

基于FPGA的 SOPC实践教程

杨 军 编著
李 彤 主审



科学出版社
www.sciencep.com

17

基于 FPGA 的 SOPC 实践教程

杨 军 编著
李 彤 主审

(1) Quartus II, 用于完成 Nios II 系统的综合、硬件优化、适配、编程下载及硬件系统调试等;

(2) SOPC Builder, 用于实现 Nios II 系统的配置;

(3) ModelSim, 用于 Nios II 系统的仿真;

(4) Nios II IDE, 用于 Nios II 系统的开发。

目前市场上同类的 SOPC 设计教材, 对设计技术细节和实际应用与大量应用不符, 本书的编写力求做到理论与实践相结合, 力求做到理论与实践相结合, 力求做到理论与实践相结合。

TP332

Y205

科学出版社

北京

内 容 简 介

本书共分8章。第1章为概述,第2章和第3章分别详细介绍了 Quartus II 8.0、ModelSim SE 6.0、SOPC 软件的使用方法,并针对每个软件选择一个经典的实例引导读者熟悉使用软件进行设计的过程;第4章介绍了8个基于FPGA的基础实验,引领读者快速入门;第5章介绍了8个综合实验,进一步促使读者熟练使用FPGA设计数字系统,掌握基本设计技巧;第6章是SOPC系统实验,循序渐进地讲解了8个实例系统的设计,每个系统都是在前一个系统之上加上特殊功能构成一个新系统,这是本书的一大特点,读者可全面了解各个模块在系统设计中所起的作用,从而掌握SOPC系统设计的关键技术;第7章介绍了使用Quartus II 8.0、ModelSim SE 6.0、Nios II 8.0设计过程中的常见错误提示和解决方法;第8章是TD-EDA/SOPC综合实验平台和DE2开发板简介。

本书可作为普通高等院校计算机科学与技术、信息安全、电子信息工程、通信工程、自动化等专业学生的教材,也可供从事FPGA开发的科研人员使用。

图书在版编目(CIP)数据

基于FPGA的SOPC实践教程/杨军编著. —北京:科学出版社,2010.6
ISBN 978-7-03-028024-4

I. ①基… II. ①杨… III. ①可编程序逻辑器件-系统设计-高等学校-教材②微处理器-系统设计-高等学校-教材 IV. ①TP332

中国版本图书馆CIP数据核字(2010)第114307号

责任编辑:毛莹 潘继敏 / 责任校对:陈玉凤

责任印制:张克忠 / 封面设计:陈敬

科学出版社出版

北京东黄城根北街16号
邮政编码:100717

<http://www.sciencep.com>

丽源印刷厂印刷

科学出版社发行 各地新华书店经销

*

2010年6月第 一 版 开本:B5(720×1000)

2010年6月第一次印刷 印张:14

印数:1—3 000 字数:268 000

定价:28.00元(含光盘)

(如有印装质量问题,我社负责调换)

前 言

基于 FPGA(Field Programmable Gate Array)的电子系统设计技术是 21 世纪电子应用工程师必备的基本技能之一,而基于 FPGA 的 SOPC(System On Programmable Chip)设计技术是当前电子系统设计领域最前沿的技术之一。Altera 公司、Xilinx 公司、Lattice 公司等全球最重要的 FPGA 及 EDA 公司都分别推出 SOPC 系统解决方案。SOPC 设计技术将是 21 世纪的技术发展趋势,是现在高等院校和社会嵌入式培训班主要的培训内容和亮点。

本书的 SOPC 设计具体包括以 32 位 Nios II 为核心的嵌入式系统的硬件配置、硬件设计、硬件仿真、软件设计以及软件调试等。SOPC 系统设计的基本软件工具包括以下几方面。

(1) Quartus II:用于完成 Nios II 系统的综合、硬件优化、适配、编程下载及硬件系统调试等;

(2) SOPC Builder: Altera Nios II 嵌入式处理器开发软件包,用于实现 Nios II 系统的配置、生成;

(3) ModelSim:用于对 SOPC 生成的 Nios II 系统的 HDL 描述进行系统功能仿真;

(4) Nios II IDE:用于进行软件开发、调试及向目标开发板进行 Flash 下载。

目前市场上同类的 SOPC 书籍比较少,而且清一色地介绍编程语言和基础原理,对设计技术细节和实际工程案例涉及甚微,与 SOPC 设计技术现在的蓬勃发展与大量应用不符,本书的出版正可以填补这方面的空白。本书以实用和应用为基本原则,根据作者多年积累的开发经验,通过讲练结合、循序渐进的形式来讲解,便于读者快速入门与提高。

本书从可编程器件+EDA 软件+硬件描述语言+SOPC Builder 的现代数字系统的设计方法出发,使读者在掌握了 VHDL 或 Verilog HDL 后,进一步学习本书介绍的最新设计软件 Quartus II、SOPC Builder、ModelSim 以及 Nios II IDE,这对他们今后的设计工作有很大帮助。

本书是作者结合近几年的实践教学经验,针对学生面临的实际问题,参考了大量设计方面的书籍和技术文献组织编写的,在这里向这些资料的作者表示衷心的感谢。本书的实验内容充分吸纳借鉴了西安唐都公司和 Altera 公司工程师的经验和资料,尤其感谢唐都公司的技术人员,他们在实例设计中给予了大量的技术支持,提高了本书的水平和实用价值。

作为本书的主审,李彤教授在本书的框架设计、内容安排、材料组织等方面给予了极大的帮助和支持,并提出了许多宝贵意见。另外,参加本书编写的人员有丁俊、郭义雄、李娜、唐佐侠、董寅、郭跃东、宋克俭、皇甫正茂、柳爱华等,他们在资料的收集、整理,源代码的设计、分析、仿真,硬件平台的验证,以及书稿的录入、排版、绘图等方面做了大量工作,在此一并向他们表示衷心的感谢!

基于FPGA的SOPC电子系统设计技术涉及的知识范围广,本书目的是为初学者提供一些帮助和指导。本书的光盘中包含了丰富的实例工程文件和程序源代码,读者稍加修改便可应用于自己的工作中或者完成自己的课题。

由于作者水平有限,加之编写时间仓促,书中难免存在疏漏之处,诚望读者批评指正。

作者

2010年3月

第4部分 常见问题与常用实验平台简介	3.3
第7章 常见问题	191
7.1 Quartus II 常见问题	191
7.2 ModelSim 常见问题	199
前言	199
第一部分 FPGA 设计软件及工具入门	204
第1章 概述	1
第2章 常用FPGA开发工具	4
2.1 硬件开发工具 Quartus II 8.0	4
2.1.1 Quartus II 8.0 简介	4
2.1.2 Quartus II 8.0 设计流程	5
2.1.3 Quartus II 设计方法	9
2.1.4 Quartus II 功能详解	11
2.1.5 时序约束与分析	20
2.1.6 设计优化	28
2.1.7 SignalTap II	36
2.1.8 实例讲解	40
2.2 ModelSim 开发工具	49
2.2.1 ModelSim 简介	50
2.2.2 基本仿真步骤	50
2.2.3 ModelSim 各界面介绍	53
2.2.4 ModelSim 调试功能	59
2.2.5 实例讲解	63
2.3 本章小结	70
第3章 SOPC 系统设计入门	71
3.1 SOPC 技术简介	71
3.1.1 SOPC 技术的主要特点	71
3.1.2 SOPC 技术的实现方式	72
3.1.3 SOPC 系统的开发流程	74
3.2 基于 SOPC 的 Nios II 处理器设计	74
3.2.1 SOPC Builder 功能	75
3.2.2 SOPC Builder 组成	77
3.2.3 SOPC Builder 组件	81

3.2.4 SOPC Builder 应用实例	83
-------------------------------	----

3.3 本章小结	95
----------------	----

第二部分 FPGA 实验

第4章 数字系统基础实验设计	96
----------------------	----

4.1 编码器实验	96
-----------------	----

4.2 译码器实验	97
-----------------	----

4.3 加法器实验	99
-----------------	----

4.4 乘法器实验	102
-----------------	-----

4.5 寄存器实验	105
-----------------	-----

4.6 计数器实验	107
-----------------	-----

4.7 分频器实验	108
-----------------	-----

4.8 存储器实验	111
-----------------	-----

第5章 数字系统综合实验设计	114
----------------------	-----

5.1 键盘扫描输入实验	114
--------------------	-----

5.2 扫描数码显示器实验	116
---------------------	-----

5.3 点阵显示实验	118
------------------	-----

5.4 交通灯控制实验	120
-------------------	-----

5.5 数字钟实验	123
-----------------	-----

5.6 液晶显示实验	125
------------------	-----

5.7 PS/2 接口实验	133
---------------------	-----

5.8 VGA 显示实验	135
--------------------	-----

第三部分 基于 FPGA 的 SOPC 系统实验

第6章 SOPC 嵌入式系统实验	139
------------------------	-----

6.1 流水灯实验	139
-----------------	-----

6.2 JTAG UART 通信实验	150
--------------------------	-----

6.3 LCM 显示实验	159
--------------------	-----

6.4 按键中断实验	162
------------------	-----

6.5 计数显示实验	168
------------------	-----

6.6 串口通信实验	171
------------------	-----

6.7 外部 Flash 扩展实验	175
-------------------------	-----

6.8 添加用户组件外设实验	181
----------------------	-----

第四部分 常见问题与常用实验平台简介

第 7 章 常见问题	191
7.1 Quartus II 常见问题	191
7.2 ModelSim 常见问题	197
7.3 Nios II 常见问题	199
第 8 章 FPGA 常用综合实验平台	204
8.1 TD-EDA/SOPC 综合实验平台简介	204
8.2 DE2 开发板简介	210
参考文献	213

FPGA 的 SOPC 系统设计实验与实践教程是计算机科学与技术、信息工程、电子信息工程、通信工程、自动化等专业的学生必修的一门专业基础课。它要求学生掌握数字逻辑的基本理论、基本分析与设计方法,具备用 VHDL(或 Verilog HDL)语言进行数字逻辑设计的能力,为后续专业课程的学习和今后从事数字系统设计工作打下良好基础。《基于 FPGA 的 SOPC 实践教程》是一门理论与实践相结合的教程,目的是提高学生对所学内容的感性认识和对知识点的理解,培养学生分析问题、解决问题的能力。

本课程为《数字逻辑》课程的第一门实验课,旨在通过实验,使学生掌握数字系统设计的思想和方法,熟悉数字系统设计自上而下的层次概念及模块化的设计思路。随着电子技术的发展,芯片的复杂程度越来越高,用可编程逻辑器件设计出的数字系统电路,具有简化系统设计、增强系统可靠性及灵活性的优良性能。可编程技术是当前电子工程设计人员设计数字系统时所采用的先进技术手段,体现了现代 EDA 电子技术的发展动态,有着较强的实际应用价值。为使学生跟上电子技术的发展步伐,我们将先进的基于 FPGA 的 SOPC 系统设计技术引入实验教学,目的就是让学生在初步掌握数字系统设计思想和方法的同时,能够在计算机上使用 Quartus II 8.0 进行 VHDL(或 Verilog HDL)语言的编程、编译,掌握使用专业仿真软件 ModelSim SE 6.0 对数字系统进行功能和时序仿真,进一步学习使用 SOPC Builder 进行嵌入式系统设计。因此实验就是设计的过程,通过对这些设计软件平台和学习与运用,要求学生掌握使用 EDA 软件进行数字系统的设计与调试方法,掌握基于 VHDL(或 Verilog HDL)语言的模块设计方法,最终学会多种数字系统的分析、设计、电路调试及错误排查方法。目的是培养学生在整个实验过程中耐心、细致的科研作风,鼓励他们勇于开拓创新,培养学生的实践动手能力和团队合作精神,以及分析和解决实际问题的能力。

本书从三个层次来引导读者学习基于 FPGA 的 SOPC 系统设计技术。

第一,基础数字系统的设计。通过对 FPGA 芯片及相应开发工具的学习,读

第一部分 FPGA 设计软件及工具入门

第1章 概述

基于FPGA的SOPC系统设计实验与实践教程是计算机科学与技术、信息安全、电子信息工程、通信工程、自动化等专业的学生必修的一门专业基础课。它要求学生掌握数字逻辑的基本理论、基本分析与设计方法,具备用VHDL(或Verilog HDL)语言进行数字逻辑设计的能力,为后续专业课程的学习和今后从事数字系统设计工作打下良好基础。《基于FPGA的SOPC实践教程》是一门理论与实践相结合的教程,目的是提高学生对所学内容的感性认识和对知识点的理解,培养学生分析问题、解决问题的能力。

开设基于FPGA的SOPC实践课程,可以巩固、加深和拓宽课堂教学的内容;可以帮助学生更好地了解数字系统设计的思想和方法,熟悉数字系统设计自上而下的层次概念及模块化的设计思路。随着电子技术的发展,芯片的复杂程度越来越高,用可编程逻辑器件设计出的数字系统电路,具有简化系统设计、增强系统可靠性及灵活性的优良性能。可编程技术是当前电子工程设计人员设计数字系统时所采用的先进技术手段,体现了现代EDA电子技术的发展动态,有着较强的实际应用价值。为使学生跟上电子技术的发展步伐,我们将先进的基于FPGA的SOPC系统设计技术引入实验教学,目的就是让学生在初步掌握数字系统设计思想和方法的同时,能够在计算机上使用Quartus II 8.0进行VHDL(或Verilog HDL)语言的编程、编译,掌握使用专业仿真软件ModelSim SE 6.0对数字系统进行功能和时序仿真,进一步学习使用SOPC Builder进行嵌入式系统设计。因此实验就是设计的过程,通过对这些设计软件平台和学习与运用,要求学生掌握使用EDA软件进行数字系统的设计与调试方法;掌握基于VHDL(或Verilog HDL)语言的模块设计方法;最终学会多种数字系统的分析、设计、电路调试及错误排查方法。目的是培养学生在整个实验过程中耐心、细致的科研作风,鼓励他们勇于开拓创新;培养学生的实践动手能力和团队合作精神,以及分析和解决实际问题的能力。

本书从三个层次来引导读者学习基于FPGA的SOPC系统设计技术。

第一,基础数字系统的设计。通过对FPGA芯片及相应开发工具的学习,读

者对于 EDA 技术及其相关概念有所了解;结合读者对数字逻辑电路知识的掌握,通过具体采用自上而下的设计方法并应用图形输入、硬件描述语言等设计手段,来设计、实现一些简单的数字逻辑及数字系统应用电路,使读者能初步掌握 FPGA 芯片的结构、功能以及 EDA 技术的应用,为后续学习打下基础。

第二,综合数字系统的设计。通过一些实例,将 FPGA 应用于接口、控制、通信等领域,使读者掌握 EDA 技术的软硬件开发环境、开发方法及对 FPGA 芯片进行外围接口控制和 IP 模块的使用,让读者能独立设计一些实际工作中可能用到的模块,为更高层次的应用打下基础。

第三,基于 FPGA 的嵌入式 SOPC 系统设计。随着嵌入式处理器、专用数字器件和 DSP 算法以 IP 核的形式嵌入 FPGA 中,以单片 FPGA 完成整个嵌入式系统数字部分的设计已经成为现实。此部分教学通过学习 Altera 公司提供的 Nios II 软核处理器,了解 SOPC 的基本概念和基于 FPGA 的嵌入式系统的开发方法,掌握 SOPC 硬件开发工具、软件调试工具的使用。

本书强调从基础、综合、嵌入式 SOPC 三个层面讲解基于 FPGA 的设计技术。在实例的安排上,着重突出“应用”和“实用”的原则;在实例的讲解上,既介绍了设计原理、基本步骤和流程,也穿插了一些经验技巧和注意事项,在潜移默化的过程中提高读者的理论知识和实践能力。对于有一定基础的读者,建议直接从实例部分入手,一边阅读一边实际操作,这样印象会比较深,效果更好。基础知识欠缺的读者请先详细学习书中基础部分的理论知识,然后再进行应用实例的学习。在学习中,尽量做到反复理解和演练,以达到融会贯通、举一反三的功效。特别希望尽量和自己的工作设计联系起来,以达到“即学即会,学以致用”的最大化境界。

本书对实验过程中将用到的软件进行了详细的实例讲解,建议读者在开始实验之前先认真学习本书前面的理论知识,并按实例讲解进行演练,这将有助于读者快速掌握设计软件的使用。

基于 FPGA 的 SOPC 系统设计实验可分为实验准备、设计调试和实验结束后的总结分析与书写实验报告三个阶段。实验前要认真预习和充分准备,实验过程中仔细操作并认真记录,对实验中出现的故障和问题,要逐级按流程查找,在排除故障和问题的过程中,应对错误和问题的现象、查找错误的方法、修改后的设计方案等做详细的分析记录。为完成好每次实验任务,学生需要做好以下三方面的工作。

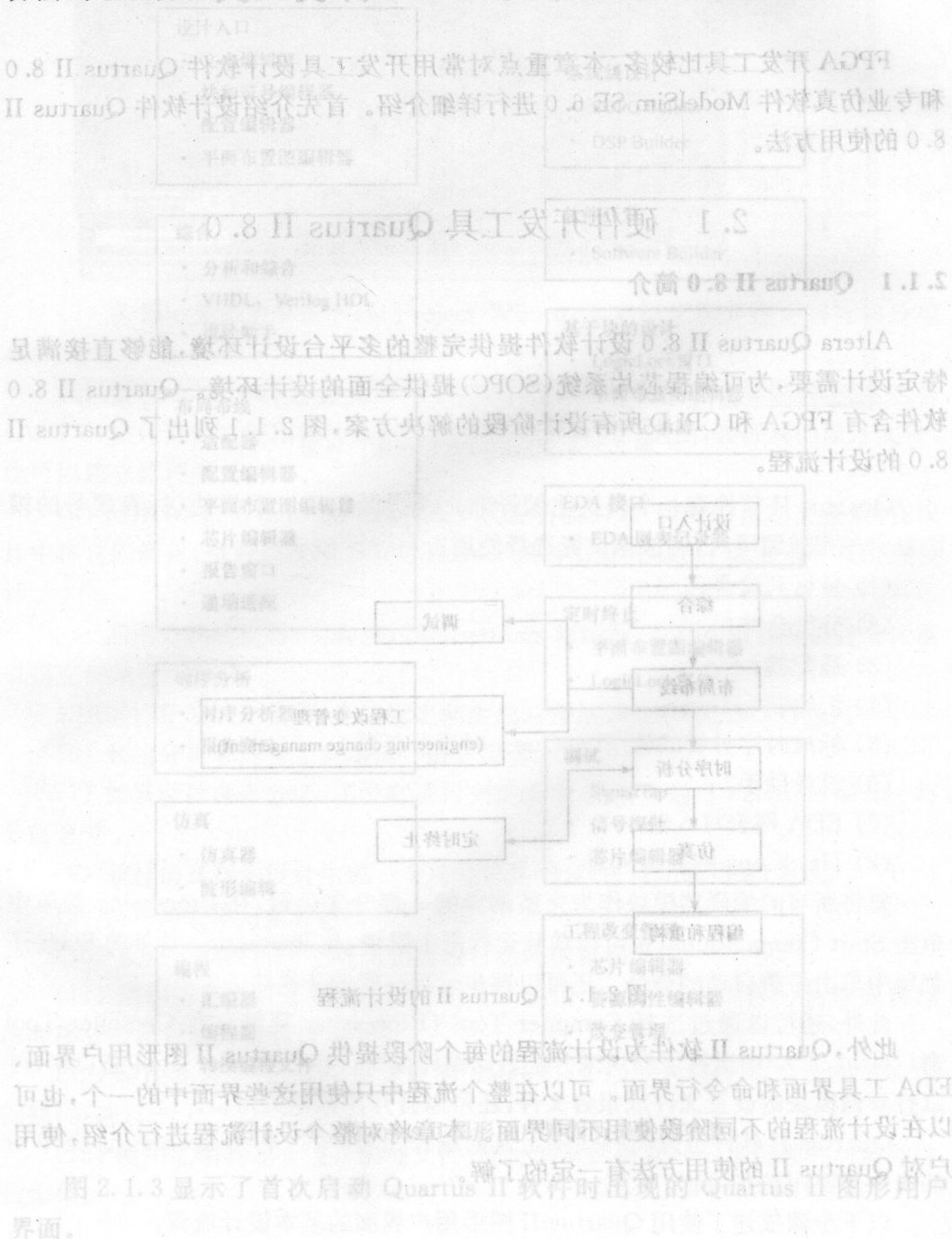
(1) 实验课前必须认真预习,写出实验预习报告。
学生根据实验任务书中的任务,复习相关的理论知识,了解实验目的、实验原理、实验任务及要求、方法、实验设备,并完成设计任务,写出预习报告。

(2) 实验课中认真仔细地操作,完成实验任务。
实验过程中积极思考、认真操作、互相配合。对实验中遇到的故障、问题及解

决方案,进行分析、总结和归纳。

(3) 课后认真进行实验总结、分析,书写实验报告。

对实验结果进行总结、分析,书写实验报告,实验报告要体现出设计者的设计方法、手段及分析问题和解决问题的能力,还应包含实验现象及结论。



者对于 EDA 技术及其相关概念有所了解,结合读者对数字逻辑电路知识的掌握,通过具体采用自上而下的设计方法并应用图形输入来理解数字逻辑设计,实现一些简单的数字逻辑设计。本章将介绍 Quartus II 8.0 的使用方法,使读者掌握 EDA 技术,并能进行硬件开发环境、开发方法及对 FPGA 芯片进行设计、实现一些简单的数字逻辑设计。

第 2 章 常用 FPGA 开发工具

FPGA 开发工具比较多,本章重点对常用开发工具设计软件 Quartus II 8.0 和专业仿真软件 ModelSim SE 6.0 进行详细介绍。首先介绍设计软件 Quartus II 8.0 的使用方法。

2.1 硬件开发工具 Quartus II 8.0

2.1.1 Quartus II 8.0 简介

Altera Quartus II 8.0 设计软件提供完整的多平台设计环境,能够直接满足特定设计需要,为可编程芯片系统(SOPC)提供全面的设计环境。Quartus II 8.0 软件含有 FPGA 和 CPLD 所有设计阶段的解决方案,图 2.1.1 列出了 Quartus II 8.0 的设计流程。

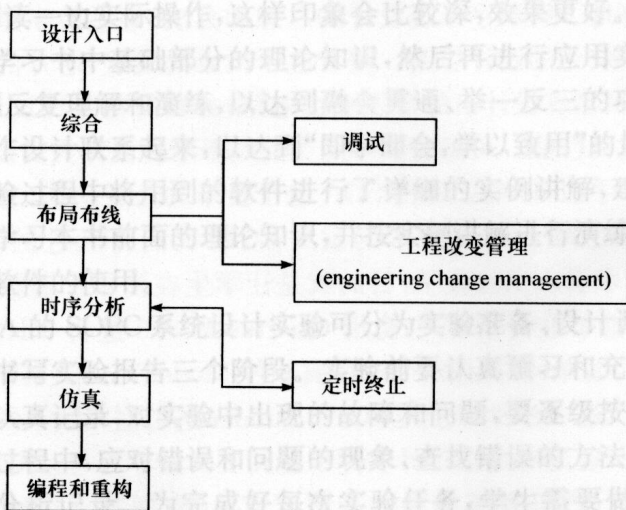


图 2.1.1 Quartus II 的设计流程

此外,Quartus II 软件为设计流程的每个阶段提供 Quartus II 图形用户界面、EDA 工具界面和命令行界面。可以在整个流程中只使用这些界面中的一个,也可以在设计流程的不同阶段使用不同界面。本章将对整个设计流程进行介绍,使用户对 Quartus II 的使用方法有一定的了解。

2.1.2 Quartus II 8.0 设计流程

可以使用 Quartus II 软件完成设计流程的所有阶段,它是一个全面、易用的独立解决方案。图 2.1.2 显示了 Quartus II 图形用户界面所提供的功能。

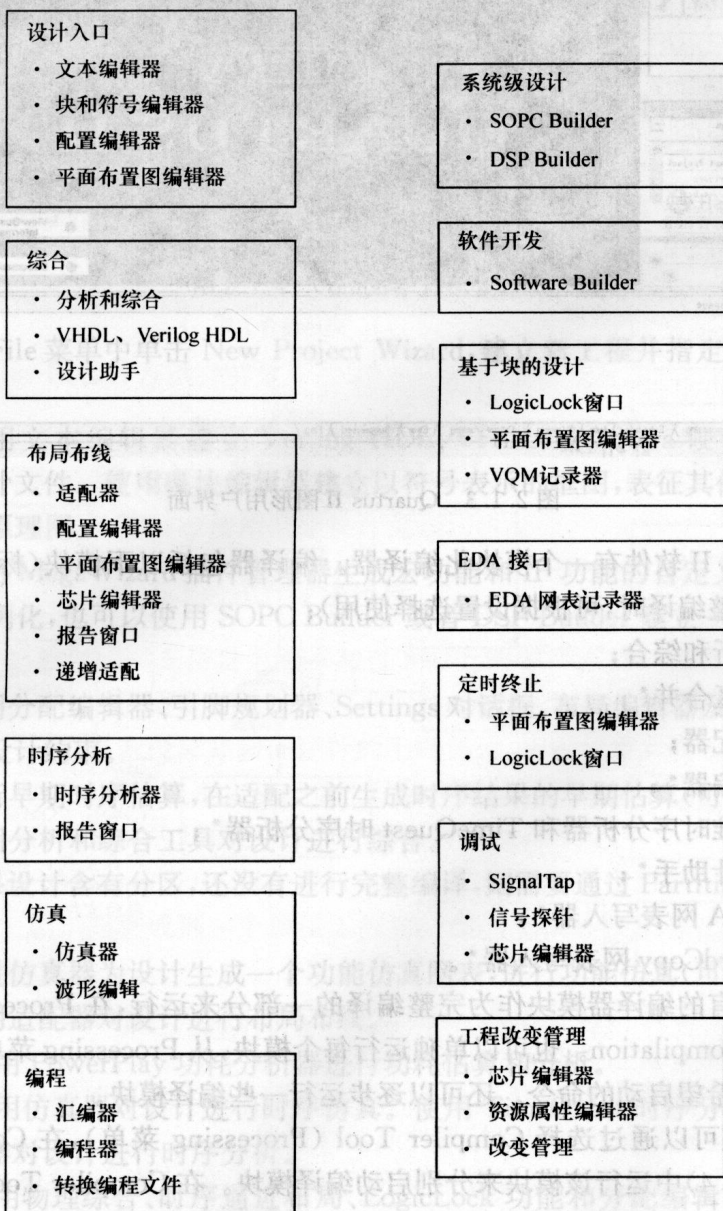


图 2.1.2 Quartus II 图形用户界面所提供的功能

图 2.1.3 显示了首次启动 Quartus II 软件时出现的 Quartus II 图形用户界面。

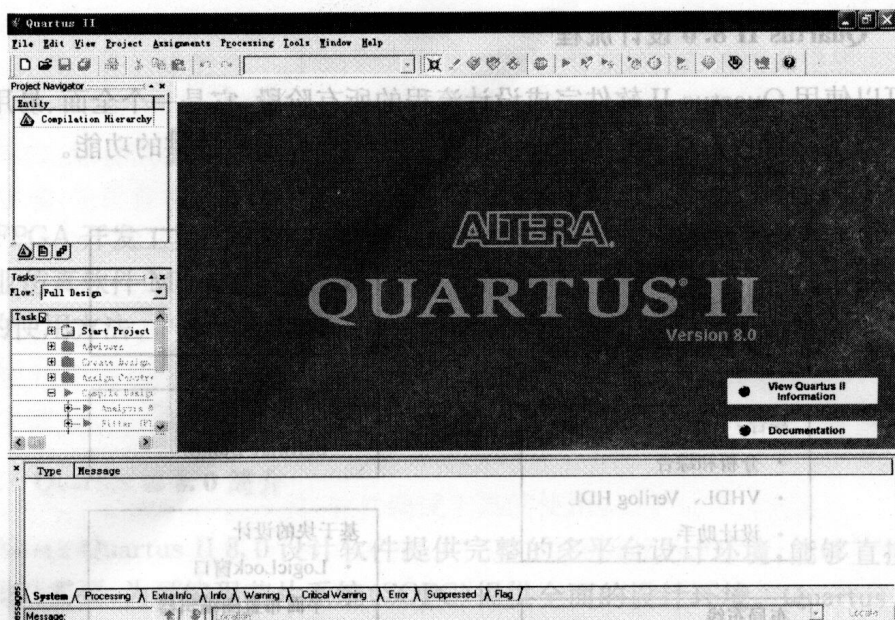


图 2.1.3 Quartus II 图形用户界面

Quartus II 软件有一个模块化编译器。编译器包括以下模块(标有星号的模块表示在完整编译时,可根据设置选择使用):

- (1) 分析和综合;
- (2) 分区合并*;
- (3) 适配器;
- (4) 汇编器*;
- (5) 标准时序分析器和 TimeQuest 时序分析器*;
- (6) 设计助手*;
- (7) EDA 网表写入器*;
- (8) HardCopy 网表写入器*。

要将所有的编译器模块作为完整编译的一部分来运行,在 Processing 菜单中单击 Start Compilation。也可以单独运行每个模块,从 Processing 菜单的 Start 子菜单中单击希望启动的命令。还可以逐步运行一些编译模块。

此外,还可以通过选择 Compiler Tool (Processing 菜单),在 Compiler Tool 窗口(图 2.1.4)中运行该模块来分别启动编译模块。在 Compiler Tool 窗口中,可以打开该模块的设置文件或报告文件,还可以打开其他相关窗口。

Quartus II 软件也提供一些预定义的编译流程,可以利用 Processing 菜单中的命令来使用这些流程。

以下步骤描述了使用 Quartus II 图形用户界面的基本设计流程。

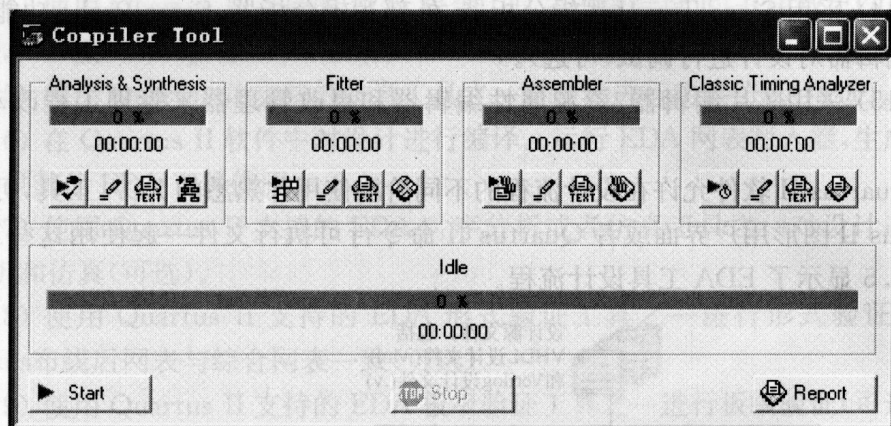


图 2.1.4 Compiler Tool 窗口

(1) 在 File 菜单中单击 New Project Wizard, 建立新工程并指定目标器件或器件系列。

(2) 使用文本编辑器建立 Verilog HDL、VHDL 或 Altera 硬件描述语言 (AHDL) 设计文件。使用模块编辑器建立以符号表示的框图, 表征其他设计文件, 也可以建立原理图。

(3) 使用 MegaWizard 插件管理器生成宏功能和 IP 功能的自定义变量, 在设计中将它们例化, 也可以使用 SOPC Builder 或者 DSP Builder 建立一个系统级设计。

(4) 利用分配编辑器、引脚规划器、Settings 对话框、布局编辑器及设计分区窗口指定初始设计约束。

(5) 进行早期时序估算, 在适配之前生成时序结果的早期估算(可选)。

(6) 利用分析和综合工具对设计进行综合。

(7) 如果设计含有分区, 还没有进行完整编译, 则需要通过 Partition Merge 将分区合并。

(8) 通过仿真器为设计生成一个功能仿真网表, 进行功能仿真(可选)。

(9) 使用适配器对设计进行布局布线。

(10) 使用 PowerPlay 功耗分析器进行功耗估算和分析。

(11) 使用仿真器对设计进行时序仿真。使用 TimeQuest 时序分析器或者标准时序分析器对设计进行时序分析。

(12) 使用物理综合、时序逼近布局、LogicLock 功能和分配编辑器纠正时序问题(可选)。

(13) 使用汇编器建立设计编程文件, 通过编辑器和 Altera 编程硬件对器件进行编程。

(4) 使用 Quartus II 支持的 EDA 综合工具之一综合设计,并生成 EDIF 网表文件(.edf)或 Verilog Quartus 映射文件(.vqm)。

(5) 使用 Quartus II 支持的仿真工具之一对设计进行功能仿真(可选)。

(6) 在 Quartus II 软件中对设计进行编译。运行 EDA 网表写入器,生成输出文件,供其他 EDA 工具使用。

(7) 使用 Quartus II 支持的 EDA 时序分析或者仿真工具之一对设计进行时序分析和仿真(可选)。

(8) 使用 Quartus II 支持的 EDA 形式验证工具之一进行形式验证,确保 Quartus 布线后网表与综合网表一致(可选)。

(9) 使用 Quartus II 支持的 EDA 板级验证工具之一进行板级验证(可选)。

(10) 使用 Quartus II 支持的 EDA 物理综合工具之一进行物理综合(可选)。

(11) 使用编程器和 Altera 硬件对器件进行编程。

2.1.3 Quartus II 设计方法

在建立新设计时,应重视和考虑 Quartus II 软件提供的设计方法,包括自上而下或自下而上的渐进式设计流程,以及基于模块的设计流程。不管是否使用 EDA 设计输入和综合工具,都可以使用这些设计流程。

1. 自上而下与自下而上的设计方法比较

Quartus II 软件同时支持自上而下和自下而上的编译流程。在自上而下的编译过程中,一个设计人员或者工程负责人在软件中对整个设计进行编译。不同的设计人员或者 IP 提供者设计并验证设计不同部分,工程负责人在设计实体完成后将其加入工程中。工程负责人从整体上编译并优化顶层工程。设计中完成的部分得到适配结果,当设计其他部分改动时,其性能保持不变。

在自下而上的设计流程中,每个设计人员在各自的工程中对其设计进行优化后,将每一个底层工程集成到一个顶层工程中。渐进式编译提供导出和导入功能来实现这种设计方法。

作为底层模块设计人员,可以针对设计导出优化后的网表和一组分配(如 LogicLock 区域)。然后,工程负责人将每一个设计模块作为设计分区导入顶层工程中。在这种情况下,工程负责人必须指导底层模块设计人员,保证每一分区使用适当的器件资源。

在完整的渐进式编译流程中,应该认识到,如果以前出于保持性能不变的原因而采用自下而上的方法,那么现在可以采用自上而下的方法来达到同样的目的。这一功能之所以重要是出于两方面的原因:第一,自上而下的流程要比对应的自下而上的流程执行起来简单一些,例如,不需要导入和导出底层设计;第二,自上而下