

EDA技术及其嵌入式系统设计

袁海林 著

EDA

中央民族大学出版社

湖北民族学院学术专著出版基金资助

EDA 技术及其嵌入式系统设计

袁海林 著

中央民族大学出版社

图书在版编目(CIP)数据

EDA 技术及其嵌入式系统设计/袁海林著. —北京:中央民族大学出版社,2007.8

ISBN 978-7-81108-373-6

I. E… II. 袁… III. ①电子电路—电路设计:计算机辅助设计②微型计算机—系统设计 IV. TN702 TP360.21

中国版本图书馆 CIP 数据核字(2007)第 078114 号

EDA 技术及其嵌入式系统设计

作 者 袁海林

责任编辑 李苏幸

封面设计 姚俊男

出 版 者 中央民族大学出版社

北京市海淀区中关村南大街 27 号 邮编:100081

电话:68472815(发行部) 传真:68932751(发行部)

68932218(总编室) 68932447(办公室)

发 行 者 全国各地新华书店

印 刷 者 湖北民族学院印刷厂

开 本 787 × 1092(毫米) 1/16 印张:14

字 数 250 千字

版 次 2007 年 8 月第 1 版 2007 年 8 月第 1 次印刷

书 号 ISBN 978-7-81108-373-6

定 价 20.00 元

版权所有 翻印必究

EDA 技术及其嵌入式系统设计

内 容 简 介

本书分两个部分对 EDA 及其嵌入式系统设计技术进行了全面、系统地论述。第 1~5 章为基础部分,主要分析可编程逻辑器件硬件结构、VHDL 语言、MAX + PLUSII 和 Quartus II 软件的使用方法和基本数字系统设计实例;第 6~7 章为提高部分,主要论述了 SOPC 设计方法、NiosII CPU 的使用技巧、SOPC 的嵌入式系统开发流程和基于 NiosII 的嵌入式系统设计实例。

本书适用于从事 EDA 和 SOPC 技术开发的研究人员、相关专业的研究生、本科生,以及对该领域感兴趣的爱好者。

序 言

EDA 及其嵌入式系统设计技术是目前电子设计领域日益广泛应用的新的设计技术。它是一门综合性的学科,涉及传统软硬件设计技术的方方面面,同时还融入了多任务实时操作系统的新内容。之前,嵌入式系统开发一般采用嵌入式处理器加 ASIC 器件的方式,而 EDA 技术在很多时候则是作为嵌入式处理器的外部模块电路的实现技术。随着可编程逻辑器件工艺和开发工具功能的迅速提高,可编程逻辑器件在嵌入式系统设计中的角色发生了巨大的变化。现在,我们可以在一个可编程逻辑器件上实现整个嵌入式系统的数字部分的设计。这就是 SOPC(System on a Programmable Chip)的概念,即所谓的可编程芯片系统。并且,FPGA 在复杂逻辑电路以及数字信号处理领域中扮演着越来越重要的角色,SOPC 正以其低功耗、高性能、低成本、高灵活性、高可靠性等优点成为嵌入式系统的发展趋势。

本书分两个部分对 EDA 及其嵌入式系统设计技术进行了全面、系统地论述。第 1~5 章为基础部分,主要分析可编程逻辑器件硬件结构、VHDL 语言、MAX + PLUSII 和 Quartus II 软件的使用方法和基本数字系统设计实例;第 6~7 章为提高部分,主要论述了 SOPC 设计方法、NiosII CPU 的使用技巧、SOPC 的嵌入式系统开发流程和基于 NiosII 的嵌入式系统设计实例。章节的具体安排如下:

第一章 简要介绍 EDA 技术的概念,并对 EDA 技术所涉及的几个关键技术——PLD、HDL 和 SOPC 等进行了介绍。

第二章 介绍 PLD 的结构及其 FPGA 和 CPLD 的特点

第三章 介绍 VHDL 的语法

第四章 介绍 VHDL 的设计实例

第五章 介绍 EDA 工具软件 QuartusII 和 MAX + P LUSII 的基本使用方法

第六章 介绍 SOPC 的嵌入式系统设计的概念和 NiosII 的特点

第七章 通过一个设计实例,介绍基于 NiosII 软 CPU 核的 SOPC 设计流程,详细介绍使用 QuartusII、SOPC Builder 和 NiosII IDE 进行嵌入式系统的软、硬件设计的流程和方法。

本书在编写中引用了诸多学者和专家的著作和论文中的研究成果,在这里向他们表示衷心的感谢和敬意。由于本人水平有限,错漏和不严谨之处在所难免,欢迎大家批评指正。

目 录

第一章 EDA 及其嵌入式系统概述	1
1.1 EDA 技术的发展概况	2
1.1.1 CAD 阶段	2
1.1.2 CAE 阶段	2
1.1.3 EDA 阶段	2
1.2 PLD 概况	3
1.3 VHDL 概述	4
1.4 EDA 开发工具	5
1.5 SOPC 概述	6
第二章 可编程逻辑器件	8
2.1 PLD 器件	8
2.1.1 可编程逻辑器件的分类及特点	8
2.1.2 可编程阵列逻辑器件的结构	10
2.1.3 可编程通用逻辑器件的结构	12
2.2 复杂可编程逻辑器件	15
2.3 现场可编程门阵列 FPGA	19
2.3.1 FPGA 的基本结构	19
2.3.2 FPGA 的电路结构	22
第三章 VHDL 基础	29
3.1 VHDL 的基本构件	29
3.1.1 实体	29
3.1.2 结构体	32
3.1.3 库、程序包和配置	33
3.2 VHDL 的数据对象和数据类型	36

3.2.1	VHDL 的数据对象	36
3.2.2	VHDL 的数据类型	37
3.3	VHDL 语言的操作符和表达式	38
3.3.1	逻辑操作符和逻辑表达式	38
3.3.2	算数操作符和算数表达式	39
3.3.3	关系操作符和关系表达式	40
3.3.4	并置操作符和并置表达式	40
3.4	VHDL 基本语句	40
3.4.1	顺序描述语句	41
3.4.2	并行语句	48
3.5	VHDL 语言的描述风格	61
3.5.1	行为描述	62
3.5.2	数据流描述	63
3.5.3	结构描述	64
第四章	VHDL 设计实例	67
4.1	组合逻辑电路	67
4.2	时序电路设计	75
4.3	综合电路设计	97
第五章	开发工具	112
5.1	MAX + PLUSII 的使用	112
5.1.1	MAX + PLUSII 的系统安装	113
5.1.2	设计输入	114
5.1.3	新立一个项目	117
5.1.4	编译	117
5.1.5	仿真	118
5.1.6	分配管脚	124
5.1.7	下载	125
5.1.8	MAX + PLUS II 的图形编辑器的使用	126

5.1.9	MAX + PLUS II 的宏模块及其应用	130
5.1.10	Flex10k 中 RAM 的使用	132
5.2	QUARTUSII 的使用	135
5.2.1	QUARTUSII 的系统安装	136
5.2.2	创建工程	147
5.2.2	建立工作库文件夹和编辑设计文件	152
5.2.3	引脚设置和下载	157
第六章	SOPC 设计基础	167
6.1	SOPC 的系统方案	167
6.2	NiosII 软核处理器	169
6.2.1	NiosII 软核处理器简介	169
6.2.2	Avalon 总线简介	170
6.2.3	外围接口	172
6.3	SOPC 的开发流程	172
6.3.1	硬件开发	174
6.3.2	软件开发	174
6.3.3	SOPC 基本开发流程简介	175
第七章	NiosII 嵌入式系统开发实例	179
7.1	分析系统需求	179
7.2	使用 Quartus II 建立工程	181
7.2.1	启动 Quartus II 软件并建立工程	181
7.2.2	建立顶层文件	184
7.3	使用 SOPC Builder 创建 NiosII 系统	185
7.3.1	SOPC Builder 菜单命令介绍	185
7.3.2	启动 SOPC Builder	186
7.3.3	指定目标 FPGA 和时钟设置	187
7.3.4	添加 NiosII 处理器内核	188
7.3.5	添加 PIO	191

7.3.6	添加片内存储器	193
7.3.7	添加 SDRAM	194
7.3.8	指定基础地址和中断请求优先级	195
7.3.9	设置 NiosII 复位和异常地址	196
7.3.10	生成 NiosII 系统	197
7.4	集成 NiosII 系统到 QuartusII 工程	198
7.4.1	添加 NiosII 系统模块到 Quartus II 顶层模块	198
7.4.2	添加引脚和其它基本单元	199
7.4.3	Analysis 和 Elaboration 编译	200
7.4.4	选择器件型号和管脚分配	201
7.4.5	分配 FPGA 管脚本	202
7.5	设置编译选项并编译硬件系统	202
7.5.1	设置编译选项	202
7.5.2	编译硬件系统	203
7.5.3	查看编译报告	204
7.6	下载硬件设计到目标 FPGA	204
7.7	使用 NiosII IDE 建立用户程序	205
7.7.1	创建一个新的 C/C++ 应用工程	205
7.7.2	设置 C/C++ 应用工程系统属性	210
7.7.3	编译连接工程	210
7.8	在目标硬件上运行程序	211
7.8.1	在目标硬件上运行程序	211
7.8.2	在目标板上调试程序	212
参考文献		214

第 1 章

EDA 及其嵌入式系统概述

电子设计自动化(EDA, Electronic Design Automation)技术是一种以计算机作为工作平台,以 EDA 软件工具为开发环境,以硬件描述语言和原理图描述为设计入口,以可编程逻辑器件为实验载体,以 ASIC(Application Specific Integrated Circuits)、SOC(System On Chip)和 SOPC(System On Programmable Chip)嵌入式系统为设计目标,以数字系统设计为应用方向的电子产品自动化设计技术。它是融合了电子技术、计算机技术、信息处理技术、智能化技术等最新成果而开发的高新技术,是现代电子系统设计、制造不可缺少的技术。EDA 技术涉及面广,包含描述语言、软件、硬件等多方面知识。

EDA 技术代表了当今电子设计技术的最新发展方向,它是电子设计领域的一场革命。它的基本特征是:设计人员按照“自顶向下”的设计方法,对整个系统进行方案设计和功能划分,然后采用硬件描述语言(HDL)完成系统行为级设计,最后通过综合和适配生成最终的目标器件。这样的设计方法称为高层次的电子设计方法。

本章首先介绍 EDA 技术的发展概况,再对 EDA 技术所涉及的 PLD(Programmable Logic Device)、VHDL(Very High Speed Integrated Circuits Hardware Description Language)、EDA 工具等三个要素进行简要介绍、最后对 EDA 的嵌入式

系统设计技术——SOPC 技术进行介绍。目的是对本书的内容进行一个概要性的描述,以起到明确本书的知识体系的作用。

1.1 EDA 技术的发展概况

EDA 技术发展迅猛,可以用日新月异来形容。从七十年代以来,在计算机技术和集成电路制造技术的推动下,EDA 技术水平不断提高,使其应用越来越广泛,现在已涉及到各行各业。EDA 技术发展至今,经历了三个阶段。

1.1.1 CAD 阶段

(七十年代)电子线路的计算机辅助设计(CAD, Computer Aided Design)是 EDA 发展的初级阶段。它利用计算机的图形编辑、分析和存储等能力,协助工程师进行电子系统的 IC(Integrated Circuit)版图编辑和 PCB(Printed Circuit Board)布局布线,取代了手工操作。它可以减少设计人员的繁琐重复劳动,但自动化程度低,需要人工干预整个设计过程。这类专用软件大多以微机为工作平台,易于学习和使用,设计中小规模电子系统可靠有效。现仍有很多这类专用软件被广泛应用于工程设计。由此产生了计算机辅助设计的概念。

1.1.2 CAE 阶段

(八十年代)与 CAD 相比,除了纯粹的图形绘制功能外,该阶段已具备了设计自动化的功能。其主要特征是具备了自动布局布线和电路的计算机仿真、分析和验证功能。又增加了电路功能设计和结构设计,并且通过电气连接网络表将两者结合在一起,以实现工程设计,这就是计算机辅助工程的概念 CAE(Computer Aided Engineering)。其作用已不仅仅是辅助设计,而且可以代替人进行某种思维。CAE 的主要功能有:原理图输入,逻辑仿真,电路分析,自动布局布线,PCB 后分析。

1.1.3 EDA 阶段

(九十年代至今)高级 EDA 阶段,又称为电子系统设计自动化(ESDA, Electronic System Design Automation)。过去传统的电子系统产品的设计方法是采用“自底而上”的顺序。依照这种顺序,设计者先对系统结构分块,直接进行电路级的设计。这种设计方式使设计者不能预测下一阶段的问题,而且每一阶段是否存在问题,往往在系统整机调试时才确定,也很难通过局部电路的调整使整个

系统达到既定的功能和指标,不能保证设计一举成功。EDA 技术高级阶段采用一种新的“自顶向下”的设计顺序和“并行工程”的设计方法。依照这种方法,设计者的精力主要集中在所要设计系统的准确定义上,EDA 系统去完成电子产品的系统级到物理级的设计。该阶段 EDA 技术的基本特征是,设计人员以计算机为工具,按照“自顶向下”的设计方法,对整个系统进行方案设计和功能划分。设计人员再将划分后的各个模块用硬件描述语言等设计描述方法完成系统行为级设计,再利用先进的开发工具自动完成逻辑编译、化简、分割、综合、优化、布局布线、仿真及特定目标芯片的适配编译和编程下载,这也称为数字逻辑电路的高层次设计方法。

1.2 PLD 概况

可编程逻辑器件(PLD, Programmable Logic Device)是集成电路的一个重要分支。它是 20 世纪 70 年代发展起来的一种新型的集成电路制造技术。它是直接支持 EDA 技术发展的物理基础。

可编程逻辑器件的逻辑功能可利用 EDA 技术完全由用户根据需要,通过对器件的编程设计来实现。可编程逻辑器件的这种设计方法是由集成电路的工艺所支持的。如图 1.1 所示,在集成电路系列中按照制造工艺的不同可分为全定制和半定制两种。可编程逻辑器件属于半定制制造工艺中的一个子类。

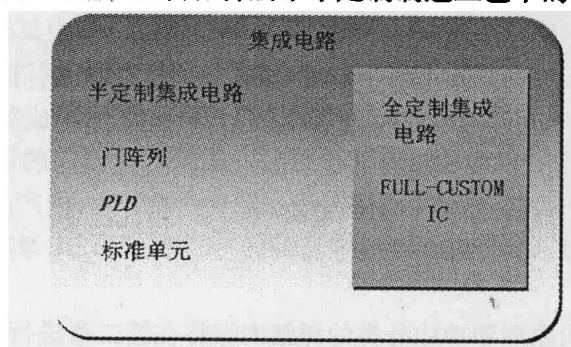


图 1.1 集成电路按照制造工艺分类的示意图

按照全定制工艺设计制造集成电路芯片时,设计师要根据芯片要实现的电路特性,定义芯片上所有晶体管的几何图形和工艺规则,最后将设计结果交由 IC 厂家掩膜制造完成。全定制制造工艺的优点是芯片可以获得最优的性能、面积利用率高、速度快、功耗低;缺点是开发周期长、费用高、只适合大批量产品开发。

与全定制工艺不同,按照半定制工艺集成电路芯片的计方法,IC 制造厂家在接到制造任务之前就已经以模块的形式完成了初期的布局工序。当接到制造任务时,IC 制造厂家的主要工作是根据芯片要实现的电路对前述的半成品芯片进行布线和掩模,最终完成全部制造工序。这种方法的主要目的是简化设计、缩短开发制造周期。

在半定制工艺中,按照半成品芯片的模块的构成形式和是否封装,又可以分为门阵列、标准单元和可编程逻辑等三种设计方法。门阵列、标准单元的半定制工艺设计方法是 IC 制造厂家以门阵列、标准单元的模块的形式完成初期的布局工序,并将这种半成品芯片保存在工厂以备进一步的设计所用。

可编程逻辑芯片与门阵列、标准单元的半定制工艺设计方法不同之处在于,IC 制造厂家以标准单元的模块的形式完成初期的布局和可编程布线等工序。更大的区别在于可编程逻辑芯片是将这种“待完成工序”的芯片进行封装,以成品的形式进入市场,供设计者根据自己的设计需要进行“再开发”。在 PLD 先进的工艺的支持下,设计人员完成版图设计后,在实验室内就可以烧制出自己的芯片,无须 IC 厂家的参与。它不仅具有设计灵活、性能高、速度快等优势,而且开发周期短、成本低廉。在半导体领域中随着设计技术和制造工艺的完善,器件性能、集成度、工作频率等指标不断提升,PLD 应用日益普及,已成为集成电路中最具活力和前途的产业,已越来越多地成为系统级芯片设计的首选。

可编程逻辑器件自七十年代以来,经历了 PAL(Programmable Array Logic)、GAL(Generic Array Logic)、CPLD、FPGA 几个发展阶段。其中复杂可编程逻辑器件(CPLD, Complex Programmable Logic Device)和场可编程门阵列(FPGA, Field Programmable Gate Array)属高密度可编程逻辑器件,目前集成度已高达几百万门/片,它将掩膜 ASIC 的集成度高的优点和可编程逻辑器件的设计生产方便的特点结合在一起,特别适合于样品研制或小批量产品开发,使产品能以最快的速度上市,而当市场扩大时,它可以很容易的转变成掩膜 ASIC 实现,因此开发风险也大为降低。

关于 PLD 结构原理和功能分类的详细内容将在第二章进行介绍。

1.3 VHDL 概述

硬件描述语言(HDL, Hardware Description Language)是一种用于设计硬件电子系统的计算机语言。它是 EDA 技术的重要组成部分,是现代 EDA 技术的发展在设计输入方式上的一个重要标志。它用软件编程的方式来描述电子系统

的逻辑功能、电路结构和连接形式。与传统的门级描述方式相比,它更适合大规模系统的设计。例如一个 32 位的加法器,利用图形输入软件需要输入 500 至 1000 个门,而利用 HDL 只需要书写一行 $A = B + C$ 即可,而且 HDL 语言可读性强,易于修改和发现错误。早期的硬件描述语言,如 ABEL-HDL、AHDL 是由不同的 EDA 厂商开发的,互相不兼容,而且不支持多层次设计,层次间翻译工作要由人工完成。为了克服以上缺陷,1985 年美国国防部正式推出了 VHDL(Very High Speed IC Hardware Description Language)语言,1987 年电气和电子工程师协会(IEEE, Institute of Electrical and Electronics Engineers)采纳 VHDL 为硬件描述语言标准(IEEE STD-1076)。它是目前使用最普遍的一种硬件描述语言之一。

VHDL 是一种全方位的硬件描述语言,包括系统行为级、寄存器传输级和逻辑门级等多个设计层次,支持结构、数据流、行为等三种描述风格以及它们的混合描述。因此 VHDL 几乎覆盖了以往各种硬件描述语言的功能。整个自顶向下或自底向上的电路设计过程都可以用 VHDL 来完成。另外,VHDL 还具有以下优点:

- VHDL 的宽范围描述能力使它成为高层次设计的核心,将设计人员的工作重心提高到了系统功能的实现与调试,只需花较少的精力用于物理实现。
- VHDL 可以用简洁明确的代码描述来进行复杂控制逻辑的设计,灵活且方便,而且也便于设计结果的交流、保存和重用。
- VHDL 的设计不依赖于特定的器件,方便了工艺的转换。VHDL 是一个标准语言,为众多的 EDA 厂商支持,因此移植性好。

关于 VHDL 的语法规则和设计实例的详细内容将在第三、四章进行介绍。

1.4 EDA 开发工具

EDA 开发工具在 EDA 技术中具有及其重要的作用。因为 EDA 技术的重要特点是利用了计算机的强大和高速度的数据处理能力来完成电子系统的设计。因此,基于计算机硬、软件平台基础之上的一系列 EDA 软件工具在实现 EDA 技术的电子系统设计时,是必不可少的开发环境支持。

根据 EDA 技术的功能要求,EDA 工具应具有如下五个模块:

- 设计输入编辑器:接受包括硬件描述语言、原理图等多种形式的输入。
- 综合器:对设计输入文件进行逻辑化简、综合、优化和适配,最后生成编程用的编程文件。

- 仿真器:将设计描述转换成一个与之对应的软件模型,供设计者在电脑上对其设计进行仿真和定时分析。其目的是检验设计描述的逻辑功能是否正确,同时测试目标器件在最差情况下的时间关系等。

- 适配器:将由综合器产生的网表文件配置于指定的目标器件中,产生最终的下载文件。

- 下载器:将适配后产生的下载文件下载到相应 CPLD 或 FPGA 等可编程逻辑器件中,使其成为一个具有设计功能的专用集成芯片。

本书介绍的 ALTERA 公司的 MAX + PLUSII 和 QuartusII 这两个工具软件是将上述功能集成为一体的集成软件。它们是 ALTERA 公司先后推出的两个 EDA 工具软件。MAX + PLUSII 是 ALTERA 公司的一个早期的软件。它的特点是系统规模适中、在性能较弱的 PC 机上也很好地运行、可适配的是一些早期的 PLD 芯片系列。该软件适合受条件限制的个人和实验室在初期进行 EDA 技术学习和设计时使用。Quartus II 是 ALTERA 公司开发的一款继 MAX + PLUSII 之后的又一个功能更强大的集成化的 EDA 工具软件,它在具有 MAX + PLUSII 全部功能的同时,可适配的 PLD 芯片系列更多,功能更强大。它是 EDA 技术的研究、开发设计时的首选设计工具。在 Quartus II 中集成了一个自动化的系统开发工具 SOPC Builder。它支持 SOPC 嵌入式系统的全部的硬件设计,并提供了一个强大的设计平台以搭建基于总线的系统,其内部包含了一系列的模块,如处理器、存储器、总线、DSP (Digital Signal Processing) 等 IP 核 (Intellectual Property core) 等。使用 SOPC Builder,设计人员能够快速调用和集成内建的 IP 核库,定义一个完整的硬件系统。

支持系统的软件设计的工具有 ALTERA 公司开发的 NiosII IDE。该软件用于支持基于 NiosII 系统的软件开发和调试。关于 MAX + PLUSII 和 QuartusII 这两个工具软件的基本使用方法将在第五章进行介绍。关于 QuartusII 所支持的 SOPC 的硬件设计和 NiosII IDE 的使用方法将在第六、七章进行介绍。

1.5 SOPC 概述

SOPC(System On Programmable Chip)技术是 SOC(System On Chip)技术和电子设计自动化(EDA)技术结合的产物。它可以将处理器、存储器、I/O 接口、硬件协处理器和普通的用户逻辑等系统设计需要的功能模块都集成到一个 FPGA 芯片里,构建一个可编程的片上系统。它还具有灵活的设计方式,可裁减、可扩充、可升级,具备系统可编程等功能,是一种优秀的嵌入式系统设计技术。

SOPC 设计技术实际上涵盖了嵌入式系统设计技术的全部内容,除了以处理器和实时多任务操作系统(RTOS, Real-Time Operating System)为中心的软件设计技术、以 PCB 和信号完整性分析为基础的高速电路设计技术以外,SOPC 还涉及目前已引起普遍关注的软硬件协同设计技术。

SOPC 结合了 SOC 和 FPGA 各自的优点,具备以下基本特征:

- 至少包含一个以上的嵌入式处理器 IP Core
- 具有小容量片内高速 RAM 资源
- 丰富的 IP Core 资源可供灵活选择
- 足够的片上可编程逻辑资源
- 处理器调试接口和 FPGA 编程接口共用或并存
- 可能包含部分可编程模拟电路
- 单芯片、低功耗、微封装

本书将在第六、七章主要论述 SOPC 设计方法、NiosII CPU 的使用技巧、SOPC 的嵌入式系统开发流程和基于 NiosII 的嵌入式系统设计实例等内容。