



EDA 应用技术

<http://www.phei.com.cn>

基于Quartus II 的数字系统Verilog HDL 设计实例详解 (第2版)

周润景 姜 攀 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

014025648

TP332.1
92-2

内容简介

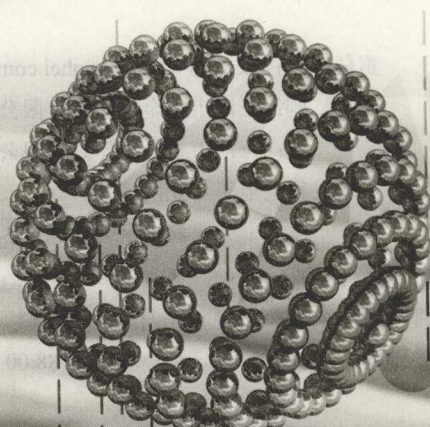
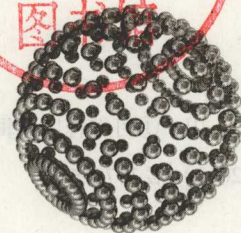
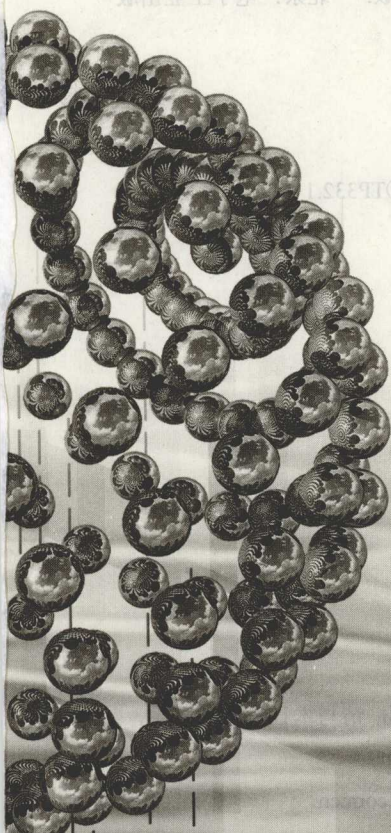
EDA应用技术

基于Quartus II 的数字系统Verilog HDL 设计实例详解 (第2版)

周润景 姜 攀 编著



TP332.1
92-2



电子工业出版社

Publishing House of Electronics Industry



北航

C1711150

内 容 简 介

本书以语法与实例结合的方式来讲解可编程逻辑器件的设计方法,软件开发平台为 Altera 公司的 Quartus II 9.0 FPGA/CPLD 设计软件。本书由浅入深地介绍了利用 Quartus II 进行数字系统开发的设计流程、设计思想和设计技巧。书中的例子非常丰富,既有简单的数字逻辑电路实例,也有复杂的数字系统设计实例。

本书适合从事数字系统设计的工程技术人员阅读使用,也可作为高等学校相关专业的 EDA 技术开发、课程设计、毕业设计及电子设计竞赛等的教学用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

基于 Quartus II 的数字系统 Verilog HDL 设计实例详解 / 周润景,姜攀编著. —2 版. —北京:电子工业出版社,2014.1

(EDA 应用技术)

ISBN 978-7-121-22269-6

I. ①基… II. ①周… ②姜… III. ①可编程序逻辑器件—系统设计 IV. ①TP332.1

中国版本图书馆 CIP 数据核字(2013)第 317876 号

责任编辑:张 剑(zhang@phei.com.cn)

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:33 字数:845 千字

印 次:2014 年 1 月第 1 次印刷

印 数:3 000 册 定价:88.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

随着电子技术、计算机应用技术和 EDA 技术的不断发展,利用 FPGA/CPLD 进行数字系统的开发已广泛应用于通信、航天、医疗电子、工业控制等领域。与传统电路设计方法相比,FPGA/CPLD 具有功能强大,开发过程投资小、周期短、便于修改,以及开发工具智能化等特点。近年来,FPGA/CPLD 市场发展迅速,并且随着电子工艺不断改进,低成本高性能的 FPGA/CPLD 器件不断涌现,FPGA/CPLD 业已成为当今硬件设计的首选方式之一。熟练掌握 FPGA/CPLD 设计技术已经是电子设计工程师的基本要求。

Verilog HDL 语言作为国际标准的硬件描述语言,已经成为相关专业的工程技术人员和高校学生必须掌握的编程语言之一。本书范例中的文本编辑均采用 Verilog HDL 语言编写,并且均已通过仿真和硬件测试。

本书主要以实例为主来介绍以 Quartus II 9.0 为设计平台的 FPGA/CPLD 数字系统设计。本书共分为 14 章和一个附录,其中第 1~2 章主要介绍 Quartus II 9.0 的开发流程和设计方法;第 3 章介绍第三方仿真工具 ModelSim 和综合工具 Synplify 的使用;第 4~5 章介绍 Verilog HDL 设计的语法基本知识;第 6~10 章以数字电路的设计(包括门电路、组合逻辑电路、触发器、时序逻辑电路)为实例,介绍原理图编辑、文本编辑及混合编辑的设计方法,同时也巩固了数字电路的基础知识;第 11 章介绍一些课程设计中涉及的数字系统设计实例,以便读者更深入地掌握 Quartus II 9.0 的设计方法和熟练运用 Verilog HDL 语言;第 12 章介绍宏功能模块及 IP 核的使用方法和简单的范例;第 13~14 章给出了两个大型数字系统的设计实例,以便读者更深入地掌握数字系统的设计方法;附录 A 为本书实验开发箱的有关说明。

本书由周润景和姜攀编著。其中,姜攀编写了第 4 章和第 5 章,全书由周润景教授统稿、定稿。参加本书编写的还有张丽娜、张红敏、张丽敏、宋志清、陈雪梅、刘怡芳、陈艳梅、贾雯、张龙龙、托亚、魏晓敏和周敬。

本书的例子经过北京百科融创公司开发的 RC-EDA/SOPC-IV 实验箱的验证,对该公司的支持表示感谢。

本书适合从事数字系统设计的工程技术人员阅读使用,也可作为高等学校相关专业的 EDA 技术开发、课程设计、毕业设计及电子设计竞赛等的教学用书。为便于读者阅读、学习,特提供本书范例的下载资源,请访问 <http://yydz.phei.com.cn> 网站,到“资源下载”栏目下载。

由于作者水平有限,书中难免存在错误和不足之处,敬请读者批评指正。

编著者

目 录

第1章 Altera Quartus II 开发流程	1
1.1 Quartus II 软件综述	1
1.2 设计输入	6
1.3 约束输入	8
1.4 综合	14
1.5 布局布线	17
1.6 仿真	21
1.7 编程与配置	26
第2章 Quartus II 的使用	31
2.1 原理图和图表模块编辑	31
2.2 文本编辑	55
2.3 混合编辑(自底向上设计)	71
2.4 混合编辑(自顶向下设计)	76
第3章 第三方 EDA 工具的使用	85
3.1 第三方 EDA 工具简介	85
3.2 ModelSim 仿真工具的使用	86
3.2.1 仿真简介	87
3.2.2 ModelSim 简介	90
3.2.3 使用 ModelSim 进行功能仿真	95
3.2.4 使用 ModelSim 进行时序仿真	105
3.2.5 在 Quartus II 中调用 ModelSim 进行仿真	111
3.2.6 ModelSim 仿真工具的高级应用	115
3.3 Synplify/Synplify Pro 综合工具的使用	126
3.3.1 Synplify/Synplify Pro 简介	126
3.3.2 Synplify Pro 综合流程	134
3.3.3 Synplify Pro 的其他综合技巧	157
第4章 Verilog HDL 语言概述及基本要素	167
4.1 Verilog HDL 语言简介	167
4.2 Verilog HDL 设计流程	169
4.3 程序模块的说明	170
4.4 Verilog HDL 的层次化设计	172
4.5 时延	178
4.6 Verilog HDL 语言的描述形式	178
4.6.1 结构描述形式	179

4.6.2	行为描述形式	181
4.6.3	混合设计模式	187
4.7	Verilog HDL 语言基本要素	188
4.7.1	标志符	188
4.7.2	注释	188
4.7.3	格式	189
4.7.4	系统任务和函数	189
4.7.5	编译指令	189
4.7.6	逻辑数值	192
4.7.7	常量	192
4.7.8	数据类型	194
4.7.9	运算符和表达式	200
第 5 章	行为描述语句	210
5.1	触发事件控制	210
5.2	条件语句	212
5.3	循环语句	217
5.4	逻辑验证与 Testbench 编写	221
5.5	状态机	238
第 6 章	门电路设计范例	252
6.1	与非门电路	252
6.2	或非门电路	254
6.3	异或门电路	256
6.4	三态门电路	258
6.5	单向总线缓冲器	260
6.6	双向总线缓冲器	261
6.7	使用 always 过程语句描述的简单算术逻辑单元	262
第 7 章	组合逻辑电路设计范例	264
7.1	编码器	264
7.1.1	8 线—3 线编码器	264
7.1.2	8 线—3 线优先编码器	265
7.2	译码器	269
7.2.1	3 线—8 线译码器	269
7.2.2	BCD—七段显示译码器	273
7.3	数据选择器	274
7.3.1	4 选 1 数据选择器	274
7.3.2	8 选 1 数据选择器	277
7.3.3	2 选 1 数据选择器	279
7.4	数据分配器	280
7.5	数值比较器	281
7.6	加法器	283

7.6.1	半加器	283
7.6.2	全加器	286
7.6.3	4位全加器	288
7.6.4	16位加法器	290
7.7	减法器	292
7.7.1	半减器	292
7.7.2	全减器	294
7.7.3	4位全减器	295
7.8	七人投票表决器	296
7.9	乘法器	297
第8章	触发器设计范例	299
8.1	R-S 触发器	299
8.2	J-K 触发器	301
8.3	D 触发器	302
8.4	T 触发器	304
第9章	时序逻辑电路设计范例	306
9.1	同步计数器	306
9.1.1	同步4位二进制计数器	306
9.1.2	同步二十四进制计数器	308
9.1.3	模为60的BCD码加法计数器	310
9.2	异步计数器	312
9.3	减法计数器	314
9.4	可逆计数器	315
9.5	可变模计数器	317
9.5.1	无置数端的可变模计数器	317
9.5.2	有置数端的可变模计数器	319
9.6	寄存器	320
9.7	锁存器	322
9.8	移位寄存器	324
9.8.1	双向移位寄存器	325
9.8.2	串入/串出移位寄存器	326
9.8.3	串入/并出移位寄存器	328
9.8.4	并入/串出移位寄存器	329
9.9	顺序脉冲发生器	330
9.10	序列信号发生器	332
9.11	分频器	333
9.11.1	偶数分频器	333
9.11.2	奇数分频器	336
9.11.3	半整数分频器	342

第 10 章 存储器设计范例	344
10.1 只读存储器 (ROM)	344
10.2 随机存储器 (RAM)	346
10.3 堆栈	348
10.4 FIFO	351
第 11 章 数字系统设计范例	354
11.1 跑马灯设计	354
11.2 8 位数码扫描显示电路设计	358
11.3 4×4 键盘扫描电路设计	360
11.4 数字频率计	364
11.5 乒乓游戏机	366
11.6 交通控制器	372
11.7 数字钟	380
11.8 自动售货机	389
11.9 出租车计费器	397
11.10 电梯控制器	409
第 12 章 可参数化宏模块及 IP 核的使用	421
12.1 ROM、RAM、FIFO 的使用	421
12.2 乘法器和锁相环的使用	429
12.3 正弦信号发生器	432
12.4 NCO IP 核的使用	433
第 13 章 基于 FPGA 的射频热疗系统	438
13.1 肿瘤热疗的生物学与物理学技术概论	439
13.2 温度场特性的仿真	441
13.3 射频热疗系统设计	441
13.4 系统硬件电路设计	442
13.4.1 硬件整体结构	442
13.4.2 高精度数字温度传感器 DS18B20	443
13.4.3 ACEX 1K 系列的 FPGA 器件的特点	447
13.4.4 ACEX 1K 器件的配置电路设计	448
13.4.5 电源电路	451
13.4.6 驱动电路设计	451
13.5 软件实现	454
13.5.1 系统软件设计电路图	454
13.5.2 温度测量模块	456
13.5.3 指定温度设置模块	462
13.5.4 控制算法的选择及设计	465
13.5.5 信号调制	475
13.5.6 温度显示模块	475
13.5.7 分频模块	481

13.6 温度场测量与控制的实验	482
13.6.1 实验材料及方法	482
13.6.2 实验结果	483
13.6.3 实验结果分析	486
13.7 结论	487
第 14 章 基于 FPGA 的直流电机伺服系统	488
14.1 电机控制发展情况	488
14.2 系统控制原理	489
14.3 算法设计	491
14.4 系统硬件设计原理	493
14.5 系统软件设计原理	500
14.5.1 系统软件设计电路图	501
14.5.2 AD1674 控制模块	503
14.5.3 ADC0809 控制模块	505
14.5.4 反馈控制模块	507
14.5.5 前馈控制模块	508
14.5.6 前馈和反馈量求和模块	509
14.5.7 过电流控制模块	510
14.5.8 PWM 波生成模块	510
14.5.9 分频模块	511
14.6 系统调试及结果分析	512
14.6.1 硬件调试	512
14.6.2 可靠性、维修性、安全性分析	513
14.6.3 软件调试	514
14.7 结论	517
附录 A RC-EDA/SOPC 实验平台简介	518

第1章 Altera Quartus II 开发流程

【知识目标】

- ☺ 通过对本书第1章至第3章的学习，了解EDA开发工具的作用与开发流程。
- ☺ 熟悉Quartus II、ModelSim、Synplify等EDA工具的操作界面。

【能力目标】

了解常用EDA工具的操作界面，熟练掌握各个软件的操作流程。通过对本书第1章至第3章的学习，建立起对学习Verilog HDL的整体感知，了解在各个开发环节如何选择合适的开发工具。

- ☺ 初级要求：了解Quartus II、ModelSim、Synplify各自的开发流程。
- ☺ 中级要求：了解Quartus II、ModelSim、Synplify这3款软件各自的应用特点。
- ☺ 高级要求：掌握EDA开发工具之间如何建立合作，并发挥各自的优势来完成设计。



1.1 Quartus II 软件综述

Quartus II 是 Altera 公司在 21 世纪初推出的 FPGA/CPLD 开发环境，是 Altera 前一代 FPGA/CPLD 集成开发环境 MAX+plus II 的更新换代产品，具有功能强大、界面友好、使用便捷等优点。Quartus II 软件集成了 Altera 的 FPGA/CPLD 开发流程中所涉及的所有工具和第三方软件接口。利用此开发工具，设计者可以创建、组织和管理自己的设计。Quartus II 软件的开发流程如图 1-1-1 所示。

1. Quartus II 软件的特点及支持的器件

【Quartus II 的特点】

☺ 支持多时钟定时分析、LogicLock™ 基于块的设计、SOPC（可编程片上系统）、内嵌 SignalTap II 逻辑分析器、功率估计器等高级工具；

☺ 易于引脚分配和时序约束；

☺ 具有强大的 HDL 综合能力；

☺ 包含 Maxplus II 的 GUI，且易于将 Maxplus II 工程平稳地过渡到 Quartus II 开发环境中；

☺ 对于 Fmax 的设计具有很好的效果；

☺ 支持的器件种类众多；

☺ 支持 Windows、Solaris、Hpux 和 Linux 等多种操作系统；

☺ 支持与第三方工具如综合、仿真等的链接。

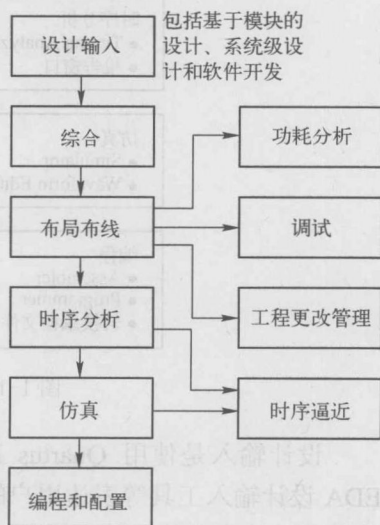


图 1-1-1 Quartus II 软件的开发流程

Quartus II 软件支持的器件主要有 Stratix、Stratix II、Stratix III、Cyclone、Cyclone II、Cyclone III、HardCopy II、APEX II、FLEX10k、FLEX6000、MAX II、MAX3000A、MAX7000、MAX9000 等系列。

2. Quartus II 软件的工具及功能简介

Altera 的 Quartus II 软件提供了完整的多平台设计环境,可以轻松满足特定的设计需求,是 SOPC 设计的综合环境。此外,Quartus II 软件允许用户在设计流程的每个阶段使用其软件图形用户界面、EDA 工具界面或命令行方式。图 1-1-2 所示为 Quartus II 软件图形用户界面为设计流程的各个阶段提供的功能。



图 1-1-2 Quartus II 软件图形用户界面的功能

设计输入是使用 Quartus II 软件的模块输入方式、文本输入方式、Core 输入方式和 EDA 设计输入工具等表达用户的电路构思,同时使用分配编辑器 (Assignment Editor) 设定初始设计约束条件。

综合是将 HDL 语言、原理图等设计输入翻译成由与、或、非门, RAM, 触发器等基本逻辑单元组成的逻辑链接 (网络表), 并根据目标与要求 (约束条件) 优化所生成的逻辑链接, 输出 .edf 或 .vqm 等标准格式的网络表文件, 供布局布线器进行实现。除了可以用 Quartus II 软件的 “Analysis & Synthesis” 命令综合外, 也可以使用第三方综合工具生成与

Quartus II 软件配合使用的.edf 网络表文件或.vqm 文件。

布局布线的输入文件是综合后的网络表文件。Quartus II 软件中布局布线包含分析布局布线结果、优化布局布线、增量布局布线和通过反向标注分配等。

时序分析是允许用户分析设计中所有逻辑的时序性能,并协助引导布局布线满足设计中的时序分析要求。默认情况下,时序分析作为全编译的一部分自动运行,它观察和报告时序信息,如建立时间、保持时间、时钟至输出延时、最大时钟频率及设计的其他时序特性,可以使用时序分析生成的信息分析、调试和验证设计的时序性能。

仿真分为功能仿真和时序仿真。功能仿真主要是验证电路功能是否符合设计要求;时序分析包含了延时信息,它能较好地反映芯片的设计工作情况。可以使用 Quartus II 集成的仿真工具进行仿真,也可以使用第三方工具对设计进行仿真,如 ModelSim 仿真工具。

编程和配置是在全编译成功后,对 Altera 器件进行编程或配置,包括 Assemble (生成编程文件)、Programmer (建立包含设计所用器件名称和选项的链式文件)、转换编程文件等。

系统级设计包括 SOPC Builder 和 DSP Builder。Quartus II 与 SOPC Builder 一起为建立 SOPC 设计提供标准化的图形环境,其中 SOPC 由 CPU、存储器接口、标准外围设备和用户自定义的外围设备等组件组成。SOPC Builder 允许选择和自定义系统模块的各个组件和接口,它将这些组件组合起来,生成对这些组件进行实例化的单个系统模块,并自动生成必要的总线逻辑。DSP Builder 是帮助用户在易于算法应用的开发环境中建立 DSP 设计的硬件表示,缩短了 DSP 设计周期。

软件开发中的 Quartus II 软件 Software Builder 是集成编程工具,可以将软件源文件转换为用户配置 Excalibur 器件的闪存格式编程文件或无源格式编程文件。Software Builder 在创建编程文件时,自动生成仿真器初始化文件。仿真器初始化文件指定了存储单元的每个地址的初始值。

LogicLock 模块化设计流程支持对复杂设计的某个模块独立地进行设计、实现与优化,并将该模块的实现结果约束在规划好的 FPGA 区域内。

EDA 界面中的 EDA Netlist Writer 是生成时序仿真所需要的包含延迟信息的文件,如.vo、.sdo 文件等。

时序逼近是通过控制综合和设计的布局布线来达到时序目标。使用时序逼近流程可以对复杂的设计进行更快的时序逼近,减少优化迭代次数,并自动平衡多个设计约束。

SignalTap II 逻辑分析器和 SignalProb 功能可以分析内部器件节点和 I/O 引脚,同时在系统内以系统速度运行。SignalTap II 逻辑分析器可以捕获和显示 FPGA 内部的实时信号行为。SignalProbe 可以在不影响设计中现有布局布线的情况下,将内部电路中特定的信号迅速布线到输出引脚,从而无须对整个设计另作一次全编译。

工程更改管理是在全编译后对设计作的少量修改或调整。这种修改是直接在设计数据库上进行的,而不是修改源代码或配置文件,这样就无须重新运行全编译而快速地实施这些更改。

除了 Quartus II 软件集成的上述工具外,Quartus II 软件还提供第三方工具的链接,如综合工具 Synplify、SynplifyPro、LeonardoSpectrum,仿真工具 ModelSim、Aldec HDL 等,它们都是业内公认的专业综合、仿真工具,以其功能强大、界面友好、易学易用而得到广泛使用。

3. Quartus II 软件的用户界面

启动 Quartus II 软件后, 其默认界面如图 1-1-3 所示。它由标题栏、菜单栏、工具栏、资源管理窗、编译状态显示窗、信息显示窗和工程工作区等部分组成。

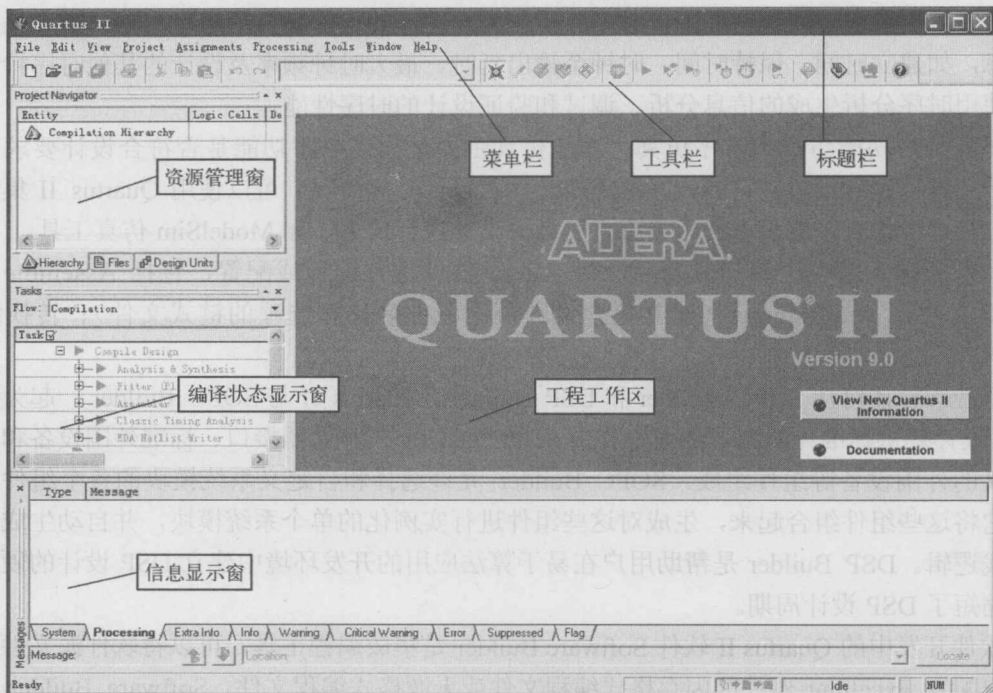


图 1-1-3 Quartus II 软件图形用户界面

- 1) 标题栏 用于显示当前工程的路径和程序的名称。
- 2) 菜单栏 主要由文件 (File)、编辑 (Edit)、视图 (View)、工程 (Project)、资源分配 (Assignments)、操作 (Processing)、工具 (Tools)、窗口 (Window) 和帮助 (Help) 9 个菜单组成。其中 (Project)、资源分配 (Assignments)、操作 (Processing)、工具 (Tools) 菜单集中了 Quartus II 软件较为核心的全部操作命令。

(1) “Project” 菜单: 主要是对工程的一些操作。

- ☉ Add Current File to Project: 添加当前文件到本工程。
- ☉ Add/Remove Files in Project: 添加或新建资源文件。
- ☉ Revisions: 创建或删除工程。
- ☉ Archive Project: 为工程归档或备份。
- ☉ Restore Archived Project: 恢复工程备份。
- ☉ Import Database/Export Database...: 导入或导出数据库。
- ☉ Import Design Partition/Export Design Partition: 导入或导出设计分区。
- ☉ Generate Bottom-Up Design Partition Scripts: 生成自底向上设计分区脚本。
- ☉ Generate Tcl File for Project: 生成工程的 Tcl 脚本文件, 选择好要生成的文件名及路径后, 单击 “OK” 按钮即可。
- ☉ Generate PowerPlay Early Power Estimator File: 生成估算静态和动态功耗的表单。

☺ Organize Quartus II Settings File: 管理 Quartus II 的设置文件, 可以生成一个分组注释的设置文件 (.qsf), 包括设计文件路径、引脚分配、引脚电平类型和器件类型等一些基本的配置内容。

☺ HardCopy Utilities/ HardCopy II Utilities: 与 HardCopy 和 HardCopy II 器件相关的功能。

☺ Locate: 将 Assignment Editor 中的节点或源代码中的信号在 Timing Closure Floorplan、编译后布局布线图、Chip Editor 或源文件中定位。

☺ Set as Top-Level Entity: 把工程工作区打开的文件设定为顶层文件。

☺ Hierarchy: 打开工程工作区显示的源文件的上一层或下一层的源文件及顶层文件。

(2) “Assignments” 菜单: 主要是对工程的参数进行配置, 如引脚分配、时序约束、参数设置等。

☺ Device: 设置目标器件型号。

☺ Pins: 打开分配引脚对话框, 给设计的信号分配 I/O 引脚。

☺ Timing Analysis Settings: 打开时序分析设置对话框。

☺ EDA Tool Settings: 设置 EDA 工具, 如 Synplify、ModelSim 等。

☺ Settings: 打开参数设置页面, 可以切换到使用 Quartus II 软件开发流程的每个步骤所需要的参数设置页面。

☺ Classic Timing Analyzer Wizard: 时序分析向导, 启动后可以进行时序约束设置等。

☺ Assignment Editor: 分配编辑器, 用于分配引脚、设定引脚电平标准、设置时序约束等。

☺ Pin Planner: 打开引脚分配对话框。

☺ Remove Assignments: 删除设定的类型的分配, 如引脚分配、时序分配、SignalProbe 分配等。

☺ Demote Assignments: 允许用户降级使用当前较不严格的约束, 使编辑器更高效地编译分配和约束等。

☺ Back-Annotate Assignments: 允许用户在工程中反向标注引脚、逻辑单元、LogicLock 区域、节点、布线分配等。

☺ Import Assignments/Export Assignments: 给当前工程导入分配文件。

☺ Assignments (Time) Groups: 建立引脚分配组。

☺ Timing Closure Floorplan: 启动时序逼近平面布局规划器。

☺ LogicLock Regions Window: 允许用户查看、创建和编辑 LogicLock 区域约束, 以及导入/导出 LogicLock 区域约束文件。

☺ Design Partition Window: 打开设计分区窗口。

(3) “Processing” 菜单: 包含了对当前工程执行各种设计流程, 如开始综合、开始布局、开始布线、开始时序分析等。

(4) “Tools” 菜单: 调用 Quartus II 软件中集成的一些工具, 如 MegaWizard Plug-In Manager (用于生成 IP 核和宏功能模块)、Chip Editor、RTL Viewer、Programmer 等工具。

3) 工具栏 (Tool Bar) 包含了常用命令的快捷图标。将光标移到相应图标时, 在光标下方出现此图标对应的含义, 而且每种图标在菜单栏中均能找到相应的命令菜单。用户可以根据需要将自己常用的功能定制为工具栏上的图标, 方便在 Quartus II 软件中灵活、快速地进行各种操作。

4) 资源管理窗 用于显示当前工程中所有相关的资源文件。资源管理窗左下角有 3 个选项卡, 分别是结构层次 (Hierarchy) 文件 (File) 和设计单元 (Design Units)。结构层次窗口在工程编译前只显示顶层模块名, 工程编译了一次后, 此窗口按层次列出工程中所有的模块, 并列出每个源文件所用资源的具体情况。顶层可以是用户产生的文本文件, 也可以是图形编辑文件。文件窗口列出了工程编译后的所有文件, 文件类型有设计器件文件 (Design Device Files)、软件文件 (Software Files) 和其他文件 (Others Files)。设计单元窗口列出了工程编译后的所有单元, 如 AHDL 单元、Verilog 单元、VHDL 单元等, 一个设计器件文件对应生成一个设计单元, 参数定义文件没有对应设计单元。

5) 工程工作区 器件设置、定时约束设置、底层编辑器和编译报告等均显示在工程工作区中。当 Quartus II 实现不同功能时, 此区域将打开相应的操作窗口, 显示不同的内容, 可以进行不同的操作。

6) 编译状态显示窗 主要显示模块综合、布局布线过程及时间。模块 (Module) 列出工程模块, 过程 (Process) 显示综合、布局布线进度条, 时间 (Time) 表示综合、布局布线所耗费时间。

7) 信息显示窗 显示 Quartus II 软件综合、布局布线过程中的信息, 如开始综合时调用源文件、库文件、综合布局布线过程中的定时、报警、错误等, 如果是报警和错误, 则会给出具体的引起报警和错误的原因, 方便设计者查找及修改错误。



1.2 设计输入

Quartus II 软件中的工程由所有设计文件与与设计文件有关的设置组成。用户可以使用 Quartus II 原理图输入方式、文本输入方式、模块输入方式和 EDA 设计输入工具等表达自己的电路构思。设计输入的流程如图 1-2-1 所示。

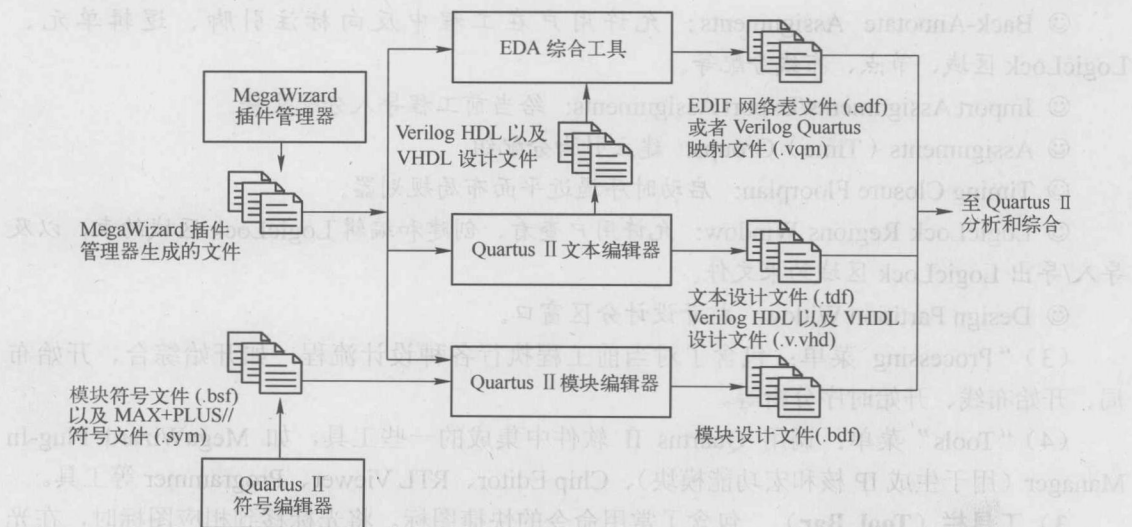


图 1-2-1 设计输入流程

进行设计前, 执行菜单命令 “File” → “New Project Wizard”, 可以创建新工程。创建

工程时,要指定工程工作目录,分配工程名称,指定顶层设计实体的名称。还可以指定在工程中使用的设计文件、其他源文件、用户库和 EDA 工具,以及目标器件。

1. 设计输入方式

创建好工程后,需要给工程添加设计输入文件。设计输入文件可以使用文本形式的文件(如 VHDL、Verilog、HDL、AHDL 等),存储数据文件(如 HEX、MIF 等),原理图设计输入,以及第三方 EDA 工具产生的文件(如 EDIF、HDL、VQM 等)。同时,还可以混合使用以上多种设计输入方法进行设计。

【Verilog HDL/VHDL 硬件描述语言设计输入方式】大型设计中一般都采用 HDL 语言设计方法。HDL 语言设计方法是大型模块化设计工程中最常用的设计方法。目前较为流行的 HDL 语言有 VHDL、Verilog HDL 语言等,其共同特点是易于使用自顶向下的设计方法,易于模块划分和复用,移植性强,通用性好,设计不因芯片工艺和结构的改变而变化,利于向 ASIC 的移植。HDL 语言是纯文本文件,用任何编辑器都可以编辑,有些编辑器集成了语言检查、语法辅助模板等功能,这些功能给 HDL 语言的设计和调试带来了很大的便利。

【AHDL 输入方式】AHDL (Altera Hard Description Language)是完全集成到 Quartus II 软件系统中的一种高级模块化语言。可以用 Quartus II 软件文本编辑器或其他文本编辑器生成 AHDL 文件。一个工程中可以全部使用 AHDL 语言,也可以和其他类型的设计文件混用。AHDL 语言只能用于使用 Altera 器件的 FPGA/CPLD 设计,其代码不能移植到其他厂商器件上(如 Xilinx、Lattice 等)使用,通用性不强,所以比较少用。

【模块/原理图输入方式 (Block Diagram/Schematic Files)】原理图输入方式是 FPGA/CPLD 设计的基本方法之一,几乎所有的设计环境都集成有原理图输入法。这种设计方法直观、易用,支撑它的是一个功能强大、分门别类的器件库。然而由于器件库元件通用性差,导致其移植性差,如更换设计实现的芯片信号或厂商不同时,整个原理图需要作很大修改,甚至是全部重新设计。所以原理图设计方式主要是一种辅助设计方式,多用于混合设计中的个别模块设计。

【使用 Mega Wizard Plug-In Manager 产生 IP 核/宏功能模块】Mega Wizard Plug-In Manager 工具的使用基本可以分为工程的创建和管理,查找使用的 IP 核/宏功能模块及参数设计与生成,IP 核/宏功能模块的仿真与综合等步骤。

2. 设计规划

在建立设计时,必须考虑 Quartus II 软件提供的设计法,如 LogicLock 功能提供从自顶向下和自底向上的设计方法,以及基于块的设计流程。在自顶向下的设计流程中,整个设计只有一个输出网络表,用户可以对整个设计进行跨设计边界和结构层次的优化处理,且管理容易;在自底向上的设计方法中,每个设计模块具有单独的网络表,允许用户单独编译每个模块,且单个模块的修改不会影响其他模块的优化。基于块的设计流程使用 EDA 设计输入和综合工具分别设计和综合各个模块,然后将各模块整合到 Quartus II 软件的最高层设计中。在设计时,用户可根据实际情况灵活使用这些设计方法。

在本书第 2 章中,将以具体实例来详细地介绍几种常用的设计方法。



1.3 约束输入

建立好工程和设计后, 需要给设计分配引脚和时序约束。可以使用分配编辑器、“Setting”对话框、TimeQuest 分析器、引脚规划器、设计划分窗口和时序逼近平面布局来指定初始设计约束, 如引脚分配、器件选项、逻辑选项和时序约束等。另外, 还可以执行菜单命令“Assignments”“Import Assignments”或者“Export Assignments”, 进行导入和导出分配。Quartus

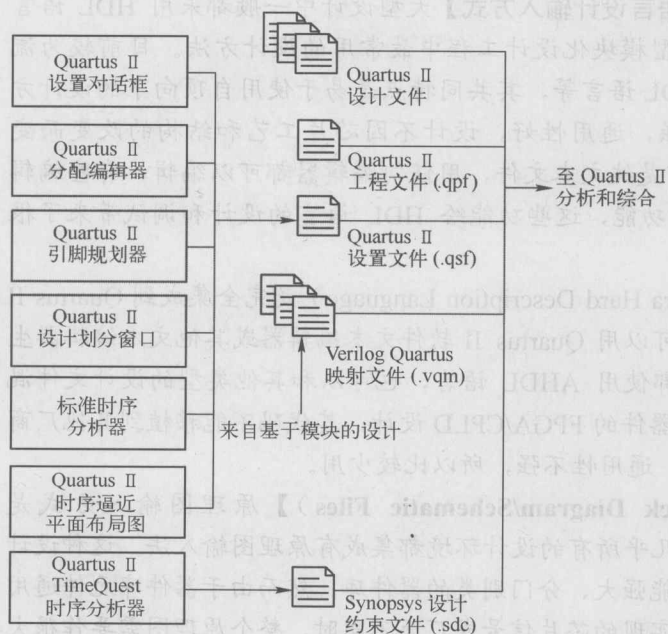


图 1-3-1 约束和分配输入流程

II 软件还提供时序向导, 协助用户指定初始标准时序约束。还可以使用 Tcl 命令或脚本从其他 EDA 综合工具中导入分配。图 1-3-1 所示的是约束和分配输入流程。

分配引脚是将设计文件的 I/O 信号指定到器件的某个引脚, 设置此引脚的电平标准、电流强度等。

时序约束尤其重要, 它是为了使高速数字电路设计满足运行速率方面的要求, 在综合、布局布线阶段附加约束。要分析工程是否满足用户的运行速率要求, 也需要对工程的设计输入文件添加时序约束, 时序分析工具是以用户的时序约束判断时序是否满足设计要求标准的, 因此要求设计者正确输入约束, 以便得到正确的时序分析报告。

附加约束还能提高设计的工作速率, 这对分析设计的时序是否满足设计要求非常重要, 而且时序约束越全面, 对分析设计的时序就越有帮助。如果设计中有多个时钟, 有一个时钟没有约束, 则 Quartus II 软件的时序分析工具将不对没有约束的时钟路径作分析, 从而使得设计者不知道此部分时序是否满足要求, 因此设计者在约束时序时一定要全面。

引脚分配时序约束的通常做法是设计者编写约束文件并导入到综合、布局布线工具中, 在 FPGA/CPLD 综合、布局布线步骤时指导逻辑映射、布局布线。也可以使用 Quartus II 软件中集成的工具, 如分配编辑器 (Assignment Editor)、引脚规划器 (Pin Planner) 和“Settings”对话框等, 进行引脚分配和时序约束。

1. 分配编辑器 (Assignments Editor)

分配编辑器界面用于在 Quartus II 软件中建立、编辑节点和实体级分配。分配用于在设计中为逻辑指定各种选项和设置, 包括位置、I/O 标准、时序、逻辑选项、参数、仿真和引脚分配。可以使能或禁止单独分配功能, 也可以为分配加入注释。使用分配编辑器可以进行标准格式时序分配。对于 Synopsys 设计约束, 必须使用 TimeQuest 时序分析器。