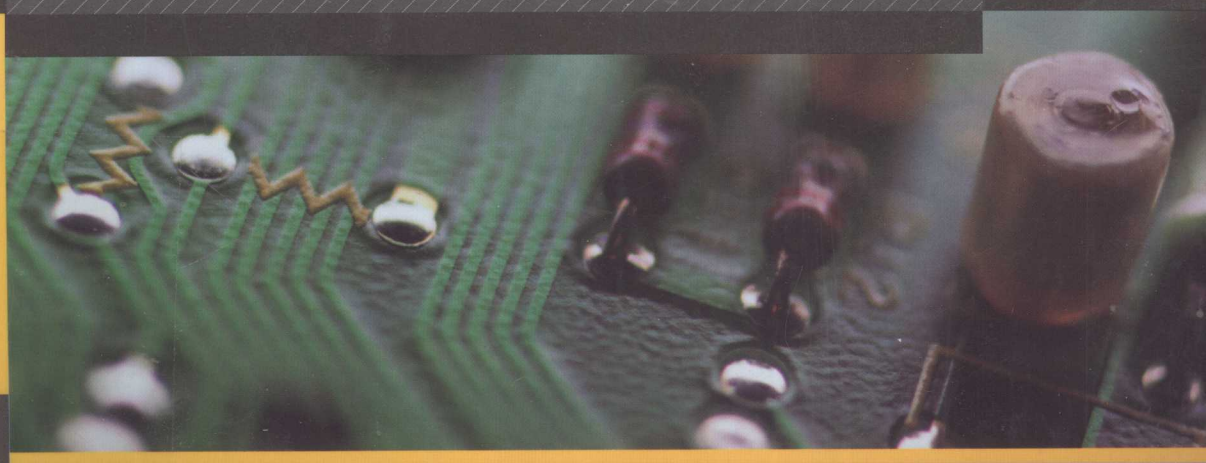
21 世纪高等院校电气工程与自动化规划教材
21 century institutions of higher learning materials of Electrical Engineering and Automation Planning
云南民族大学“十二五”规划教材
云南民族大学教材建设基金资助教材

The EDA Design Technology

EDA设计技术

杨光永 凌永发 编著



人民邮电出版社
POSTS & TELECOM PRESS

21 世纪高等院校电气工程与自动化规划教材
21 century institutions of higher learning materials of Electrical Engineering and Automation Planning

云南民族大学“十二五”规划教材
云南民族大学教材建设基金资助教材

The EDA Design Technology

EDA设计技术

杨光永 凌永发 编著

人民邮电出版社
北京

图书在版编目（CIP）数据

EDA设计技术 / 杨光永，凌永发编著. -- 北京 : 人民邮电出版社，2013.9
21世纪高等院校电气工程与自动化规划教材
ISBN 978-7-115-32706-2

Ⅰ. ①E… Ⅱ. ①杨… ②凌… Ⅲ. ①电子电路－电路设计－计算机辅助设计－高等学校－教材 Ⅳ. ①TN702

中国版本图书馆CIP数据核字(2013)第197517号

内 容 提 要

本书采用 Verilog HDL 语言，较为系统地介绍了 EDA 设计技术的语法基础、基本原理和设计方法。全书共 12 章，分为逻辑器件和语言基础、EDA 设计方法、EDA 设计工具、Nios Ⅱ处理器及嵌入式系统设计四大部分，主要内容包括 Verilog HDL 语言基础，可编程逻辑器件结构，各级抽象的建模设计方法，功能校验和时序校验设计方法，Quartus Ⅱ、ModelSim 和 Nios Ⅱ SBT for Eclipse 设计工具，基于 Nios Ⅱ的嵌入式系统设计流，配置 BSP 工程和应用工程，硬件抽象层及其 API 服务等。各章节配有实例和习题，所有实例均通过编译和验证。

本书可作为电子信息工程、通信工程、电气工程及其自动化等相关专业本科学生的教材，也可作为研究生或工程技术人员的参考书。

◆ 编　　著　杨光永　凌永发
责任编辑　李海涛
责任印制　彭志环　杨林杰

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京艺辉印刷有限公司印刷

◆ 开本：787×1092　1/16
印张：23.5　　2013 年 9 月第 1 版
字数：592 千字　　2013 年 9 月北京第 1 次印刷

定价：49.80 元

读者服务热线：(010)67170985　印装质量热线：(010)67129223
反盗版热线：(010)67171154
广告经营许可证：京崇工商广字第 0021 号

前言

EDA 设计技术是电子系统设计的核心技术。近 30 年来，电子设计技术在设计方法、设计规模、工艺技术等方面飞速进步，广阔的市场需求使 EDA 技术取得了丰硕的技术成果和可喜的经济效益。同时，电子设计技术的巨大变化也向我国高校电类专业电子设计课程的理论教学和实验教学提出了更高要求，鉴于此，自 20 世纪 90 年代后期，全国各高校相继开设“EDA 设计技术”或“数字系统设计技术”课程。由于电子设计技术的高速发展，国内很多高校对课程教学体系进行教学改革，很多教学仪器厂商也顺应市场需求，推出门类齐全的 EDA 实验教学设备和辅助工具。

本书的目的是为了适应教学的需要，供电子工程、通信工程、电气工程、自动化等专业本科学生作为理论教材，让读者循序渐进地掌握电子设计自动化的基本设计方法，培养学生理论与实践紧密结合、具有解决教学大纲要求的电子设计任务的能力。全书共 12 章，按内容分为四大部分：设计技术简介、逻辑器件和语言基础（第 1 章至第 3 章），EDA 设计方法（第 4 章至第 9 章），EDA 设计工具（第 10 章和第 11 章）、Nios Ⅱ处理器及嵌入式系统设计（第 12 章）。各部分内容的编排和选材尽量独立，以适应不同教学任务或自学的需要。此外，作者有意识地扩大各章节内容的覆盖范围，侧重介绍 EDA 设计技术在器件知识、语言基础和设计技术的基本知识点和一般方法，增加教学、科研和工程实践中具有代表性的小型工程案例，同时尝试性地反映 EDA 设计技术在系统集成、知识产权核、校验方法、优化设计、嵌入式系统等方面的技术动态，以补充传统的教学内容和教学体系滞后工业技术需求的不足。

第一部分首先在第 1 章以简短的篇幅介绍了 EDA 技术的发展、技术优势、开发环境等内容，让读者了解 EDA 技术的概貌。CPLD 和 FPGA 的器件基础知识与系统设计的总体方案、器件选型、编程和调试环节有关，因此专门开辟第 2 章介绍可编程逻辑器件的基本分类、主要结构、配置和编程方法。第 3 章参考 IEEE1364 标准，重点介绍 Verilog HDL 语言的语法和语义基础，以及强度、编译指令、系统任务和系统函数。

第二部分依次介绍系统及系统的结构要素、层次化设计方法（自顶向下和自底向上的设计方法）、模块、抽象级别、例化（第 4 章），过程语句、延时控制、条件语句、循环语句、分支语句、任务、函数，以交通灯演示模型和伺服电机调速控制器为行为建模实例（第 5 章），事件控制、等待控制、顺序控制、并行控制，以线阵 CCD 图像传感器时序发生器为并行建模实例（第 6 章），组合逻辑设计、时序逻辑设计（第 7 章），用户定义原语及其

建模方法（第 8 章），设计校验、功能校验、时序校验方法，重点介绍等价验证方法、编写测试平台、代码封装、数据抽象、路径延时设计、时序检查设计方法、标准延时格式、反向标注等内容（第 9 章）。

第三部分主要介绍 Quartus II 软件设计工具和 ModelSim 仿真工具，主要介绍系统设计流程、设计输入方法、布局与布线、时序分析、时序逼近、功耗分析、编程与配置、形式验证、系统集成工具和系统调试工具（第 10 章），以及创建仿真工程和工作库、编译仿真文件、执行仿真、波形分析、时序仿真、例化存储器、值变转储文件等（第 11 章）。该部分是实施系统设计的主要工具。

第四部分即第 12 章，介绍 Nios II 处理器及嵌入式系统设计方法，主要内容包括 Nios II 处理器的算术逻辑单元、寄存器、异常处理、存储器、输入输出口、地址空间、存储器分区、JTAG 调试模块、Nios II 处理器的指令集、处理器内核类型选择、例化方法、Avalon 接口，基于 Nios II 的嵌入式系统设计流，配置 BSP 工程和应用工程，硬件抽象层及其 API 服务，提高系统任务处理速度和提高中断任务处理速度的优化设计方法，以及系统启动配置等。

全书由杨光永副教授、凌永发教授共同执笔完成，其中，杨光永编写第 1 章至第 11 章，凌永发编写第 12 章，全书由杨光永完成统稿、定稿，崔琳、陈乐对本书初稿进行全面阅读和校对，提出了很多宝贵的修改意见，研究生赵锦剑、周安然完成了部分插图、实验工作，胡国清教授和文莉给予本书中工程实验平台的技术支持，云南民族大学电气信息工程学院和教务处的领导对本书的出版给予极大的支持和关心，并提供本书的十二五规划教材项目资金资助。在此，作者谨表示衷心的感谢！

由于编者学识有限，时间仓促，选材内容和介绍定有疏漏或不当之处，恳请各位专家、教学同仁和同学不吝提出宝贵意见，电子邮箱：y.guangyong@mail.scut.edu.cn。

编　者

2013 年 6 月于广州

目　录

第1章 绪 论

电子设计自动化（Electronic Design Automation，EDA）是一种可重用设计方法，由20世纪80年代中期的计算机辅助设计（CAD）、计算机辅助制造（CAM）、计算机辅助测试（CAT）和计算机辅助工程（CAE）等技术综合发展而来。EDA技术是以计算机和计算机辅助设计软件为工具，以可编程器件为载体，利用库元件或硬件描述语言（Hardware Description Language，HDL）依次进行建立模型、设计输入、编译、综合、适配、仿真、编程、调试等环节，实现特定研究和设计任务的新兴电子设计技术，是电子信息、电子元器件制造、自动化装备等产业实现高效率、高可靠性和经济效益的重要设计方法。

1.1 EDA技术

20世纪70年代初，国际上电子和计算机技术较先进的国家，开始探索电子电路设计的新方法，以降低工程技术人员在日益复杂的系统设计任务中的技术风险，由此在设计方法、工具等方面发生了巨大的变革。同时，可编程系统器件（PSD）、复杂可编程逻辑件（CPLD）、现场可编程门阵列（FPGA）的相继问世，为数字系统的设计带来了极大的灵活性。这些器件可以通过软件编程，利用硬件描述语言对其硬件结构和工作方式进行重构、综合、布线、校验和仿真，从而使得硬件的设计可以如同软件设计那样方便快捷。这一切极大地改变了传统的数字系统设计方法、设计过程和设计观念，促进了EDA技术的迅速发展。

得益于计算机制造技术和计算机辅助设计软件的发展，20世纪80年代末的“甩图板工程”，使EDA技术出现了革命性的变革，极大地提高了电路设计的效率和可靠性，减少了工程师的劳动强度和技术风险，提高了工程决策者对市场的快速反应能力。现在，工程师可以在几分钟时间内利用EDA设计软件搭建一个百万门级的数字系统（如以32位Nios Ⅱ微处理器为核心的应用系统），而传统的逻辑设计方法在一小时内很难达到50门逻辑电路的规模。

经过30多年的发展和演变，EDA技术在设计方法学、设计工具、集成电路制造工艺、目标应用等方面已取得丰硕成果。EDA设计既适用于计算密集型的系统（如视频和图像处理、超声成像、信号处理或基带处理等DSP应用系统），也适用于中断密集型的系统（如伺服电机控制、工业仪表、通信协议桥等控制、通信或消费电子应用系统）。图1-1是基于DSP和片上系统技术的电机伺服驱动器总体设计流程，包括算法建模及算法仿真、优化算法、系统集成、编译设计和系统软件设计、系统功能和性能验证等设计阶段。

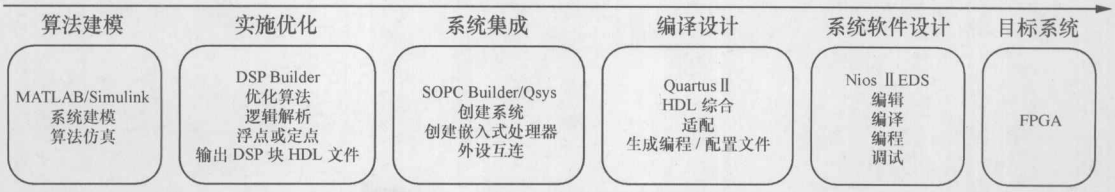


图 1-1 DSP 和片上系统技术的总体设计流程

不同的设计团队在设计工具的选择、设计风格偏好上都有所不同，但设计步骤是一致的：依据工程需求分析，制定总体实施方案并组织设计实施、设计优化、设计校验及工程管理所需的 EDA 设计工具链。

1.1.1 EDA 设计的技术优势

EDA 设计技术具有系统设计灵活性、系统自动集成、系统可重配置、采用硬件加速提高系统性能等技术优势。

1.1.1.1 系统设计灵活性

系统设计的实施载体一般分为两大类：专用集成电路（ASIC）和可编程逻辑器件（CPLD 和 FPGA）。ASIC 是专门为特定应用而设计的集成电路，其特点是种类繁多，面向特定用户需求，性能稳定，适合大批量、特定功能需求的应用，例如 Altera 公司的 MP3 解码器 AT85C51SND3、中国台湾 RAIO 公司的液晶驱动控制器 RA8803 等。但 ASIC 的研发成本高，固定的存储空间和数据通道、较窄的模拟范围、较低的系统扩展能力限制了 ASIC 系统的设计灵活性和适用性。

最早的可编程逻辑器件用于 ASIC 原型开发过程中的逻辑验证和性能测试。近十余年来，随 ASIC 产业结构的变化、EDA 设计技术的发展，可编程逻辑器件较之 ASIC 表现出迅猛的发展势头。由于可编程逻辑器件的可编程特性，可按系统设计需要增添用户定制逻辑模块、参数化模块、知识产权核、嵌入式硬核处理器、嵌入式软核处理器，利用 Qsys 或 SOPC Builder 等系统集成工具，指定系统互连，指定或自动分配处理器和组件的地址空间，指定中断向量地址、复位地址和异常处理地址，选择存储器管理或存储器保护功能，创建定制组件或定制指令，分配器件引脚、制定时序约束条件，通过上述一系列 EDA 设计方法和措施提高系统的设计灵活性。

1.1.1.2 系统设计自动化

EDA 设计的系统集成工具可按设计要求，利用交互式图形化用户接口或 Tcl 脚本命令，自动完成选定 EDA 设计流的设计输入、分析、综合、适配、编程和配置、调试等一系列过程，并实现设计规则检查、设计完整性检查、功能验证和时序验证、设计文件管理等。可设定第三方设计软件的运行条件，自动实现跨平台的综合、仿真、形式验证设计。EDA 设计使用如下工具链构成集成开发环境。

（1）设计输入编辑器。对块图/原理图输入文件、EDIF 网表文件、Qsys 系统文件、状态机文件、System Verilog HDL 文件、Tcl 脚本文件、Verilog HDL 文件、VHDL 文件进行编辑、管理。

（2）综合工具。分析、编译设计输入之后，输出网表文件的过程即综合过程，一般按照系统的抽象级别，分为门级综合、寄存器传输级综合（RTL 级综合）、行为级综合。综合

后可通过网表视图查看 RTL 综合结果，CPLD 或 FPGA 厂商提供的 EDA 工具嵌入逻辑分析与综合模块，例如 Altera 公司的 Quartus Ⅱ，也可采用专用的综合工具，例如 Synopsis 的 Synplify。

（3）仿真工具。利用测试平台（test bench）给定系统输入和激励信号，采集、分析系统输出响应的设计过程。一般分为 RTL 级仿真、门级仿真和时序仿真。RTL 级仿真对设计输入进行无延时仿真，以检查设计文件的语法错误和正确性，在综合之前进行，因此常称之为前仿真。未加入时序约束条件、使用综合工具综合之后生成的门级网表或门级模型进行仿真，以校验综合之后的功能是否满足设计要求，常称为门级仿真；布局/布线完成之后，在门级网表中加入时序标注文件进行仿真，称为时序仿真。采用 Quartus Ⅱ设计流程的网表写入器生成综合后功能仿真网表、适配后功能仿真网表和适配后时序仿真网表，利用仿真模块的测试平台生成激励输出，例化综合后功能仿真模块、适配后功能仿真模块和适配后时序仿真模块，仿真流程如图 1-2 所示。

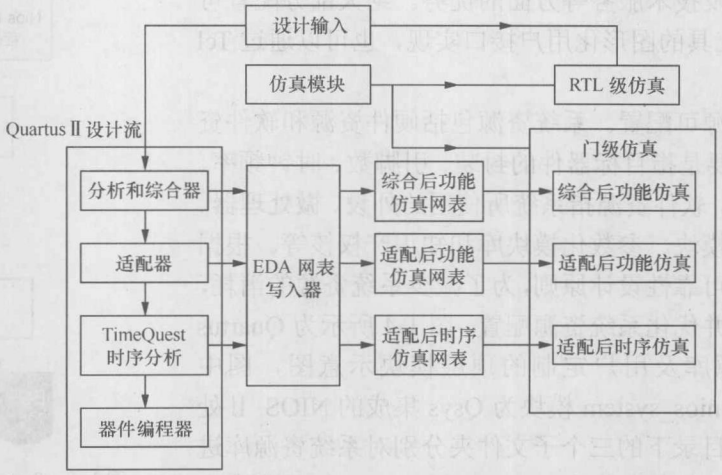


图 1-2 采用 Quartus Ⅱ设计流程的仿真设计

（4）适配工具。经分析和综合之后，对器件进行布局/布线的过程称为适配。利用器件布局布线器进行设计分区布局、器件布局，获得设计分区报告、时钟区域报告、引脚分配报告等。EDA 工具将适配工具集成在集成开发环境中，如 Quartus Ⅱ的 Chip Planner 和 Design Partition Planner。适配后输出形式验证文件（.vo），与设计输入文件一起进行形式验证。

（5）汇编工具。汇编器生成目标器件的编程数据文件或配置数据文件。

（6）编程/配置工具。利用 JTAG、计算机并行口或以太网协议接口，将编程或配置数据文件按激活串行（Active serial）、被动串行（Passive serial）、JTAG 模式等配置模式，利用诸如 Quartus Ⅱ的配置系统逻辑编程工具 Programmer 将配置数据或编程数据下载到目标器件中。

（7）系统调试工具。主要包括嵌入式逻辑分析仪、在系统资源和信号探针、存储器数据编辑器等工具，利用 JTAG 接口，将指定信号与系统设计一起进行布线、编程或适配，实时采集、监测调试对象。如 Quartus Ⅱ的 SignalTap Ⅱ、In-system Resources and Probes Editor 等。

（8）系统集成工具。利用系统集成工具的图形用户接口或 Tcl 脚本命令选择嵌入式处理器类型（如经济型、标准型或快速型 Nios Ⅱ通用处理器）、知识产权核或标准组件、用户定

义组件或定制指令，经互联总线进行连接，设置映射地址范围、复位地址和中断矢量，编辑组件的功能参数及结构参数，创建层次化结构的系统及系统仿真模型，输出系统信息文件（.sopcinfo）及片上系统文件（.ptf）。如 Quartus II 的 Qsys 或 SOPC Builder。

（9）软件工程开发工具。对于 Nios II 之类的软核处理器系统而言，软件开发工具流利用系统集成工具生成的系统信息文件（.sopcinfo），在软件开发工具（SBT）中创建电路板支持包（BSP）工程和应用工程，结合用户定制库或硬件抽象层应用可编程接口库（HAL API），按 makefile 定义的应用工程设置，编辑、编译应用工程，向应用系统的存储器输出可执行和链接文件（.elf）。其软件工程开发流如图 1-3 所示。

1.1.1.3 系统可重配置

采用 EDA 设计工具进行系统设计，很大程度上得益于系统可重配置特性，包括对系统资源、系统功能、系统性能的可重配置，使系统设计获得最大限度的设计自由度、灵活性、成本控制、管理及技术服务等方面的优势。绝大部分配置可通过 EDA 设计工具的图形化用户接口实现，也可以通过 Tcl 脚本命令实现。

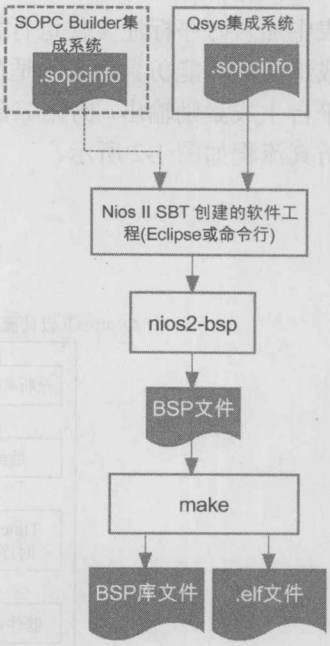


图 1-3 Qsys 或 SOPC Builder 的软件工程开发流

（1）系统资源可配置。系统资源包括硬件资源和软件资源，硬件资源主要是指目标器件的封装、引脚数、时钟频率、功率消耗等参数，软件资源指系统所使用的外设、微处理器、互联总线、定制模块、参数化模块库和知识产权核等。根据系统设计需求和可靠性设计原则，为了减少系统资源的消耗，应尽可能地权衡并优化系统资源配置。图 1-4 所示为 Quartus II 符号块中资源库及用户定制的顶层模块示意图，图中 project 文件夹下 nios_system 模块为 Qsys 集成的 NIOS II 处理器系统，安装目录下的三个子文件夹分别对系统资源库进行归类汇总：参数化模块库 magafunctions、分离模块库 others、原语模块 primitives，每一个文件夹又按模块或原语的属性进行细分，按用户授权情况开放资源使用权限。

（2）系统功能可配置。主要是指 Nios II 微处理器和知识产权核之类的功能配置，种类繁多。例如，Nios II 处理器的存储器管理功能或存储器保护功能，计数器模块的递增/递减计数、同步预置、异步复位等，与软件工程直接关联的 makefile 文件设置、BSP 编辑器的标签页设置等。图 1-5 所示为利用 Quartus II 的参数化模块库例化 Viterbi 译码算法模块的结构选项，包括混合结构、并行结构配置，节点同步配置，优化方式配置。

（3）系统性能可配置。影响系统性能的因素主要是微处理器的类型配置、存储器访问方式、是否采用硬件加速、布局布线设置、约束设计条件等。例如，经济型、标准型、快速型三种类型 Nios II 处理器内核消耗不同的逻辑资源，是影响系统性能的关键因素。或者，在同一个硬件平台上分别嵌入 ARM Cortex-A9、Nios II、DSP Builder+IP 核或状态机四种不同的微处理器或微控制器组件，将获得不同的中断延时、执行速度、数据量、确定性等系统性能，如图 1-6 所示。

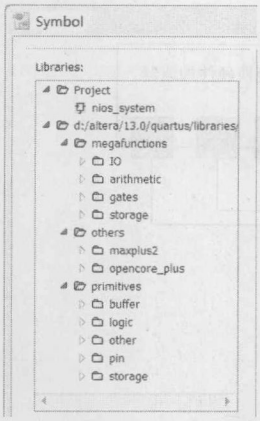


图 1-4 Quartus II符号块的资源库及用户定制资源

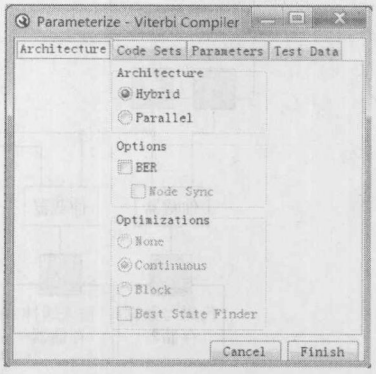


图 1-5 例化 Viterbi 译码算法模块

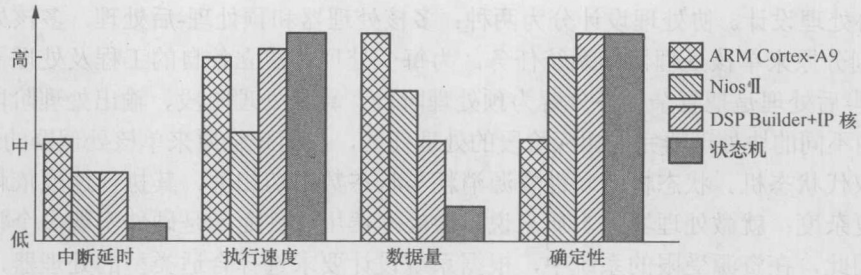


图 1-6 嵌入式微处理器/微控制器与系统性能的关系

1.1.1.4 硬件加速

硬件加速的目的是在性能受限制的 FPGA 系统平台上，实施可裁剪的硬件加速方法，使系统获得更高的执行效率。在选择更高性能的组件或模块、提高系统时钟频率可有效地实现硬件加速的同时，也增加了系统的功耗、成本和设计时间。CRC 组件、协处理设计、Nios II 定制指令、取代状态机、使用 C2H 编译器等可获得更高的硬件加速性能。

（1）采用 CRC 组件加速循环冗余校验检查。CRC 组件采用硬件模块代替处理器的循环冗余校验任务，减轻处理器的负担，提高处理器的执行速度和带宽。在系统中，Nios II 处理器利用 Avalon-MM 从端口 CRC 组件的寄存器进行读写控制访问，而 CRC 组件经 Avalon-MM 主端口访问存储器的数据，并进行 CRC 校验操作。CRC 任务从 Nios II 的软件中剥离出来，提高系统的硬件加速性能。此外，将片上存储器按照流水线结构布置、输入输出口的带宽与连接的相应组件带宽匹配，也可提高系统的硬件加速性能。

（2）使用 Nios II 定制指令。Nios II 定制指令是一种阻塞式运行、由用户自定义运算任务、与算术逻辑单元并行运行的组件，将原来 Nios II 的计算密集型运算任务从 Nios II 软件工程中分离出来，由定制指令单独完成（定制指令接口受微处理器控制），提高系统的硬件加速性能。

（3）选择 Nios II C2H 编译器。当 Nios II 的软件工程中频繁访问存储器时，选择 C2H 编译器而不是定制指令可明显提高系统的加速性能。C2H 编译器为每一个存储器创建一个 Avalon-MM 主端口，在系统集成工具中高速系统互联将硬件加速器、存储器连接，同时硬件加速器以从端口与 Nios II 的主端口连接，结果，硬件加速器作为高速互联总线上的外设组件，实现 Nios II 与存储器之间的高速数据访问和传输。结构如图 1-7 所示（以离散余弦变换硬件

加速器为例）。

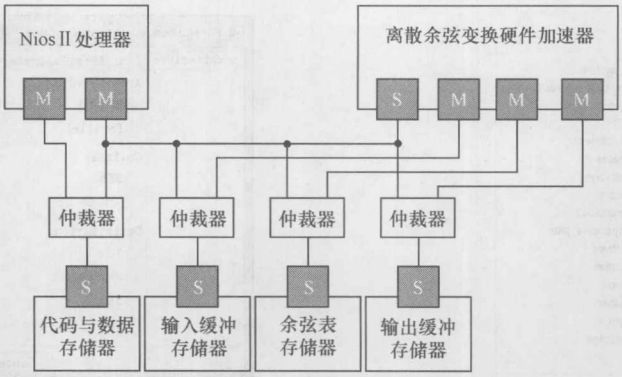


图 1-7　离散余弦变换硬件加速器结构

（4）协处理设计。协处理设计分为两种：多核处理器和预处理-后处理。多核处理器设计的目的是划分原来单核处理器的系统任务，为每个处理器建立各自的工程及处理器之间的通信。预处理-后处理是把复杂任务分解为预处理阶段、算法处理阶段、输出处理阶段等几个阶段，分别由不同的协处理器完成相应阶段的处理任务，从而减轻原来单核处理器的运算负荷。

（5）取代状态机。状态机的硬件资源消耗与状态数密切相关，其执行速度依赖于所实施的算法的复杂度；就微处理器的本质来说，微处理器用软件而不是硬件实现多个状态变换的状态机。因此，在资源受限的系统中，根据系统设计要求选择合适类型的处理器，使用软件模块或多核处理器协作，划分状态机的状态，代替状态机。对于低速状态机，可使用处理器、程序和数据存储器、激励信号接口、输出接口四种组件集成系统；对于高速状态机，可使用片上存储器和紧耦合存储器，以满足高速存储器访问和数据传输的需要，提高系统的性能。

1.1.2　EDA 设计流

EDA 设计流包括系统设计需求分析、制定设计方案、组织设计工具链、设计实践等环节。

（1）系统设计需求分析。包括系统的实时性能要求、计算性能和功能要求、工程设计周期、成本预算、分析技术风险并制定相应的技术保障措施，制定工程实施进度、预期目标等。

（2）制定设计方案并组织设计工具链。要求分解工程设计任务，估算系统的软硬件资源规模、功率消耗、时序要求，按自身和团队的专业技术特长、设计资源和开发工具平台，选择合适的器件类型、软硬件集成开发工具，必要时采用第三方提供的综合、仿真或形式验证工具，构建完备的设计工具链。

（3）设计实践。这是 EDA 设计实现的核心部分，由设计输入、分析与综合、适配、时序分析、汇编、编程或配置、调试等阶段组成，各阶段可以按照渐进式编译设计流程、增量式编译设计流程或智能编译设计流程进行，图 1-8 所示为采用增量式编译的设计流程。

在设计实践过程中，为了保证设计输出满足系统设计的功能要求和时序要求，必须进行功能仿真（包括 RTL 级仿真和门级仿真）和时序仿真，而且使用形式证明的数值方法验证设计功能是否与设计要求等价、检查形式模型是否一致，即进行相应的等价性形式验证或形式模型检查。对于使用 Quartus II 开发环境来说，就是验证被综合的门级 VerilogQuartus 映射文件（.vqm）与 Quartus II 生成的 Verilog 输出文件（.v）之间的逻辑等价性。

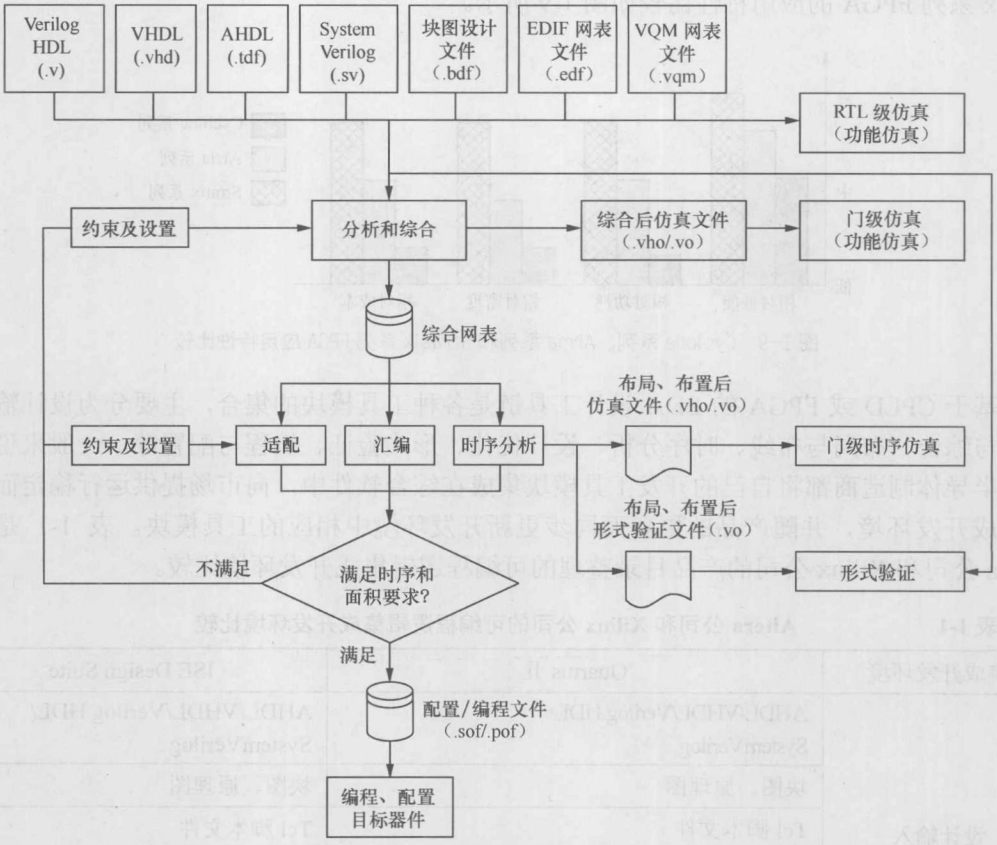


图 1-8 增量式编译设计流程

在仿真和形式验证过程中，若系统的设计输出或调试结果不满足系统的时序和面积要求，则调整设计约束条件，修改 EDA 工具设置选项，返回到相应的设计输入、综合、适配等阶段，再次进行设计实践，直到满足系统的时序和面积要求为止。

1.2 可编程逻辑器件及其开发环境

可编程逻辑器件是可编程逻辑阵列 PLA/GAL、复杂可编程逻辑器件 CPLD、现场可编程门阵列 FPGA 的统称，是由 PLA、GAL 等器件逐步发展起来的超大规模集成电路。时至今日，可编程逻辑器件的主要生产商，如 Lattice、Xilinx、Altera 公司等，都以市场需求为导向，以不同的方法设计制造通用可编程逻辑器件，CPLD 如 Lattice 公司的 ispMACH 4000ZE 系列，Xilinx 公司的 CoolRunner-Ⅱ系列，Altera 公司的 MAX Ⅱ/3000A/7000AE 系列；FPGA 如 Altera 公司的 Cyclone 系列、Arria 系列、Stratix 系列，Lattice 公司的 LatticeECP2/3/4/SC 系列、LatticeXP2、iCE40、MachXO2、MachXO 系列，Xilinx 公司的 7 系列、Virtex-4/5/6、Spartan-6、Spartan-3 系列等。

可编程逻辑器件的性能、密度、功耗、成本和应用范围各不相同。例如，Altera 公司的 Cyclone 系列 FPGA，适合低功耗、低成本应用；Arria 系列适合高性能计算；而 Stratix 系列具有最优性能和最大逻辑密度，适于应用处理系统或原型开发。Cyclone 系列、Arria 系列、

Stratix 系列 FPGA 的应用特性比较如图 1-9 所示。

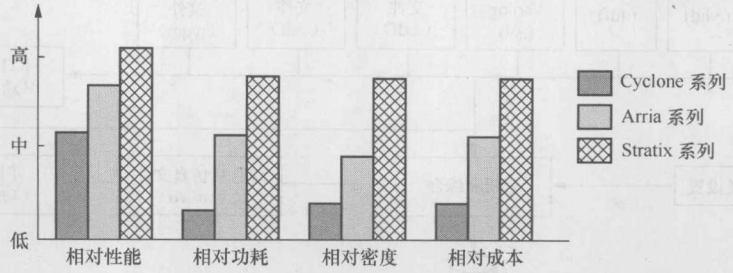


图 1-9 Cyclone 系列、Arria 系列和 Stratix 系列 FPGA 应用特性比较

基于 CPLD 或 FPGA 的 EDA 设计工具链是各种工具模块的集合，主要分为设计输入、分析与综合、布局与布线、时序分析、设计优化、形式验证、编程与配置等。一般来说，大部分半导体制造商都将自己的开发工具模块集成在综合软件中，向市场提供运行稳定而高效的集成开发环境，并随产品的更新而同步更新开发环境中相应的工具模块。表 1-1 是根据 Altera 公司和 Xilinx 公司的产品目录整理的可编程逻辑集成开发环境比较。

表 1-1　　Altera 公司和 Xilinx 公司的可编程逻辑集成开发环境比较

集成开发环境	Quartus II	ISE Design Suite
设计输入	AHDL/VHDL/Verilog HDL/ SystemVerilog	AHDL/VHDL/Verilog HDL/ SystemVerilog
	块图、原理图	块图、原理图
	Tcl 脚本文件	Tcl 脚本文件
	状态机文件	状态机文件
	系统集成信息文件	系统集成信息文件
	设计约束文件	设计约束文件
功能仿真	ModelSim-Altera	ISE Simulator
系统集成	Qsys 或 SOPC Builder	
DSP 系统创建	DSP Builder	System Generator for DSP
分析与综合	Integrated Synthesis	PlanAhead
布局/布线	Chip Planner	ISE Design Suite
	Deisn Partition Planner	
时序分析	TimeQuest Timing Analyzer	ISE Design Suite
功耗分析	PowerPlay	
系统调试	SignalTap II Logic Analyzer	ChipScope Pro Logic Analyzer
	Tansceiver Toolkit	Pro Serial I/O Toolkit
	In-System Resources and Probes Editor	
网表写入	Netlist Writer	ISE Design Suite
汇编	Assembler	ISE Design Suite
编程	Programmer	ISE Design Suite
嵌入式设计	Nios II EDS	Embeded Development Kit