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	2s150fg456_64_66.ucf 2s150fg456_64_66.ncd
2S200-PQ208-5C	33 MHz 5.0V 32 位	pcim_lc_33_5_s	2s200pq208_32_33.ucf 无引导文件
2S200-PQ208-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s150pq208_32_33.ucf 无引导文件
2S200-FG456-6C	33 MHz 5.0V 64 位	pcim_lc_33_5_s	2s200fg456_64_33.ucf 无引导文件
2S200-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s200fg456_64_33.ucf 无引导文件
2S200-FG456-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	2s200fg456_64_66.ucf 2s200fg456_64_66.ncd
Spartan-II E			
2S50E-PQ208-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s050epq208_32_33.ucf 无引导文件
2S100E-PQ208-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s100epq208_32_33.ucf 无引导文件
2S100E-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s100efg456_64_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
2S150E-PQ208-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s150epq208_32_33.ucf 无引导文件
2S150E-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s150efg456_64_33.ucf 无引导文件
2S200E-PQ208-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s200epq208_32_33.ucf 无引导文件
2S200E-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s200efg456_64_33.ucf 无引导文件
2S300E-PQ208-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2s300epq208_32_33.ucf 无引导文件
2S300E-FG456-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2s300efg456_64_33.ucf 无引导文件
2S300E-FG456-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	2s300efg456_64_66.ucf 2s300efg456_64_66.ncd
Spartan-3			
3S1000-FG456-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s1000fg456_32_33.ucf 无引导文件
3S1000-FG456-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s1000fg456_64_33.ucf 无引导文件
Spartan-3E			
3S500E-FT256-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s500eft256_32_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
3S500E-FT256-4C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s500eft256_32_66r.ucf 无引导文件
3S1200E-FG400-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s1200efg400_32_33.ucf 无引导文件
3S1200E-FG400-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s1200efg400_64_33.ucf 无引导文件
3S1200E-FG400-5C ^a	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s1200efg400_32_66r.ucf 无引导文件
3S1200E-FG400-5C ^a	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3s1200efg400_64_66r.ucf 无引导文件
Spartan-3A			
3S400A-FG400-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s400afg400_32_33r.ucf 无引导文件
3S400A-FG400-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s400afg400_64_33r.ucf 无引导文件
3S400A-FG400-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s400afg400_32_66r.ucf 无引导文件
3S400A-FG400-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3s400afg400_64_66r.ucf 无引导文件
3S700A-FG400-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s700afg400_32_33r.ucf 无引导文件
3S700A-FG400-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s700afg400_64_33r.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
3S700A-FG400-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s700afg400_32_66r.ucf 无引导文件
3S700A-FG400-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3s700afg400_64_66r.ucf 无引导文件
3S700A-FG484-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s700afg484_32_33r.ucf 无引导文件
3S700A-FG484-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s700afg484_64_33r.ucf 无引导文件
3S700A-FG484-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s700afg484_32_66r.ucf 无引导文件
3S700A-FG484-5C/I	66 MHz 3.3 64 位	pcim_lc_66_3_s	3s700afg484_64_66r.ucf 无引导文件
3S1400A-FG484-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s1400afg484_32_33r.ucf 无引导文件
3S1400A-FG484-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s1400afg484_64_33r.ucf 无引导文件
3S1400A-FG484-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s1400afg484_32_66r.ucf 无引导文件
3S1400A-FG484-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3s1400afg484_64_66r.ucf 无引导文件
3S1400A-FG676-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s1400afg676_32_33r.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
3S1400A-FG676-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3s1400afg676_64_33r.ucf 无引导文件
3S1400A-FG676-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s1400afg676_32_66r.ucf 无引导文件
3S1400A-FG676-5C/I	66 MHz 3.3 64 位	pcim_lc_66_3_s	3s1400afg676_64_66r.ucf 无引导文件
Spartan-3AN			
3S400AN-FGG400-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s400anfgg400_32_33r.ucf 无引导文件
3S400AN-FGG400-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s400anfgg400_32_66r.ucf 无引导文件
3S700AN-FGG484-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s700anfgg484_32_33r.ucf 无引导文件
3S700AN-FGG484-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s700anfgg484_32_66r.ucf 无引导文件
3S1400AN-FGG676-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3s1400anfgg676_32_33r.ucf 无引导文件
3S1400AN-FGG676-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3s1400anfgg676_32_66r.ucf 无引导文件
Spartan-3ADSP			
3SD1800A-FG676-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3sd1800afg676_32_33r.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
3SD1800A-FG676-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3sd1800afg676_64_33.ucf 无引导文件
3SD1800A-FG676-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3sd1800afg676_32_66.ucf 无引导文件
3SD1800A-FG676-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3sd1800afg676_64_66.ucf 无引导文件
3SD3400A-FG676-4C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	3sd3400afg676_32_33.ucf 无引导文件
3SD3400A-FG676-4C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	3sd3400afg676_64_33.ucf 无引导文件
3SD3400A-FG676-5C/I	66 MHz 3.3V 32 位	pcim_lc_66_3_s	3sd3400afg676_32_66.ucf 无引导文件
3SD3400A-FG676-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	3sd3400afg676_64_66.ucf 无引导文件
Virtex			
V200-FG256-6C	66 MHz 3.3V 32 位	pcim_lc_66_3_d	v200fg256_32_66.ucf v200fg256_32_66.ncd
V300-BG432-5C	33 MHz 5.0V 32 位	pcim_lc_33_5_s	v300bg432_32_33.ucf 无引导文件
V300-BG432-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	v300bg432_32_33.ucf 无引导文件
V300-BG432-5C	33 MHz 5.0V 64 位	pcim_lc_33_5_s	v300bg432_64_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
V300-BG432-5C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	v300bg432_64_33.ucf 无引导文件
V300-BG432-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	v300bg432_64_66.ucf v300bg432_64_66.ncd
V1000-FG680-5C	33 MHz 5.0 V 32 位	pcim_lc_33_5_s	v1000fg680_32_33.ucf 无引导文件
V1000-FG680-5C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	v1000fg680_32_33.ucf 无引导文件
V1000-FG680-5C	33 MHz 5.0V 64 位	pcim_lc_33_5_s	v1000fg680_64_33.ucf 无引导文件
V1000-FG680-5C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	v1000fg680_64_33.ucf 无引导文件
V1000-FG680-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	v1000fg680_64_66.ucf v1000fg680_64_66.ncd
Virtex-E			
V100E-BG352-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	v100ebg352_32_33.ucf 无引导文件
V100E-BG352-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	v100ebg352_64_33.ucf 无引导文件
V200E-FG256-6C	66 MHz 3.3V 32 位	pcim_lc_66_3_d	v200efg256_32_66.ucf v200efg256_32_66.ncd
V300E-BG432-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	v300ebg432_32_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
V300E-BG432-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	v300ebg432_64_33.ucf 无引导文件
V300E-BG432-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	v300ebg432_64_66.ucf v300ebg432_64_66.ncd
V400E-FG676-6C	66 MHz 3.3V 32 位	pcim_lc_66_3_d	v400efg676_32_66.ucf v400efg676_32_66.ncd
V1000E-FG680-6C	33 MHz 3.3V 32 位	pcim_lc_33_3_s	v1000efg680_32_33.ucf 无引导文件
V1000E-FG680-6C	33 MHz 3.3V 64 位	pcim_lc_33_3_s	v1000efg680_64_33.ucf 无引导文件
V1000E-FG680-6C	66 MHz 3.3V 64 位	pcim_lc_66_3_d	v1000efg680_64_66.ucf v1000efg680_64_66.ncd
Virtex-II			
2V1000-FG456-4C/I/M	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2v1000fg456_32_33.ucf 无引导文件
2V1000-FG456-4C/I/M	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2v1000fg456_64_33.ucf 无引导文件
2V1000-FG456-5C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2v1000fg456_64_66.ucf 2v1000fg456_64_66.ncd
Virtex-II Pro			
2VP7-FF672-5C/I	33 MHz 3.3V 32 位	pcim_lc_33_3_s	2vp7ff672_32_33.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
2VP7-FF672-5C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2vp7ff672_64_33.ucf 无引导文件
2VP7-FF672-6C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2vp7ff672_64_66.ucf 2vp7ff672_64_66.ncd
2VP20-FF1152-5C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2vp20ff1152_64_33.ucf 无引导文件
2VP20-FF1152-6C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2vp20ff1152_64_66.ucf 2vp20ff1152_64_66.ncd
2VP30-FF1152-5C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2vp30ff1152_64_33.ucf 无引导文件
2VP30-FF1152-6C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2vp30ff1152_64_66.ucf 2vp30ff1152_64_66.ncd
2VP40-FF1152-5C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2vp40ff1152_64_33.ucf 无引导文件
2VP40-FF1152-6C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2vp40ff1152_64_66.ucf 2vp40ff1152_64_66.ncd
2VP50-FF1152-5C/I	33 MHz 3.3V 64 位	pcim_lc_33_3_s	2vp50ff1152_64_33.ucf 无引导文件
2VP50-FF1152-6C/I	66 MHz 3.3V 64 位	pcim_lc_66_3_s	2vp50ff1152_64_66.ucf 2vp50ff1152_64_66.ncd
Virtex-4			
4VFX20-FF672-10C/I 全局时钟	33 MHz 3.3V 32 位	pcim_lc_33_3_g	4vfx20ff672_32_33g.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
4VFX20-FF672-10C/I 全局时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_g	4vfx20ff672_64_33g.ucf 无引导文件
4VFX20-FF672-10C/I 区域时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_r	4vfx20ff672_64_33r.ucf 无引导文件
4VFX20-FF672-10C/I 区域时钟	33MHz 3.3V 32 位	pcim_lc_33_3_r	4vfx20ff672_32_33r.ucf 无引导文件
4VFX20-FF672-11C/I 区域时钟	66 MHz 3.3V 32 位	pcim_lc_66_3r	4vfx20ff672_32_66r.ucf 无引导文件
4VFX20-FF672-11C/I 区域时钟	66 MHz 3.3V 64 位	pcim_lc_66_3_r	4vfx20ff672_64_66r.ucf 无引导文件
4VLX25-FF668-10C/I 全局时钟	33 MHz 3.3V 32 位	pcim_lc_33_3_g	4vlx25ff668_32_33g.ucf 无引导文件
4VLX25-FF668-10C/I 区域时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_r	4vlx25ff668_64_33r.ucf 无引导文件
4VLX25-FF668-10C/I 区域时钟	33MHz 3.3V 32 位	pcim_lc_33_3_r	4vlx25ff668_32_33r.ucf 无引导文件
4VLX25-FF668-11C/I 区域时钟	66 MHz 3.3V 32 位	pcim_lc_66_3r	4vlx25ff668_32_66r.ucf 无引导文件
4VLX25-FF668-11C/I 区域时钟	66 MHz 3.3V 64 位	pcim_lc_66_3_r	4vlx25ff668_64_66r.ucf 无引导文件
4VLX25-FF668-10C/I 全局时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_g	4vlx25ff668_64_33g.ucf 无引导文件

表 3-1: 器件和接口选择表 (续表)

支持的器件	总线类型	封装文件	约束文件 / 引导文件
4VSX35-FF668-10C/I 全局时钟	33 MHz 3.3V 32 位	pcim_lc_33_3_g	4vsx35ff668_32_33g.ucf 无引导文件
4VSX35-FF668-10C/I 区域时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_r	4vsx35ff668_64_33r.ucf 无引导文件
4VSX35-FF668-10C/I 区域时钟	33MHz 3.3V 32 位	pcim_lc_33_3_r	4vsx35ff668_32_33r.ucf 无引导文件
4VSX35-FF668-11C/I 区域时钟	66 MHz 3.3V 32 位	pcim_lc_66_3r	4vsx35ff668_32_66r.ucf 无引导文件
4VSX35-FF668-11C/I 区域时钟	66 MHz 3.3V 64 位	pcim_lc_66_3_r	4vsx35ff668_64_66r.ucf 无引导文件
4VSX35-FF668-10C/I 全局时钟	33 MHz 3.3V 64 位	pcim_lc_33_3_g	4vsx35ff668_64_33g.ucf 无引导文件

a. XC3S1200E 需要对已布线的 NCD 文件进行后处理，以满足 66 MHz 核的保持时间要求。有关更多信息，请见第 6 章“实现设计”。

有关完整的目录结构和文件列表，请见核附带的产品版本说明。

封装文件

封装文件包含 Initiator/Target core for PCI 的实例和该核所用所有 I/O 元件的实例。每个封装文件针对一种特定的 PCI 总线信令环境。

< 安装路径 >/hdl/src/wrap 目录中的封装文件实际上是 < 安装路径 >/hdl/src/xpci 目录中的 pcim_lc.hdl 文件的变体。

开始新设计时，将相应的封装文件从 wrap/ 目录复制到 xpci/ 目录，然后将其重命名为 pcim_lc.hdl。

约束文件

用户约束文件 (UCF) 包含 Initiator/Target core for PCI 所需的各种约束，在处理设计时必须始终使用这些文件。每个约束文件针对特定的器件与核；在用 Xilinx 实现工具处理设计时，请使用 < 安装路径 >/hdl/src/ucf 目录中的相应约束文件。

引导文件和定向布线

引导文件包含高性能 PCI 接口版本所需的布线信息。引导文件位于 < 安装路径 >/hdl/src/guide 目录中。必须以准确引导模式使用引导文件。

使用引导文件的替代方法是使用定向布线约束。这样做可以免除对引导文件的需要，同时保留核的功能和性能。强烈建议使用 < 安装路径 >/hdl/src/dirt 目录中的含有定向布线约束的新 UCF。

其方法是将实现脚本修改成让 **ngdbuild** 使用 < 安装路径 >/hdl/src/dirt 中而非 < 安装路径 >/hdl/src/ucf 中的新 UCF (UCF 的名称保持不变)，并且删除 PAR 的 -gm 和 -gf 选项。

在标明处，必须使用引导文件或定向布线，以确保满足接口性能要求。在任一种情况下，验证布线是否成功都很重要。对于引导文件，表 3-2 指定了每个引导文件中包括的引导元件和引导连接数目。请在实现之后查阅此表，以对照引导报告文件 (GRF) 中的引导摘要验证结果。对于定向布线约束，可以通过检查 PAR 报告文件 (PAR) 来确定布线约束是否成功。

示例设计需要 xpci/ 目录中存在默认的 pcim_lc.hdl 封装文件。如果变更此文件，也必须变更处理脚本中使用的约束和引导文件。

表 3-2: 引导文件信息

引导文件	元件数	连接数
2s150fg456_64_66.ncd	150	134
2s200fg456_64_66.ncd	150	134
2s300efg456_64_66.ncd	214	134
v300bg432_64_66.ncd	214	134
v300ebg432_64_66.ncd	214	134
v1000fg680_64_66.ncd	214	134
v1000efg680_64_66.ncd	214	134
2v1000fg456_64_66.ncd	150	240
2vp7ff672_64_66.ncd	153	246
2vp20ff1152_64_66.ncd	153	246
2vp30ff1152_64_66.ncd	153	246
2vp40ff1152_64_66.ncd	153	246
2vp50ff1152_64_66.ncd	153	246
v200fg256_32_66.ncd	90	86
v200efg256_32_66.ncd	90	86
v400fg676_32_66.ncd	90	86

器件初始化

紧接 FPGA 配置之后，存在于所有 Virtex 和 Spartan 器件中的启动机制对 Initiator/Target core for PCI 和用户应用进行初始化。在正常操作期间，将 PCI 总线上的 RST# 置为有效会重新初始化核并且三态化所有 PCI 总线信号。此行为完全符合《PCI 局部总线技术规范》。核设计为能够正确处理异步复位。

通常，每当初始化 Initiator/Target core for PCI 接口时都要初始化用户应用。在此情况下，请将核的 RST 输出用作用户应用的异步复位信号。如果部分用户应用需要与 PCI 总线复位异步的初始化功能，只要将用户应用设计成使用独立的复位信号即可。

请注意，因为不使用全局资源，这类复位方案需要使用布线资源来分配复位信号。不推荐使用全局复位资源。

配置引脚

设计人员应注意，不要将 PCI 总线接口引脚放置在用于配置的两用 I/O 引脚上。请验证所选 UCF，以免这些引脚与用于所选配置模式的引脚冲突。也不妨将 PCI 引脚放置在两用 I/O 配置引脚上，只要后者并非还用于配置即可。有关配置引脚的位置，请查阅相应的器件管脚指南。

总线宽度检测

提供 64 位数据通路的 Initiator/Target core for PCI 必须知道所连接的是 64 位总线还是 32 位总线。为此，向 PCI64 接口输入一个 SLOT64 信号。PCI 总线规范提供了一种机制，让 PCI Agent 在 RST# 的上升沿对 REQ64# 信号的状态进行采样，从而确定总线宽度。

在嵌入式系统中，设计已知总线宽度，用户应用只要用相应值驱动 SLOT64 即可。请注意，切勿用静态值驱动 SLOT64，应始终由触发器的输出对其进行驱动。

在开放系统的设计中，预先不知道总线宽度。对于这种情况，请在 FPGA 外部增加一个独立的锁存器或触发器对 REQ64# 进行采样。图 3-1 所示为这种配置方法。

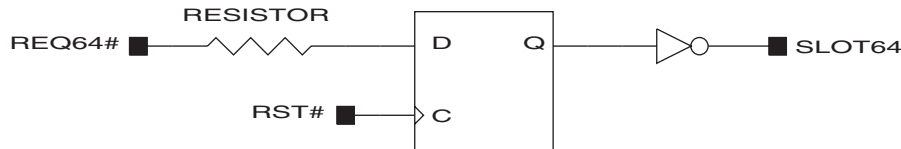


图 3-1: SLO64 样本信号的生成

虽然这种方法从技术上讲不符合 PCI 规范（由于 REQ64# 和 RST# 上的额外负载），但使用一个大串联电阻有助于尽量减少这种影响。可以将倒相器置入 FPGA。

一种变通方法是将整个电路置入 FPGA，并且使用为用户应用提供的 REQ64Q_N 和 RST 信号。这种方法需要 FPGA 完全由 RST# 的上升沿配置。当 SLO64 置为无效时，64 位 Initiator/Target core for PCI 自动三态化 64 位扩展信号。在这种情况下，不驱动 64 位扩展信号，这样会致使输入缓冲器增加功耗。

如果功耗的增加由于设计要求而举足轻重，请考虑改变 HDL 配置文件中的“禁用扩展驱动”选项。当 SLO64 置为无效时，此选项启用后强制 64 位 Initiator/Target core for PCI 主动驱动扩展信号。

注：此选项虽然可以降低功耗，但会造成电气上的不良影响。当在 32 位插槽中安装 64 位板卡时，64 位总线扩展部分悬空，从而容易被晃动的螺丝起子损坏。

数据通路输出时钟使能

Initiator/Target core for PCI 使用以下方法之一生成并分配数据通路输出时钟使能信号：

- 专用器件资源：PCIIOB、PCILOGIC 和 PCI_CE
- 通用器件资源：IOB、LUT 和通用布线

专用器件资源提供较高且较容易预见的性能。但并非所有器件系列中都有这种资源。当使用专用器件资源时，核被约束在 FPGA 器件的中左或中右侧，而 Initiator/Target core for PCI 接口的实例数限为两个。通用器件资源性能较低且可预见性较差，但可以提供较大的灵活性。

表 3-3: 输出时钟使能设置

实现	输出时钟使能设置
Virtex、Virtex-E（全部）	“0”（专用）
Spartan-II、Spartan IIE（全部）	“0”（专用）
Virtex-II、Virtex-II Pro（全部）	“1”（通用）
Spartan-3、Spartan-3A、Spartan-3E (33 MHz)	“1”（通用）
Spartan-3、Spartan-3A、Spartan-3E (66 MHz)	“0”（专用）
Virtex-4（全部）	“1”（通用）

表 3-3 所示为正确的输出时钟使能配置。要使用通用器件资源，请将 HDL 配置文件中的 CFG[251] 数据位设置为逻辑一。要使用专用器件资源，请将 HDL 配置文件中的 CFG[251] 数据位设置为逻辑零。

输入延迟缓冲器

输入延迟缓冲器用来保证所有总线输入上的保持时间。在可能处，以 Virtex 器件为目标的 Initiator/Target core for PCI 使用 FPGA 器件的 IOB 中的输入延迟单元。这些延迟缓冲器的使用是通过针对实现的约束文件来选择。

有些实现用 HDL 配置文件中的 CFG[248:245] 数据位来选择使用另一些延迟缓冲器。

表 3-4 所示为 HDL 配置文件中 CFG[248:245] 数据位的所需设置。

表 3-4: 延迟缓冲器设置

实现	延迟设置
Virtex、Virtex-E（全部）	“0000”
Spartan-II、Spartan IIE（全部）	“0000”
Virtex-II、Spartan-3、Spartan-3E（全部）	“0000”
Virtex-II Pro（全部 33 MHz）	“0000”
Virtex-II Pro（全部 66 MHz）	“0000”
Virtex-4（全局时钟）	“0000”
Virtex-4（区域时钟）66 MHz	“0000”
Virtex-4（区域时钟）33 MHz	“0000”

您必须针对所选实现相应地设置 CFG[248:245]。尽管默认设置适合大多数实现，但设置与所选实现失配可能导致硬件故障。

Virtex-4 实现使用 IDELAY 输入延迟缓冲器基元。IDELAY 输入延迟缓冲器是标定延迟时间的可调延迟线。这种延迟机制可提供优于传统输入延迟缓冲器的性能。请务必遵守表 3-4 所示设置，并且使用针对设计的相应 UCF。

使用 IDELAY 基元的设计还需要使用 IDELAYCTRL 基元。IDELAYCTRL 基元的功能是标定 IDELAY 延迟线的延迟时间。要完成此标定，IDELAYCTRL 基元需要 200 MHz 的输入时钟。使用参考时钟的设计和封装文件包含 IDELAY 实例、IDELAYCTRL 实例和来自 I/O 引脚的 200 MHz 参考时钟的附加输入 RCLK。此参考时钟用一个全局时钟缓冲器分配到所有适用的 IDELAYCTRL 基元。

此 200 MHz 参考时钟的来源、生成和使用具有一定灵活性。所提供设计和封装文件代表的是一个泛例，可以进行如下修改以适合具体的设计要求：

- 对于需要将 IDELAY 和 IDELAYCTRL 用于其他 IP 核或定制用户逻辑的设计，可以共用 200 MHz 参考时钟。封装文件中的参考时钟经全局缓冲器驱动之后可以引出。此信号可供其他 IDELAY 和 IDELAYCTRL 实例使用。
- 对于已经在全局时钟缓冲器上分配 200 MHz 参考时钟的设计，可以共用此时钟。可以修改封装文件，删除外部 I/O 引脚和全局时钟缓冲器实例。只要引出现有的 200 MHz 时钟信号并将其加入封装文件供接口使用即可。
- 对于没有 200 MHz 参考时钟的设计，可以用数字时钟管理器 (DCM) 和另一时钟生成 200 MHz 参考时钟。另一时钟可以从内部或外部获得，但必须具有固定频率。在此情况下，必须进行以下验证：
 - ◆ 验证源时钟的抖动，以确定其是否适合用作 DCM 的输入。
 - ◆ 验证 DCM 配置，以便在任意相应 DCM 输出（CLKFX、CLKDV 等）上生成 200 MHz 时钟。
 - ◆ 验证所生成 200 MHz 参考时钟的抖动，以确定其是否适合用作 IDELAYCTRL 的输入。

- ◆ IDELAYCTRL 复位必须与 DCM 锁定输出联系起来，以使 IDELAYCTRL 在 DCM 锁定之前一直保持复位状态。

有关相关时序参数的更多信息，请见《[Virtex-4 技术文档](#)》。像其他实现选项一样，生成的 200 MHz 参考时钟必须由全局时钟缓冲器分配到 IDELAYCTRL 实例。

重要提示！ 源时钟的固定频率要求不允许使用 PCI 总线时钟，除非设计是用于嵌入 / 封闭式系统，这种系统中的 PCI 总线时钟具有已知的固定频率。有关 PCI 总线时钟在兼容的系统中的允许行为的进一步信息，请见“[总线时钟用法](#)”。

区域时钟用法

有些 Virtex-4 实现将区域时钟缓冲器 (BUFR) 用于 PCI 总线时钟，而不使用全局时钟缓冲器 (BUFG)。使用区域时钟资源可大大改善接口的引脚间时钟到输出延迟，同时保持完全的兼容性。（对于 PCI，引脚间时钟到输出延迟是重要的硅片（芯片）性能参数。）

设计人员务必了解因使用区域时钟造成的附加约束。Virtex-4 器件划分为若干时钟区域。区域时钟信号进入给定区域的中心，进而从入口区域纵跨上下两个区域。区域时钟达到的范围在物理上限制为三个时钟区域。[图 3-2](#) 所示为驱动三个时钟区域的 BUFR。有关区域时钟的更多信息，请见《[Virtex-4 数据手册](#)》和《[Virtex-4 用户指南](#)》。

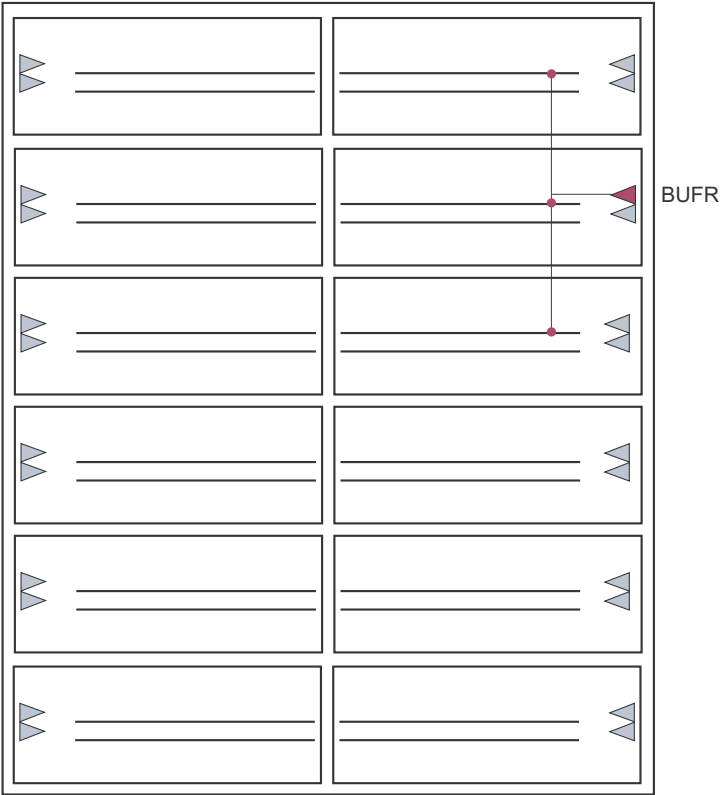


图 3-2: 区域时钟控制图解

对于使用区域时钟控制的设计，必须将核以及由 PCI 总线时钟进行时钟控制的用户应用部分完全置于区域时钟信号可以进入的三个时钟区域内。这一约束限制了可与 PCI 总线时钟同步的 FPGA 资源数。可以通过跨接另一时钟域来访问其他逻辑。

时钟区域的高度是 16 个 CLB/32 个 IOB，宽度是器件宽度的二分之一。因为区域时钟的跨度限制为三个时钟区域，所以一个区域时钟最多提供 96 个可用于 PCI 接口的 IOB。64 位 Initiator/Target core for PCI 需要 90 个 IOB，而 32 位 Initiator/Target core for PCI 需要 50 个 IOB。在某些器件与封装的组合中（通常是物理尺寸较大的器件装入引脚相对较少的封装），并非所有 IOB 区都与封装引脚连接。这样就造成某些时钟区域无法使用。这对于 32 位核接口一般不成问题，但对于 64 位核接口则关系重大。

表 3-5 详细列出了从物理上讲可能装入各种使用区域时钟的 Virtex-4 器件与封装组合的所有 64 位核接口。请注意，此表不适合使用全局时钟的 Virtex-4 设计。

表 3-5: Virtex-4 器件与封装的 64 位接口

封装	器件	64 位接口数
SF363	LX15	无
	LX25	无
	FX12	无

表 3-5: Virtex-4 器件与封装的 64 位接口 (续表)

封装	器件	64 位接口数
FF668	LX15	2
	LX25	4
	LX40	2
	LX60	2
	FX12	2
	SX25	2
	SX35	4
FF1148	LX40	4
	LX60	4
	LX80	6
	LX100	6
	LX160	6
	SX55	4
FF1513	LX100	8
	LX160	8
	LX200	8
FF672	FX20	2
	FX40	无
	FX60	无
FF1152	FX40	4
	FX60	4
	FX100	4
FF1517	FX100	6
	FX140	6
FF1760	FX140	8

总线时钟用法

核接口提供的总线时钟输出源于总线时钟输入，使用全局时钟缓冲器或区域 (Virtex-4 66 MHz) 时钟缓冲器分配。接口本身与此时钟完全同步。一般来说，与接口通讯的用户应用部分也必须与此时钟同步。

此时钟的频率不能保证一成不变。在兼容的系统中，该时钟实际上可以是允许范围内的任何频率（包括最高允许频率），而且可能在周期与周期之间发生变化。在某些条件下，核还可能对此时钟应用相移。

因此，用户应用不应将此时钟用作 DLL 或 PLL 的输入，也不应将其用于计时器（如 DRAM 刷新计数器）的设计。

电气兼容性

根据信令环境不同，以 Virtex 器件为目标的 Initiator/Target core for PCI 使用三种 Virtex I/O 缓冲器类型之一（用封装文件进行此选择）。

注：Virtex-E、Virtex-II、Virtex-II Pro、Virtex-4、Spartan-II E、Spartan-3 和 Spartan-3E 器件不能承受 5.0V 电压。请勿在 5.0V 信令环境中使用这些器件。

5.0V 信令环境的封装文件使用 Virtex 和 Spartan-II 器件上提供的 PCI33_5 I/O 缓冲器。这就需要将 V_{CCO} 设置为 3.3V，并且不需要 V_{REF} 供电。请遵守器件产品技术规范中所述相关规定。不适用其他约束条件。

3.3V 信令环境的封装文件使用 Virtex、Virtex-4、Virtex-E、Virtex-II、Virtex-II Pro、Spartan-II、Spartan-II E、Spartan-3 和 Spartan-3E 器件上提供的 PCI33_3 或 PCI66_3 I/O 缓冲器。除 Virtex-II Pro、Virtex-4、Spartan-3 和 Spartan-3E 以外，这些器件需要将 V_{CCO} 设置为 3.3V，并且不需要 V_{REF} 供电。请遵守器件数据手册中的相关规定。

对于 Virtex-E、Virtex-II、Virtex-II Pro、Virtex-4、Spartan-II E、Spartan-3 和 Spartan-3E 器件中的 3.3V 信令，不适用其他约束条件。不过，对于 Virtex 和 Spartan-II 器件中的 3.3V 信令，仍要应用附加约束。对于 Virtex 和 Spartan-II 器件中的 3.3V 信令，数据手册指明输入缓冲器的 V_{IL} 和 V_{IH} 参数是 V_{CCINT} 的函数，而 V_{CCINT} 是 2.5V 供电。在《PCI 局部总线技术规范》中，有关 3.3V 信令环境的规定将 V_{IL} 和 V_{IH} 叙述为 V_{CC} 的函数。可以将此看成是 3.3V 系统供电。

当 2.5V 和 3.3V 供电处于相反的极值时，就会违反 3.3V V_{IL} 或 V_{IH} 的规定。这只是技术上的违反，不会影响功能。 V_{IL} 或 V_{IH} 不会有超越《PCI 局部总线技术规范》中所述参数以致显著影响噪声容限的危险。对于所有供电组合， V_{IL} 会始终在规定的 35 mV 限度内，而 V_{IH} 会始终在规定的 75 mV 限度内。它们不会同时超出规定值。

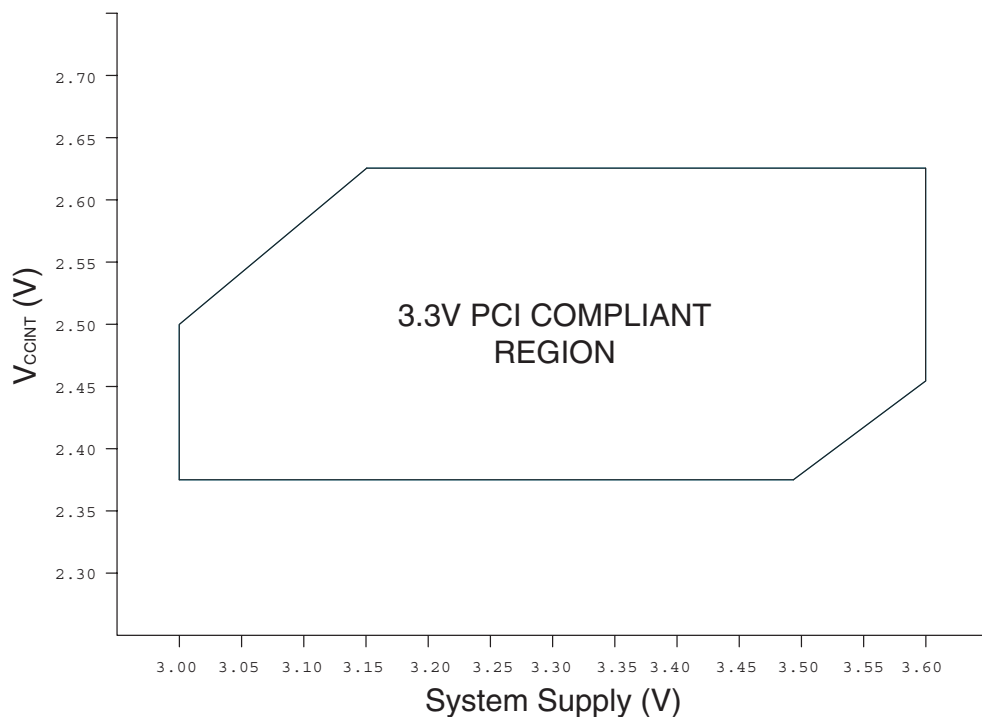


图 3-3: 3.3V 输入缓冲器的兼容性关系

图 3-3 所示为供电电压值的狭小范围，其中的 V_{IL} 或 V_{IH} 在技术上不符合。请注意，如果输入缓冲器的供电电压与驱动器件的供电电压不同，任何 PCI 器件都可能发生这种情况。为了获得最佳结果，请使用高精度电压调节器生成 V_{CCINT} 。

对于 Virtex-5、Virtex-4 和 Virtex-II Pro 中的 3.3V 信号， V_{CCO} 供电必须降至 3.0V，而且必须由高精度调节器生成。如此降低输出驱动器件的供电电压可以为器件提供有力的保护，同时无损于 PCI 的电气兼容性，甚至当 3.3V 系统供电攀升到《PCI 局部总线技术规范》允许的 3.6V 极限值时也无妨。

对于 Spartan-3 和 Spartan-3E 器件，《Spartan-3 数据手册》和《Spartan-3E 数据手册》列出的 V_{CCO} 要求是 3.3V-10%/+5%（即 3.0V 到 3.45V）。对于使用 3.0V 稳压 V_{CCO} 的 Spartan-3 和 Spartan-3E 现有设计，此值仍然是符合的，因此不需要任何改变。对于针对 PCI 的 Spartan-3 和 Spartan-3E 新设计，仍有必要调节 V_{CCO} ，因为 PCI 技术规范规定 PCI 的电源供电线为 3.3V+/-10%（即 3.0V 到 3.6V）。因此，Spartan-3 和 Spartan-3E 需要调压器。不过，现在可以将电压调到 3.3V 而非 3.0V。有关 Spartan-3 系列器件的 V_{CCO} 供电的更多信息，请见 [XAPP457](#)。

Spartan-3A 不需要调节 V_{CCO} ，因为 Spartan-3A 器件的 PCI 缓冲器的 V_{CCO} 要求是 3.3V+/-10%。有关器件电气参数的更多信息，请见《Spartan-3A 数据手册》。可以将 V_{CCO} 直接连接到 PCI 的 3.3V 电源供电线。

图 3-4 所示为一种可能的低成本解决方案，用于为 Virtex-5、Virtex-4 和 Virtex-II Pro 器件生成所需的 3.0V 输出驱动器电源。图 3-5 所示为用同样的调节器为 Spartan-3 和 Spartan-3E 器件生成 3.3V 输出驱动器电源。这两幅图显示用于建立输出电源电压的电阻

分压器网络使用了不同的电阻值。虽然也可以采纳使用其他调节器的其他方法，但 Xilinx 建议使用图 3-4 和图 3-5 所示电路。

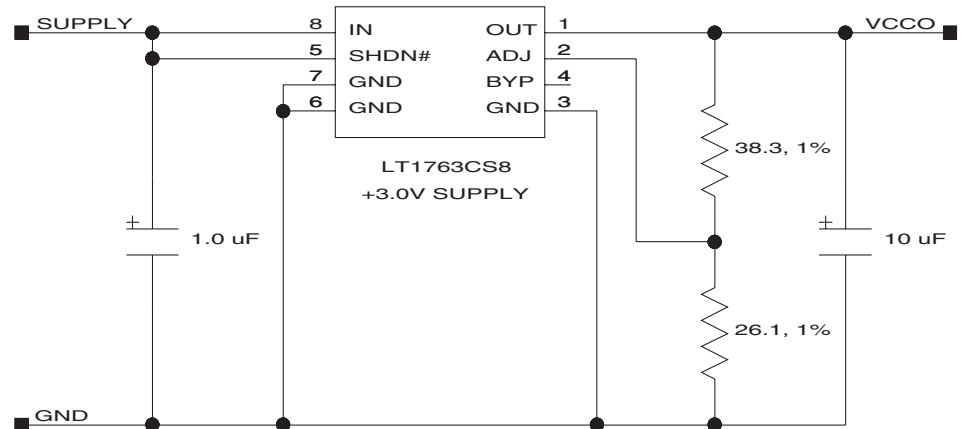


图 3-4: 3.0V 输出驱动器的 V_{CC0} 生成

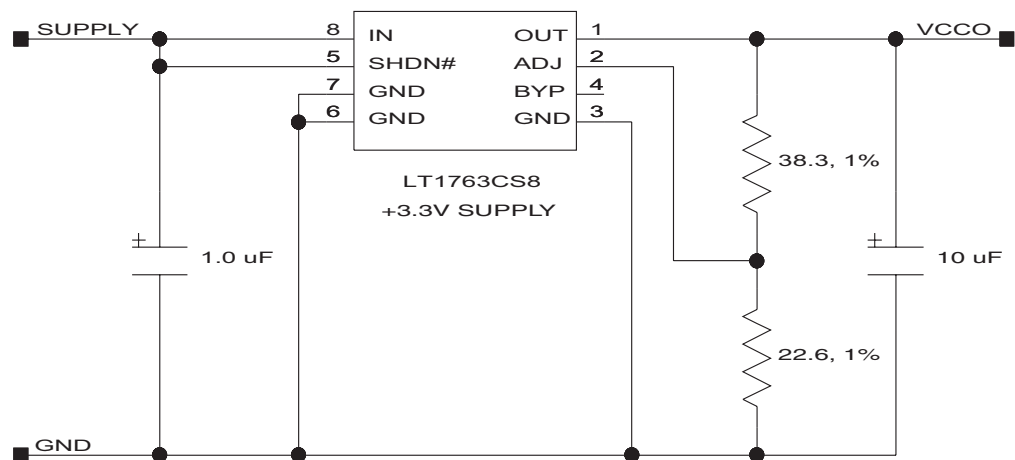


图 3-5: Spartan-3 和 Spartan-3E 的 3.3V 输出驱动器的 V_{CC0} 生成

Virtex-II、Virtex-II Pro、Virtex-4、Spartan-3 和 Spartan-3E 器件具有 10pF 引脚电容。这符合《PCI 局部总线技术规范》；只有一个例外，即规范要求 IDSEL 引脚的引脚电容为 8pF，以便实现对 AD[xx] 引脚的非阻性耦合。在实际中，此耦合可以是阻性或非阻性的，并且是在系统电路板或背板上实现。对于系统电路板或背板设计，请使用阻性耦合以避免不兼容性。对于插入式板卡，这不在设计人员的控制范围内。

虽然 Initiator/Target core for PCI 不直接为电源管理事件报告提供 PME# 信号，但可以由用户应用实现此功能。典型的实现将涉及在用户配置空间内实现电源管理功能项，同时在通用 I/O 引脚上实现专用的 PME# 输出。

对于所有器件系列，如果去除 FPGA 电源，则通用 I/O 引脚对地呈现为低阻抗，而对系统呈现为 PME# 的有效置位。因此，使用 PME# 信号的实现应采用外部缓冲方案，以免从 FPGA 器件中去除电源后出现 PME# 的假有效置位。

生成比特流

在生成用于 PCI 设计的比特流时，比特流生成程序 **bitgen** 可能发出 DRC 警告。这些警告的条数因用于核的配置选项不同而不同。通常，这些警告与在映射程序在修正期间生成没有负载的网状连线有关。这类连线有些是 UCF 中的语句有意保留的。

请注意，示例设计附带的 **bitgen** 选项仅供参考。实际使用的选项取决于在电路板上实现完整设计所需的 FPGA 配置方法和时钟速率。在选择配置方法和时钟速率时，请慎重考虑以下配置时间要求。

- 任何不自动检测总线宽度的设计都必须在总线电源供电线变为有效之后的 100 ms + 2^{25} 个总线时钟期间内配置。
- 任何必须检测总线宽度的设计都必须在总线电源供电线变为有效之后的 100 ms 内配置。
- Cardbus 设计必须在总线电源供电线变为有效之后尽快配置。

对于某些较老的 66 MHz 设计，运行 **bitgen** 时必须带有以下特别选项，以改变设计中所用全局时钟缓冲器的行为：

```
bitgen -g Gclkdel<buf>:<opt> pcim_top_routed.ncd
```

此选项用来在全局时钟网上引入附加延迟。值得注意的是，这种附加延迟对核接口的 CLK 输出影响显著，而该输出是提供给用户应用的。为用户应用生成时序约束时必须考虑到这一点。有关这一必备选项的使用和含义的进一步信息，请见产品版本说明。

功能仿真

本章讲述如何用所支持的功能仿真工具对使用全局时钟的 *ping64* 示例设计进行仿真。对于 32 位 Initiator/Target core for PCI，请用 *ping32* 替换 *ping64*。如果您用的是使用参考时钟的设计，请用 *ping_tb_r* 替换 *ping_tb*，并用 *pcim_top_r* 替换 *ping_tb*。

支持的功能仿真工具有：

- Cadence IUS
- Mentor Graphics ModelSim

Cadence IUS

在尝试功能仿真之前，请确保已正确配置 IUS 环境。

1. 首先，请转到功能仿真目录：

```
cd <安装路径>/verilog/example/func_sim
```

2. 修改 *ping_tb.f* 文件。此文件列出 IUS 的命令行参数，如下所示：

```
../source/ping_tb.v
../source/stimulus.v
../source/busrecord.v
../source/dumb_arbiter.v
../source/dumb_targ32.v
../source/dumb_targ64.v
../source/pcim_top.v
../source/ping.v
../source/cfg_ping.v
../source/global.v
../../src/xpci/pci_lc_i.v
../../src/xpci/pcim_lc.v
+libext+.vmd+.v
-y <Xilinx 安装路径>/verilog/src/unisims
-y <Xilinx 安装路径>/verilog/src/simprims
```

3. 修改库搜索路径，即将 <Xilinx 安装路径> 改成与 Xilinx 安装目录匹配，然后保存文件。

列出的大多数文件都与示例设计及其测试平台有关。对于其他测试平台，必须使用以下子集才能对核接口进行正确仿真：

```
../source/glbl.v
../../src/xpci/pci_lc_i.v
../../src/xpci/pcim_lc.v
+libext+.vmd+.v
-y <Xilinx 安装路径>/verilog/src/unisims
-y <Xilinx 安装路径>/verilog/src/simprims
```

此列表不包括任何配置文件、用户应用、顶层封装或测试平台。进行有意义的仿真需要这些附加文件。

4. 要运行 IUS 仿真，请输入以下命令：

```
ncverilog -f ping_tb.f
```

IUS 处理仿真文件，然后退出。测试平台将状态信息打印到控制台。仿真完成后，请查看 ncverilog.log 文件，检查是否有错误。

可以用 Simvision 浏览器查看仿真结果。

5. 如果需要，请用以下命令启动 Simvision：

```
simvision
```

Mentor Graphics ModelSim

在尝试功能仿真之前，请确保已正确配置 ModelSim 环境。

Verilog

1. 请转到功能仿真目录：

```
cd <安装路径>/verilog/example/func_sim
```

2. 修改 ping_tb.f 文件。此文件列出命令行参数，如下所示：

```
../source/ping_tb.v
../source/stimulus.v
../source/busrecord.v
../source/dumb_arbiter.v
../source/dumb_targ32.v
../source/dumb_targ64.v
../source/pcim_top.v
../source/ping.v
../source/cfg_ping.v
../source/glbl.v
../../src/xpci/pci_lc_i.v
../../src/xpci/pcim_lc.v
+libext+.vmd+.v
-y <Xilinx 安装路径>/verilog/src/unisims
-y <Xilinx 安装路径>/verilog/src/simprims
```

3. 修改库搜索路径，即将 <Xilinx 安装路径> 改成与 Xilinx 安装目录匹配，然后保存文件。

列出的大多数文件都与示例设计及其测试平台有关。对于其他测试平台，必须使用以下子集才能对 PCI 接口进行正确仿真：

```
../source/glbl.v
../../src/xpci/pci_lc_i.v
../../src/xpci/pcim_lc.v
+libext+.vmd+.v
-y <Xilinx 安装路径>/verilog/src/unisims
-y <Xilinx 安装路径>/verilog/src/simprims
```

此列表不包括任何配置文件、用户应用、顶层封装或测试平台。进行有意义的仿真需要这些附加文件。

4. 启动 ModelSim，确保将当前目录设置为：
<安装路径>/verilog/example/func_sim
5. 要运行仿真，请输入以下命令：
do modelsim.do

此操作编译所有模块，将其加载到仿真器中，显示波形查看器，然后运行仿真。

VHDL

1. 请转到功能仿真目录：
cd <安装路径>/vhdl/example/func_sim
2. 查看 ping.files 文件。此文件列出所需的各个源文件，如下所示：
../../src/xpci/pci_lc_i.vhd
../../src/xpci/pcim_lc.vhd
../source/cfg_ping.vhd
../source/ping.vhd
../source/pcim_top.vhd
../source/busrecord.vhd
../source/dumb_arbiter.vhd
../source/dumb_targ32.vhd
../source/dumb_targ64.vhd
../source/stimulus.vhd
../source/ping_tb.vhd

列出的大多数文件都与示例设计及其测试平台有关。对于其他测试平台，必须使用以下子集才能对 PCI 接口进行正确仿真：

```
../../src/xpci/pci_lc_i.vhd
../../src/xpci/pcim_lc.vhd
```

此列表不包括任何配置文件、用户应用、顶层封装或测试平台。进行有意义的仿真需要这些附加文件。

3. 启动 ModelSim，确保将当前目录设置为以下路径：
< 安装路径 >/vhdl/example/func_sim
4. 创建 SimPrim 和 UniSim 库。此步骤只需在首次仿真时进行一次：

```
vlib simprim  
vcom -93 -work simprim <Xilinx 安装路径  
>/vhdl/src/simprims/simprim_Vpackage_mti.vhd  
vcom -93 -work simprim <Xilinx 安装路径  
>/vhdl/src/simprims/simprim_Vcomponents_mti.vhd  
vcom -93 -work simprim <Xilinx 安装路径  
>/vhdl/src/simprims/simprim_VITAL_mti.vhd  
vlib unisim  
vcom -93 -work unisim <Xilinx 安装路径  
>/vhdl/src/unisims/unisim_VPKG.vhd  
vcom -93 -work unisim <Xilinx 安装路径  
>/vhdl/src/unisims/unisim_VCOMP.vhd  
vcom -93 -work unisim <Xilinx 安装路径  
>/vhdl/src/unisims/unisim_VITAL.vhd
```
5. 要运行仿真，请输入以下命令：
do modelsim.do

此操作编译所有模块，将其加载到仿真器中，显示波形查看器，然后运行仿真。

综合设计

本章讲述如何用支持的综合工具综合使用全局时钟的 *ping64* 示例设计。对于 32 位接口，请用 *ping32* 替换 *ping64*。如果您用的是使用参考时钟的设计，请用 *pcim_top_r* 替换 *pcim_top*，并用 *ping_tb_r* 替换 *ping_tb*。

支持的综合工具有：

- Synplicity Synplify
- Exemplar LeonardoSpectrum
- Xilinx XST

Synplicity Synplify

在尝试综合设计之前，请确保已正确配置 Synplicity Synplify 环境。

Verilog

1. 启动 Synplify，并且选择 File > New（图 5-1），或者单击工具条上的新文件图标。这时会出现 Create a New Document（创建新文档）对话框。

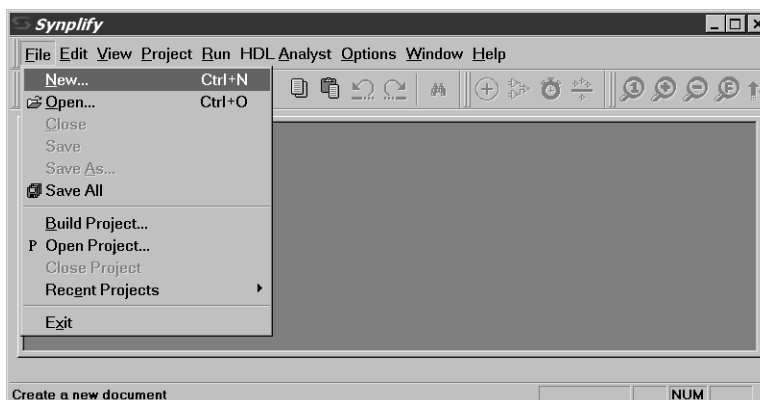


图 5-1：创建新文档

2. 在 File Type 下选择 Project File，然后输入项目名称（在此示例中是 *flowtest*）和综合目录：
< 安装路径 >/verilog/example/synthesis
3. 单击 OK 退出对话框并返回到项目窗口。

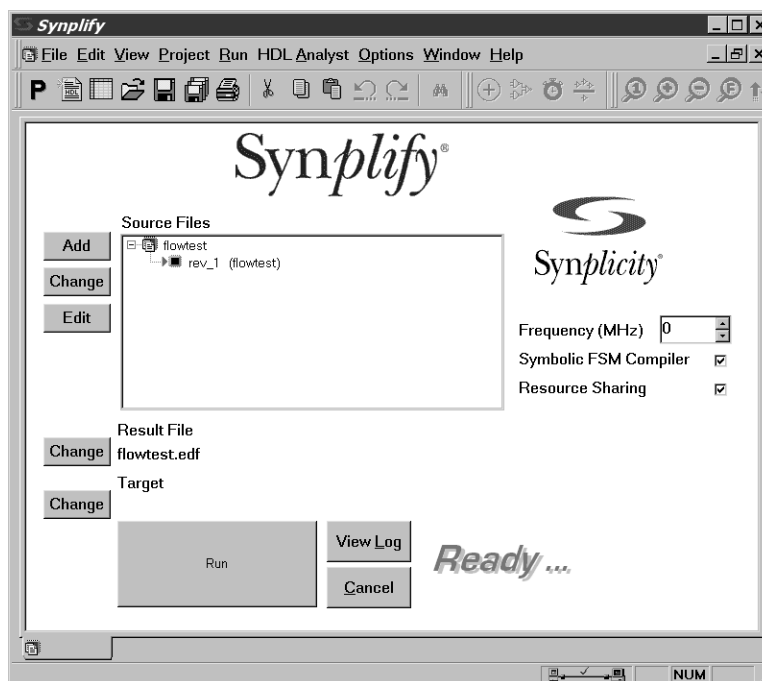


图 5-2: 主项目窗口

4. 单击 Add 将源文件添加到新项目（图 5-2）。第一个文件（由例化 Xilinx 基元的任意设计使用）位于：
<Synplicity 安装路径>/lib/xilinx
5. 转到 virtex.v 文件（图 5-3），然后单击 Add 将此源文件移入 Files To Add（要添加的文件）列表。



图 5-3: 选择要添加的文件（库）

后续文件位于：

< 安装路径 >/verilog/src/xpci

6. 转到 xpci 目录 (图 5-4)，选择仿真模型和封装文件 (pci_lc_i.v 和 pcim_lc.v)，然后单击 Add 将这些文件移入 Files To Add 列表。(用 Ctrl + 单击可选择多个文件。)



图 5-4: 要添加的文件 (LogiCORE IP 文件)

最终的设计文件集 (用户应用) 位于：

< 安装路径 >/verilog/example/source

7. 转到源目录，选择 cfg_ping.v、pcim_top.v 和 ping.v 文件 (图 5-5)，然后单击 Add。



图 5-5: 选择要添加的文件 (用户应用)

8. 添加最后三个文件（共六个源文件）之后，单击 OK 返回到主项目窗口。
9. 在 Source Files 列表中，通过双击 flowtest/verilog 文件夹（如果该文件夹尚未打开）查看新添加源文件的列表。
10. 重新排列文件夹中源文件的顺序，方法是拖动这些文件，将其按图 5-6 所示层级顺序排列。

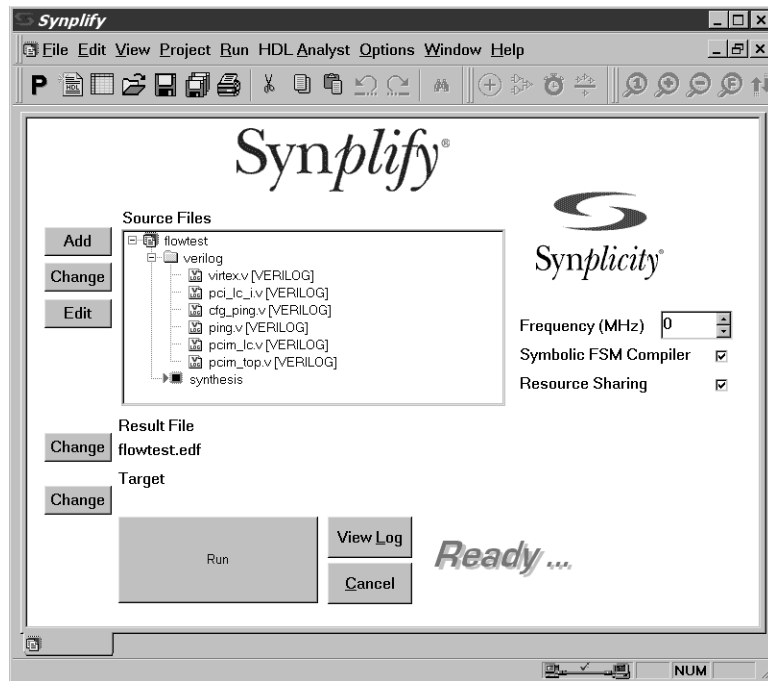


图 5-6：显示源文件的主项目窗口

11. 单击 Change Result File 以显示 EDIF Result File 对话框，然后将其移动到以下目录：
< 安装路径 >/verilog/example/synthesis
12. 为文件 pcim_top.edf 命名，然后单击 OK 设定结果文件的名称并返回到主项目窗口。
注：在实际中，不需要变更 EDIF 结果文件的目录。不过，示例设计附带的样本处理脚本假定输出 EDIF 文件位于 synthesis 目录。
13. 单击 Change Target 以显示 Options for Implementation（实现选项）对话框，如图 5-7 所示。

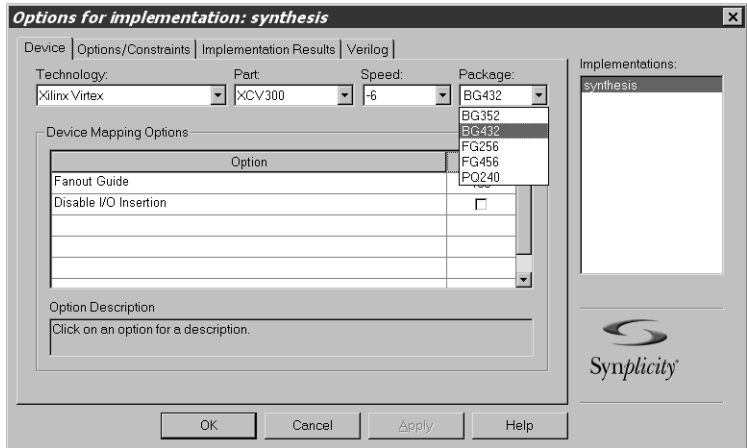


图 5-7: 实现选项: 器件

14. 在 Device 标签上设置 Technology、Part、Speed 和 Package 选项，以反映目标器件（在此示例中是 V300BG432-6）。
- 请务必勾掉 Disable I/O Insertion。
15. 在 Options/Constraints 标签上，勾掉 Symbolic FSM Compiler（但保持 Resource Sharing 为选中状态），并且将 Frequency 设置为 66 MHz。
16. 在 Implementation Results 标签上勾掉 Write Vendor Constraint File。
17. 单击 OK 返回到主项目窗口。
18. 单击 Run。

Synplify 综合设计并写出优化的 EDIF 文件。在窗口的右下角显示综合的各个阶段，如 Compiling 或 Mapping。

过程完成后显示 Done。请注意，预计 Synplify 会发出几条有关用于其他综合工具的例化 I/O 基元和属性的警告。

VHDL

1. 启动 Synplify，并且选择 File > New（图 5-8），或者单击工具条上的新文件图标。这时会出现 Create a New Document 对话框。

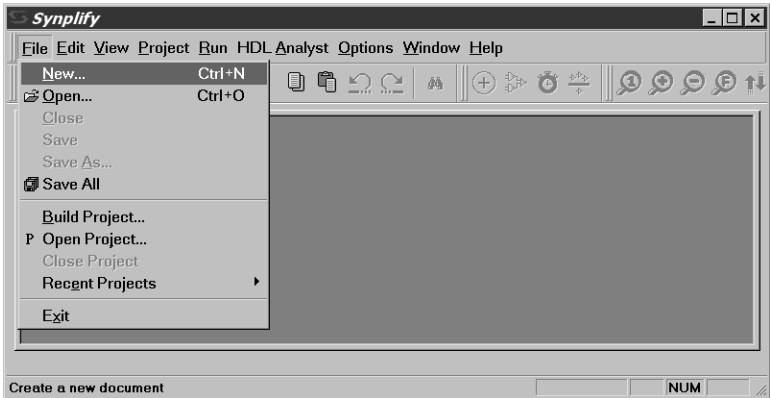


图 5-8: 创建新文档

2. 在 File Type 下选择 Project File，然后输入项目名称（在此示例中是 *flowtest*）和综合目录：
< 安装路径 >/vhdl/example/synthesis
3. 单击 OK 退出对话框并返回到项目窗口（图 5-9）。

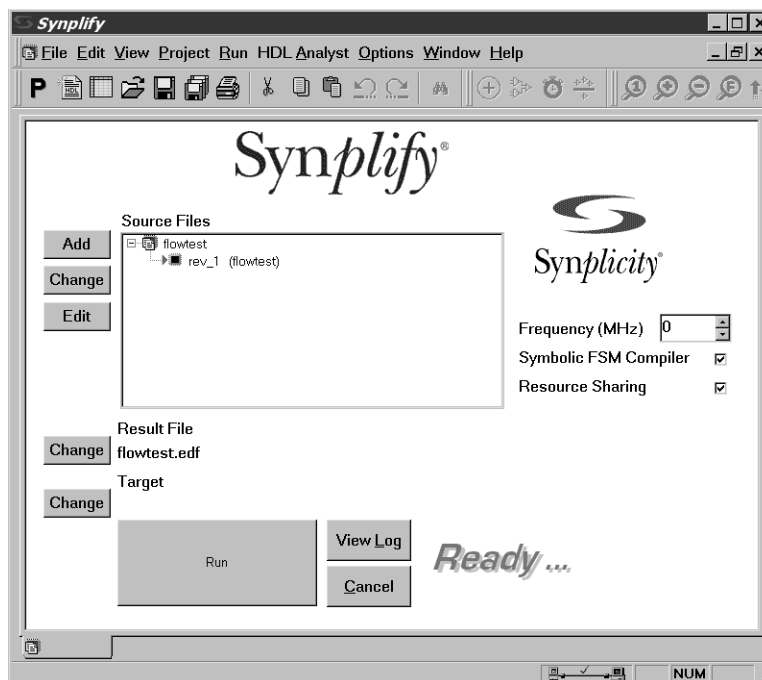


图 5-9：项目窗口

4. 单击 Add 将源文件添加到新项目。
第一个文件（由例化 Xilinx 基元的任意设计使用）位于：
<Synplicity 安装路径 >/lib/xilinx

5. 选择 virtex.v 文件 (图 5-10), 然后单击 Add 将此源文件移入 Files To Add 列表。



图 5-10: 选择要添加的文件 (库)

后续文件位于:

< 安装路径 >/vhdl/src/xpci

6. 转到 xpci 目录, 选择仿真模型和封装文件 (pci_lc_i.vhd 和 pcim_lc.vhd) (图 5-11)。
7. 单击 Add 将这些文件添加到 Files To Add 列表。(用 Ctrl + 单击可选择多个文件。)



图 5-11: 选择要添加的文件 (LogiCORE IP 文件)

最终的设计文件集（用户应用）位于：

< 安装路径 >/vhdl/example/source

8. 转到源目录，选择 cfg_ping.vhd、pcim_top.vhd 和 ping.vhd 文件，然后单击 Add。
9. 添加最后三个文件（共六个源文件）之后，单击 OK 返回到主项目窗口。
10. 在 Source Files 列表中，通过双击 flowtest/vhdl 文件夹（如果该文件夹尚未打开）查看新添加源文件的列表。单击并拖动源文件，将其按图 5-12 所示层级顺序重新排列。

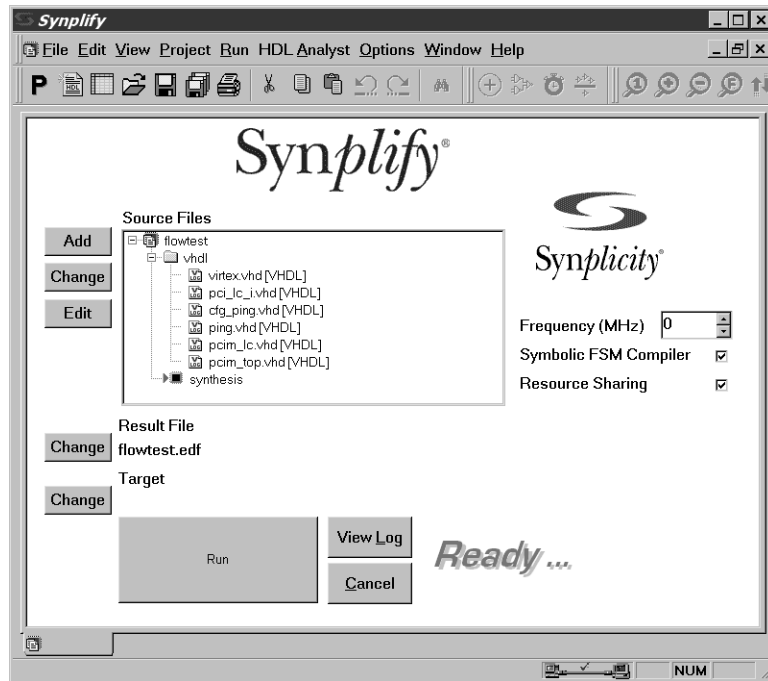


图 5-12：显示源文件的项目窗口

11. 单击 Change Result File 以显示 EDIF Result File 对话框，然后将其移动到以下目录：
< 安装路径 >/vhdl/example/synthesis
12. 为文件 pcim_top.edf 重新命名，然后单击 OK 返回到主项目窗口。

注：在实际中，不需要变更 EDIF 结果文件的目录。不过，示例设计附带的样本处理脚本假定输出 EDIF 文件位于 synthesis 目录。

13. 单击 Change Target 以显示 Options for Implementation 对话框，如 图 5-13 所示。

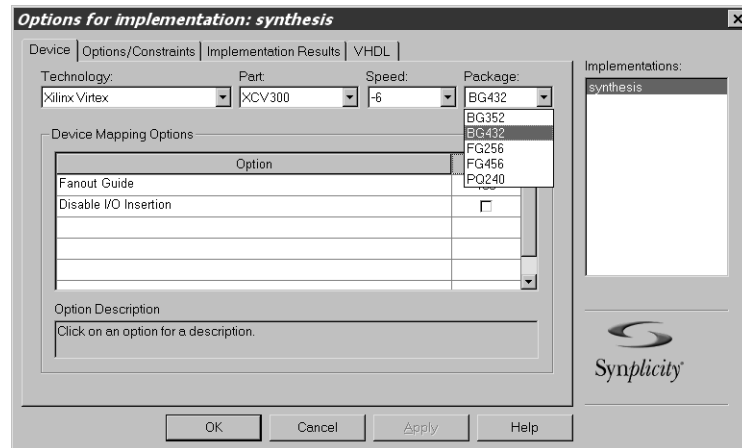


图 5-13: 实现选项

14. 在 Device 标签上设置 Technology、Part、Speed 和 Package 选项，以反映目标器件（在此示例中是 V300BG432-6）。

请务必勾掉 Disable I/O Insertion。

15. 在 Options/Constraints 标签上，勾掉 Symbolic FSM Compiler（但保持 Resource Sharing 为选中状态），并且将 Frequency 设置为 66 MHz。

16. 在 Implementation Results 标签上勾掉 Write Vendor Constraint File。

17. 单击 OK 返回到主项目窗口；然后单击 Run。

Synplify 综合设计并写出优化的 EDIF 文件。在窗口的右下角显示综合的各个阶段，如 Compiling 或 Mapping。过程完成后显示 Done。请注意，预计 Synplify 会发出几条有关用于其他综合工具的例化 I/O 基元和属性的警告。

Exemplar LeonardoSpectrum

在尝试综合设计之前，请确保已正确配置 Exemplar LeonardoSpectrum 环境。

1. 转到综合目录：

```
cd < 安装路径 >/hdl/example/synthesis
```

综合目录包含一个用于 LeonardoSpectrum 的脚本。

2. 编辑该脚本以更改下面一行：

```
cd < 安装路径 >/hdl/example/synthesis
```

将路径修改为指向实际安装位置，然后保存文件。

3. 启动 LeonardoSpectrum。

4. 通过运行脚本 leonardo.tcl 来综合设计。

注：如果用图形用户界面运行 LeonardoSpectrum，则不能使用快速启动表单来综合设计，而是需要从菜单中选择 File > Run Script。

综合步骤的最终结果是一个 EDIF 文件，该文件在实现步骤中馈入 Xilinx 实现工具。

在实际中，必须修改提供的脚本文件，以适合其他设计。为了帮助了解综合脚本，下面叙述其主要步骤：

1. 使用环境变量设置各种综合选项。

这些选项必须出现在脚本中，并且不得修改。同时也加载综合库；综合库可以修改以适合不同的器件和速度级别。

2. 通过读入设计文件来加载设计。
此时，顶层模块声明为 `present_design`。脚本将 `nopad` 属性（具有 `FALSE` 值）赋予所有 PCI 总线接口信号。这些端口的 I/O 结构直接在封装文件中例化。
3. 用 `-hierarchy preserve` 和 `-chip` 选项完成优化步骤。
`-hierarchy preserve` 选项防止 LeonardoSpectrum 取消设计层级。`-chip` 选项指明应自动插入 I/O 缓冲器。
4. 综合完成后，写出已综合的网表。
5. 预计工具会发出关于未用信号的警告。

Xilinx XST

在尝试综合设计之前，请确保已正确配置 Xilinx XST 环境。只能通过 XST 命令行进行综合。

1. 转到综合目录：

```
cd < 安装路径 >/hdl/example/synthesis
```


综合目录包含一个用于 Xilinx XST 的脚本；此脚本对于 PC 平台称为 `run_xst.bat`，对于 UNIX 平台称为 `run_xst.sh`。`run_xst.cmd` 和 `run_xst.prj` 文件由两种脚本共用。
2. 如果需要，可以修改这两个文件以适合您的应用。您可能需要更改目标架构，并且选择不同的封装和配置文件。
3. 通过运行脚本来综合设计。

综合步骤的最终结果是一个 NGC 文件，该文件在实现步骤中馈入 Xilinx 实现工具。预计工具会发出关于未用信号的警告。

实现设计

本章讲述如何用支持的 FPGA 实现工具（ISE Foundation 开发系统附带）实现使用全局时钟的 *ping64* 示例设计。对于 32 位 Initiator/Target core for PCI，请用 *ping32* 替换 *ping64*。如果您用的是使用参考时钟的设计，请用 *pcim_top_r* 替换 *pcim_top*，并用 *ping_tb_r* 替换 *ping_tb*。

ISE Foundation

实现设计之前，请确保已正确配置 Xilinx 环境并且已成功综合设计。

1. 请转到实现目录：

```
cd <安装路径>/hdl/example/xilinx
```

此目录包含一个用于 Xilinx ISE 的脚本，用来为示例设计进行布局和布线；此脚本对于 PC 平台称为 *run_xilinx.bat*，对于 UNIX 平台称为 *run_xilinx.sh*。

注：如果以 XC3S1200E 作为 PCI 66 MHz 的目标，请使用 *run_xilinx_s3e66.bat* 或 *run_xilinx_s3e66.sh* 作为脚本文件。此脚本对已布线的 NCD 文件进行特别后处理，以满足保持时间要求。

2. 请检查相应的 *run_xilinx* 脚本文件，注意以下各项：

- ◆ 在脚本的起始处设置了几个必须的特殊环境变量；请勿将其删除。
- ◆ *ngdbuild* 命令将 *../src/xpci* 和 *../synthesis* 列为搜索目录。*xpci* 目录包含 PCI 接口的网表，而 *synthesis* 目录则必须包含在设计综合期间生成的 EDIF 网表。

ngdbuild 命令还读取与所需目标器件和特定核接口版本对应的 UCF。

- ◆ 要指向不同的器件，或者要使用不同的核接口版本，必须将 UCF 修改成与所选器件和接口相匹配。第 3 章“针对系列的考虑事项”中规定了可用的选择。

核接口附带的 UCF 包括保证管脚和时序指标的约束。在处理期间必须始终使用这些约束。

与用户应用有关的其他约束也必须放在此文件中。在向用户约束文件添加内容之前，请备份原文件，以便在必要时恢复该文件。

- ◆ map 命令不需要任何特别参数，但要使用一个输入 / 输出寄存器封装选项。
- ◆ 按脚本所示，par 命令使用一个具有准确引导模式的引导文件。请注意，有些设计不需要使用引导文件。

要指向不同的器件，或者要使用不同的核接口版本，请见第 3 章“针对系列的考虑事项”。

请务必使用正确的引导文件名称。如果不需要引导文件，或者您选择使用用户约束文件中嵌入的定向布线约束，请从脚本中的 par 命令行删除以下输入参数：

```
-gm exact -gf ../../src/guide/guidefile.ncd
```

如果使用定向布线，请确保已将 ngdbuild 输入参数修改成指向替代的约束文件：

```
-uc ../../src/dirt/constraintfile.ucf
```

必要时可以调整 PAR 努力程度和其他实现选项。

注：run_xilinx_s3e66 脚本在 par 后运行以下命令：

```
xdl -ncd2xdl pcim_top_unloaded.ncd
```

```
cd s3e_66
```

```
xilperl 3s1200efg400.pl -xin ../pcim_top_unloaded.xdl -xout  
../pcim_top_routed.xdl -pads 3s1200efg400.info
```

```
cd ..
```

```
xdl -xdl2ncd pcim_top_routed.xdl
```

```
fpga_edline pcim_top_routed.ncd -p s3e_66/3s1200efg400.scr
```

这样可使 XC3S1200E 满足 66 MHz PCI 的保持时间要求。脚本达到此目的的方法是用未连接的 I/O 触发器加载数据总线时钟使能，以便重新将其建立保持窗口与全局时钟正确对齐。

- ◆ trce 命令根据用户约束文件中原指定的设计约束进行静态时序分析。
- ◆ netgen 命令生成已布局布线设计的仿真模型。

3. 通过运行相应的脚本来实现设计。

最初尝试处理时，最好不运行脚本，而是每次从命令行输入一条命令，以便检查每一步的输出。

使用引导文件

如果需要引导文件，重要的是验证引导过程是否成功。为此，可以检查 pcim_top_routed.grf 文件。本节其余部分针对需要使用引导文件的设计。

预布线连接数应完全符合第 17 页表 3-1 所示选择表中列出的数字。未布线信号数因用户应用大小不同而不同。如果该数字不符，就说明引导过程失败了。失败可能有几个原因。首先，检查是否使用了正确的用户约束和引导文件。其次，检查用户应用是否遵守了《Initiator/Target core for PCI v3.1 用户指南》中规定的所有信号驱动规则。

注：请勿尝试在已进行引导布线的设计上重新布线。不得重新布线的原因是这样做可能导致对已由引导文件引导布线的那些网状连线再度布线。