



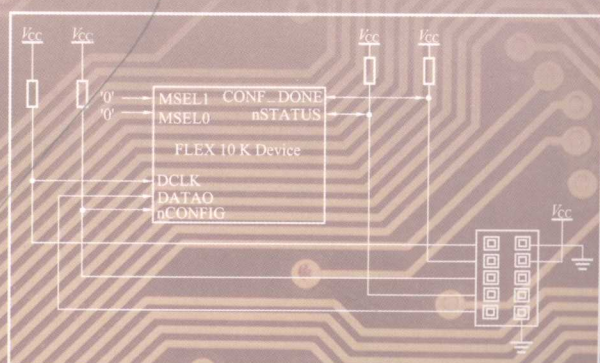
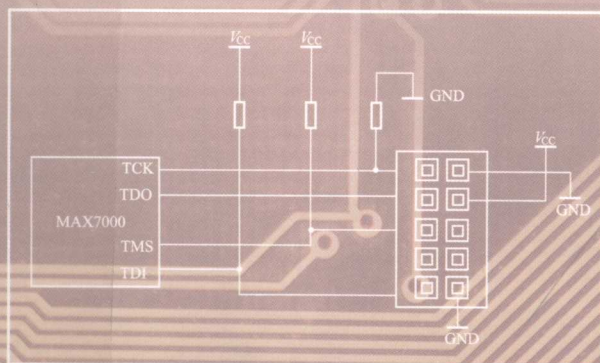
电气工程与电气信息科学技术工程系列丛书

可编程 逻辑器件应用实践

主编 杨春玲 朱 敏

eBianCheng

LuoJi QiJian YingYong ShiJian



哈尔滨工业大学出版社

TP332.1/59

2008

电气工程与电气信息科学技术工程系列丛书

内容简介

本书以高电平触发式静态 CMOS 器件为对象, 介绍其设计方法。全书共分 6 章, 第 1 章介绍 CMOS 器件的基本原理, 第 2 章介绍 CMOS 器件的静态特性, 第 3 章介绍 CMOS 器件的动态特性, 第 4 章介绍 CMOS 器件的噪声特性, 第 5 章介绍 CMOS 器件的可靠性, 第 6 章介绍 CMOS 器件的应用。本书可作为电气工程及其自动化专业及相关专业的教材, 也可供从事 CMOS 器件设计工作的工程技术人员参考。

可编程逻辑器件应用实践

主编 杨春玲 朱 敏

图在版编目(CIP)数据

可编程逻辑器件应用实践 / 杨春玲, 朱敏主编. — 2008. 7

哈尔滨工业大学出版社, 2008. 7

(哈尔滨工业大学电气工程及其自动化专业本科教材)

ISBN 978-7-5603-2303-7

I. ①可… II. 杨… III. 可编程逻辑器件 IV. TP332.1

中国版本图书馆 CIP 数据核字(2008)第 069328 号

责任编辑 王桂芝 贾学斌

出版发行 哈尔滨工业大学出版社

社址 哈尔滨市南岗区复华四道街 10 号 邮编 150006

传真 0451-86414743

网址 <http://publishing.int.hjtu.edu.cn>

印刷 哈尔滨工业大学印刷厂

本 787mm×1092mm

文 2008 年 7 月第 1 版

号 ISBN 978-7-5603-2303-7

价 25.00 元

哈尔滨工业大学出版社

(如因印刷质量问题影响阅读, 恕不退换)

本书既可作为高等学校机电一体化、电子工程、通信、工业自动化、计算机应用技术、电子对抗、仪器仪表、数字信号或图像处理等学科的 EDA 技术教材及实验指导书,亦可作为相关专业技术人员的参考书。

主 编 李 金 林

(电气工程与电气信息科学技术工程系列丛书)

I .可… II .杨… III . 可编程逻辑器件 IV .TP332.1

责任编辑 王桂芝 贾学斌

社 址 哈尔滨市南岗区复华四道街 10 号 邮编 150006

网 址 <http://hitpress.hit.edu.cn>

开 本 787mm×1092mm 1/16 印张 12.75 字数 330 千字

书 号 ISBN 978-7-5603-2703-7

定 价 25.00 元

◎ 序

Foreword

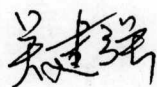
随着经济全球化、产业国际竞争的加剧和电子信息科学技术的飞速发展,电气工程与电气信息科学技术领域的国际交流日益广泛,因此,对能够参与国际化工程项目的工程师的需求愈来愈迫切,这便对高等学校电气工程与电气信息科学技术领域专业人才的培养提出了更高的要求。

近些年,国家教育部对本科生教育提出了“厚基础、宽口径”的要求,使学生掌握了比较扎实的基础知识,拓宽了学生的就业方向和就业广度。但与此同时,也显露出刚毕业的大学生工程实践能力差、难以很快适应工作的问题,对于电类专业的学生来说,缺少工程教育的过程,很多工程实际操作、实用技术,因受学时限制,不能讲深、讲透,出现了“理论基础扎实、工程实践能力欠缺”的问题;而对于那些在校时只接触过类似“电工学”这样的电类基础课程,而工作后却从事电气领域相关工作的非电类专业人员来说,这种问题就显得更为突出。目前,教育部已经认识到并着手解决这方面的问题,开始在工科高等学校开展**工程教育专业认证**工作,积极推进工程教育改革,以提高学生的工程实践能力和创新能力,培养能够参与国际化工程项目的工程师,在实质等效性的要求下推进全球工程教育的交流。

为了更好地配合高等学校的工程教育改革,我们组织编写了《电气工程与电气信息科学技术工程系列丛书》。该丛书侧重介绍当代电气工程与电气信息科学技术领域的主要知识和应用技术,重点讲述工程实践中的一些具体实例,以使这些学生能够尽快了解该领域内的新知识和新技术,领悟工程概念,提高工程实践能力,使其工作后能够尽快进入角色。该丛书的编写原则是理论上“以必须和够用为度”、“重点突出”;实例选择上“以工程实践为基础”、“实用性强”。

该丛书适合于**电类专业**的在校本科生,使其在拥有较扎实的理论基础上,加强工程实践教学,较快地了解和掌握工程实践中的一些实际应用技术;也适合于毕业后从事电气领域相关工作的**非电类专业**学生,使其能够通过该丛书系统地了解该领域的主要知识和实际应用技术,尽快进入工作角色。由于其简练的理论阐述和较强的实用特性,该丛书也可以作为**高职高专**类相关专业的教材。

该丛书作者队伍阵容强大,既有国内电工学教育界的知名学者,也有哈尔滨工业大学电气工程领域内从事多年教学和科研工作的教授、博导。他们将近年来该领域的新成果和多年来的教学、科研经验,融会于丛书中。相信该丛书必将对广大电气工程与电气信息科学技术人员和在校师生有较大的帮助。



2008年6月

◎前言

Preface

可编程逻辑器件(PLD, Programmable Logic Device)是 20 世纪 70 年代发展起来的新型逻辑器件,可以完全由用户配置以完成某种特定的逻辑功能。经过 80 年代的发展,PLD 行业初步形成,而进入 90 年代以后,PLD 已成为半导体领域中发展最快的产品之一。

可编程逻辑器件是在专用集成电路(ASIC, Application Specific Integrated Circuits)设计的基础上发展起来的,在 ASIC 设计方法中,通常采用全定制和半定制电路设计方法,但设计完成后如果不能满足要求,还要重新设计再进行验证。这样不但会导致设计开发周期变长,产品上市时间也难以保证,而且会大大增加产品的开发费用。从 ASIC 设计的风险来看,可编程逻辑器件正好解决了这一问题。随着工艺、技术及市场的不断发展,PLD 产品的价格越来越低,集成度越来越高,速度越来越快,并在越来越多的领域中取代了 ASIC。

PLD 产品的优势在于可以缩短开发周期,现场灵活性好,开发风险小,且随着芯片制造工艺的不断进步,单片集成度飞速提高,价格也越来越高,已广泛应用于电子、通信、航天及军事等领域。

PLD 产品集成度的不断提高使得产品的性能不断提高,功能不断增多。最早的 PLD 仅仅能够实现一些简单的逻辑功能,而现在片上可编程系统直接实现系统集成,在速度上可以满足一般系统的要求。其好处是用户把所有关键的功能块放上去后,可以随着标准的改变而重新配置,降低费用,并且缩短开发时间。同时,IP(Intellectual Property)核在 PLD 中的使用也使得片上可编程系统(SOPC, System on a Programmable Chip)成为可能。2000 年,Altera 推出了高密度的 APEX 系列器件和开发工具 Quartus II,再加上一系列可重复使用的 IP 核, SOPC 开始有了实质性的发展。之后,Altera 从 MIPS 和 ARM 公司获得了处理器层次结构的许可权,将这两种芯核纳入 Altera 的 IP 核,以便将晶体管级处理器内核嵌入自己的复杂可编程逻辑器件(CPLD, Complex Programmable Logic Device)层次结构中,并针对 SOPC 推出了一套嵌入式处理器产品,包括 ARM、MIPS 和 Nios 处理器三种芯核及相应的开发工具。

不久,另一 PLD 厂商 Xilinx 公司也宣布推出了新型的 SOPC 方案。Xilinx 与 ARCCores 共同为 ARC 的 32 位可配置处理器的用户提供可编程逻辑解决方案。ARC 的 32 位可配置处理器将被用于 Xilinx Virtex 和 Spartan II 系列的现场可编程门阵列(FPGA, Field Programmable Gate Array)中。ARC 处理器与传统的 RISC 处理器不同,它的指令集可以根据用户的确切需要进行配置。ARC 工具集允许按处理器规格改制,使其在较低的时钟频率下具有更好的系统性能和更低的功耗,成为理想的以 FPGA 为基础的软件解决方案。

PLD 的飞速发展使得传统设计方法及工具逐渐被抛弃,新器件、新理念、新思路的掌握与

理解对于当代有志于 PLD 技术的电子工程师来说是一个新的挑战,掌握 SOPC 开发技术、VHDL 语言与 FPGA 开发技术是每一位电子设计工程师重要而紧迫的任务。为培养我国的电子设计、ASIC/SOPC 设计开发、IP 核应用和开发、具有自主知识产权的电子系统的开发人才,以及在电子、信息、通信、电子对抗、工控类高等教学领域深入推广 EDA 和 SOPC 技术,满足高技术人才市场的需求,我们编写了本书。

全书共分6章,在编写中力求准确并注意系统性,本书具有如下特色:

- (1) 覆盖面广,包括了目前工程中应用的可编程逻辑器件及开发环境。
- (2) 内容新,包括了目前最新的可编程逻辑器件和开发环境。
- (3) 通俗易懂,由浅入深,通过实例讲解 VHDL、Verilog 语言及各种器件的编程方法。
- (4) 工程实践性强,教材的设计样例来自于工程实际,可以很好地培养学生的动手实践能力。

本书由杨春玲、朱敏主编,参加本书编写的还有刘贵栋和杨荣峰。由于编者水平有限,书中难免有错误或不当之处,恳请读者批评指正。

编者 刘彭琴

2008 年 3 月

于哈尔滨工业大学

◎ 目录

Contents

第1章 EDA 技术概述	1
1.1 EDA 技术的发展概况	1
1.2 可编程逻辑器件的发展概况	2
1.2.1 PLD 的发展	2
1.2.2 PLD 的主要特点	4
1.3 PLD 和其他技术的比较	5
1.3.1 PLD 和 ASIC 的比较	5
1.3.2 PLD 和微处理器、DSP 的比较	5
1.4 硬件描述语言概述	5
1.4.1 VHDL、Verilog 和 ABEL 三种语言的对比	6
1.4.2 VHDL 语言	6
1.5 EDA 与传统电子设计方法	7
1.5.1 EDA 设计方法概述	7
1.5.2 EDA 设计方法的优势	8
1.6 IP 核	9
1.6.1 IP 的基本概念	9
1.6.2 IP 的主要来源	9
1.6.3 IP 现状	11
第2章 PLD 结构与应用	13
2.1 PLD 的分类和特点	13
2.1.1 PLD 的基本结构	13
2.1.2 PLD 的分类	13
2.2 CPLD 简介	16
2.2.1 CPLD 产品概述	16
2.2.2 Altera 公司 MAX7000 系列 CPLD 的结构	16
2.3 CPLD 的编程与配置	19
2.3.1 大规模可编程逻辑器件的编程工艺	19

2.3.2 CPLD 的编程和 FPGA 的配置方式	20
2.4 FPGA 的基本结构	21
2.4.1 FPGA 的基本概念	21
2.4.2 FPGA 和 CPLD 的选型	25
2.4.3 FPGA 三大厂商比较	25
2.5 Altera Cyclone II FPGA 器件介绍	26
2.5.1 Cyclone II FPGA 结构	26
2.6 FPGA 的配置	31
2.6.1 串行配置器件	31
2.6.2 FPGA 配置方式	31
2.6.3 FPGA 配置举例	32
第 3 章 可编程逻辑器件的开发环境	35
3.1 可编程数字系统设计的输入	35
3.1.1 常用的可编程数字系统设计输入方式	35
3.1.2 可编程逻辑器件的开发环境	35
3.2 Altera 开发环境 MAX + plus II 使用入门	36
3.2.1 原理图方式输入	36
3.2.2 硬件描述语言 VHDL 输入	42
3.3 Quartus II 软件的使用	46
3.3.1 设计输入流程	47
3.3.2 正弦信号数据 ROM 定制	52
3.4 Xilinx 开发环境 Foundation 使用入门	65
3.5 ISE 集成开发环境使用入门	79
3.5.1 ISE 集成开发环境介绍	79
3.5.2 ISE 集成开发环境设计流程	79
第 4 章 VHDL 语言	90
4.1 VHDL 结构	90
4.1.1 组合电路:二选一选择器	90
4.1.2 D 触发器的设计	96
4.2 译码器与编码器	101
4.2.1 译码器的设计	101
4.2.2 编码器	105
4.3 比较器	107

4.4	数码转换电路	111
4.5	算术运算	115
4.5.1	数据类别	116
4.5.2	进位的处理	116
4.6	计数器	123
4.6.1	4 位加法计数器	123
4.6.2	带有复位和时钟使能的十进制计数器	125
4.7	移位寄存器	131
第 5 章	VeriLog 语言	134
5.1	Verilog 概述	134
5.1.1	Verilog HDL 历史与简介	134
5.1.2	Verilog HDL 的特点	134
5.1.3	Verilog HDL 语言的描述风格	135
5.2	Verilog HDL 结构	136
5.2.1	组合逻辑:二选一选择器	136
5.2.2	同步时序:D 触发器	138
5.2.3	Verilog 的基本规范	142
5.3	运算符	142
5.3.1	算术运算符	143
5.3.2	关系运算符	143
5.3.3	等式运算符	144
5.3.4	逻辑运算符	144
5.3.5	位操作符	144
5.3.6	缩减(Reduction)操作符	145
5.3.7	移位操作符	145
5.3.8	条件操作符	146
5.3.9	拼接操作符(Concatenation)	146
5.3.10	运算符的执行顺序	146
5.4	数据选择器	147
5.4.1	case 语句描述的 4 选 1 数据选择器	148
5.4.2	casez 描述的数据选择器	150
5.4.3	if-else 语句实现的 4 选 1 数据选择器	151
5.4.4	其他形式的数据选择器	154
5.5	编码器和译码器	155

111	5.5.1 case 语句实现的 8-3 编码器	155
112	5.5.2 for 语句实现的 8-3 编码器	158
113	5.5.3 七段数码管译码器	160
114	5.6 数字相关器	162
115	5.7 计数器	165
116	5.7.1 4 位计数器	165
117	5.7.2 4 位计数器的仿真程序	165
118	5.7.3 同步置数同步清零计数器	170
119	5.7.4 约翰逊计数器	171
120	5.7.5 模 60 的 BCD 码加法计数器	172
121	5.8 状态机	173
122	5.8.1 moore 状态机	173
123	5.8.2 mealy 状态机	175
124	5.8.3 售货机例子	177
125	第 6 章 数字系统设计实例	179
126	6.1 4 位十进制频率计设计	179
127	6.1.1 要求	179
128	6.1.2 原理	179
129	6.1.3 VHDL 源程序	180
130	6.2 数据采集与滤波系统	185
131	6.2.1 要求	185
132	6.2.2 原理	185
133	6.2.3 Verilog 源程序	186
134	6.3 交通灯控制	189
135	6.3.1 要求	189
136	6.3.2 原理	189
137	6.3.3 源程序	189
138	参考文献	192

第1章

EDA 技术概述

内容提要:本章主要介绍 EDA 技术及可编程逻辑器件的发展概况,PLD 和其他技术的比较,硬件描述语言,EDA 与传统电子设计方法比较,IP 核的概念等。通过本章的学习,使读者对 EDA 技术有一个初步的了解。

1.1 EDA 技术的发展概况

现代电子系统一般由模拟系统、数字系统和微处理系统三大部分组成。随着半导体技术、集成技术和计算机技术的发展,电子系统的设计方法和设计手段发生了很大的变化。特别是 EDA(Electronic Design Automation)技术,即电子设计自动化技术的发展和普及更是给电子系统的设计插上了腾飞的翅膀。

EDA 技术,是指利用计算机完成电子系统的设计,即以计算机为工作平台、EDA 软件工具为开发环境、硬件描述语言为设计语言、专用集成电路(ASIC, Application Specific Integrated Circuits)为实现载体的电子产品自动化设计过程。因此,EDA 技术以计算机科学和微电子技术发展为先导,汇集了计算机图形学、拓扑学、逻辑学、微电子工艺与结构学和计算数学等多种计算机应用学科的最新成果。

1. EDA 技术发展的初期 CAD 阶段

20 世纪 60 年代中期至 80 年代初期是 EDA 技术发展的初期阶段。这一阶段的主要特点是一些单独的软件工具的出现,主要有 PCB(Printed Circuit Board)布线设计、电路模拟、逻辑模拟及版图的绘制等。这个时期,人们主要借助计算机对所设计的电路的性能进行一些模拟和预测,所以这一阶段又称为计算机辅助设计 CAD(Computer Aided Design)阶段。

2. EDA 的计算机辅助设计 CAE 阶段

20 世纪 80 年代初期至 90 年代初期为 EDA 技术的计算机辅助设计 CAE(Computer Aided Engineering)阶段。这个阶段在集成电路与电子设计方法学以及设计工具集成化方面取得了长足的进步。各种设计工具,如原理图输入、编译与连接、逻辑模拟、测试码生成、版图自动布局以及各种单元库已齐全。不同功能的设计工具之间的兼容性得到了很大的改善。EDA 软件设计者采用统一数据管理技术,把多个不同功能的设计软件结合成一个 CAE 系统。按照设计方法学制定的设计流程,在一个集成的设计环境中可以实现从设计输入到版图输出的全程设计自动化。在这个阶段,基于门阵列和标准单元库设计的各种半定制 ASIC 得到了极大的发

展,将集成电路和电子系统设计推入了 ASIC 时代。

3. 第三代 EDA 技术

20 世纪 90 年代以来,微电子技术以惊人的速度发展,其工艺水平达到深亚微米级,在一个芯片上可集成上百万乃至上亿只晶体管,芯片的工作频率可达到 GHz 级。这不仅为片上系统(SOC, System On Chip)的实现提供了可能,同时也对 EDA 技术提出了更高的要求,并促进了 EDA 技术的发展。此阶段主要出现了以硬件语言描述、系统级仿真和综合技术为基本特征的第三代 EDA 技术,它使设计人员摆脱了大量的具体性和基础性工作,而把更多的精力投入到创造性的方案与概念的构思上。从而极大地提高了系统的设计效率,缩短了产品的研制周期。

4. 21 世纪的 EDA 技术

EDA 技术在进入 21 世纪后,得到了更大的发展。主要表现为在现场可编程门阵列(FPGA, Field Programmable Gate Array)上实现数字信号处理(DSP, Digital Signal Processing)成为可能,基于 FPGA 的 DSP 技术为高速数字信号处理算法提供了实现途径;嵌入式处理器软核的成熟,使得片上集成系统(SOPC, System on a Programmable Chip)步入大规模应用阶段,在一片 FPGA 上实现一个完备的数字处理系统成为可能;基于 EDA 技术的用于 ASIC 设计的标准单元已涵盖大规模电子系统及复杂 IP 核模块;软、硬 IP(Intellectual Property)核在电子产业的产业领域广泛应用;系统级、行为验证级硬件描述语言的出现(如 System C),使复杂电子系统的设计和验证趋于简单。

1.2 可编程逻辑器件的发展概况

传统电子产品设计的基本思路一般是先选用标准通用集成电路芯片,再由这些芯片“自下而上”地构成电路、子系统和系统。EDA 技术则采用“自上而下”的设计方法。在这种新的设计方法中,由整机系统用户对整个系统进行方案设计和功能划分,系统的关键电路由一片或几片专用集成电路 ASIC 完成。ASIC 的设计与制造,已不再完全由半导体厂商独立承担,系统设计师在实验室里就可以设计出合适的 ASIC 芯片,并且立即投入实际应用之中,这都得益于可编程逻辑器件(PLD, Programmable Logic Device)的出现。现在应用最广泛的 PLD 主要是现场可编程门阵列(FPGA, Field Programmable Gate Array)和复杂可编程逻辑器件(CPLD, Complex Programmable Logic Device)。

可编程逻辑器件是一种由用户根据自己要求来构造逻辑功能的数字集成电路,利用计算机辅助设计,经一系列编译或转换程序,生成相应的目标文件,再由编程器或下载电缆将设计文件配置到目标文件中,这时 PLD 就可作为满足用户要求的专用集成电路使用了。PLD 适宜于小批量生产的系统,或在系统开发研制过程中采用,因此,应用较为广泛。它的应用和发展不仅简化了电路设计,降低了成本,提高了系统的可靠性和保密性,而且给数字设计方法带来了重大变化。

1.2.1 PLD 的发展

1. 可编程只读存储器(PROM)

最早的 PLD 是 1970 年制成的可编程只读存储器(PROM, Programmable Read Only Memory),它由固定的与阵列和可编程的或阵列组成。PROM 采用熔丝工艺编程,只能写一次,不能擦除

和重写。随着技术的发展和应用要求,此后又出现了紫外线可擦除只读存储器 UVEPROM 和电可擦除只读存储器 E²PROM。由于其阵列规模大、速度低、价格便宜、易于编程,适合于存储函数和数据表格,因此,主要用作存储器。典型的 EPROM 有 2716 和 2732 等。

2. 可编程逻辑阵列(PLA)

20 世纪 70 年代中期出现了可编程逻辑阵列(PLA, Programmable Logic Array),它是由可编程的与阵列和可编程的或阵列组成,但由于器件的资源利用率低,价格较贵,编程复杂,支持 PLA 的开发软件有一定难度,因而没有得到广泛应用。

3. 可编程阵列逻辑(PAL)器件

1977 年美国 MMI 公司(单片存储器公司)率先推出可编程阵列逻辑(PAL, Programmable Array Logic)器件,它由可编程的与阵列和固定的或阵列组成,采用熔丝编程方式,双极性工艺制造,器件的工作速度很高。由于它的输出结构种类很多,设计很灵活,因而成为第一个得到普遍应用的可编程逻辑器件,如 PAL16L8。

4. 通用阵列逻辑器件

1985 年 Lattice 公司最先发明了通用阵列逻辑(GAL, Generic Array Logic)器件。GAL 器件在 PAL 器件基础上采用了输出逻辑宏单元形式 E²CMOS 工艺结构,具有可电擦写、重复编程、长期保存数据、设置加密位和重新组合结构等优点。具有代表性的 GAL 芯片有 GAL16V8 和 GAL20V8,这两种 GAL 几乎能够仿真所有类型的 PAL 器件。在实际应用中,GAL 器件对 PAL 器件仿真具有百分之百的兼容性,所以 GAL 器件几乎完全代替了 PAL 器件,并可以取代大部分 SSI、MSI 数字集成电路,如标准的 54/74 系列器件,因而得到广泛应用。

PAL 和 GAL 器件都属于简单的 PLD,结构简单,设计灵活,对开发软件的要求低,但规模小,难以实现复杂的逻辑功能。随着技术的发展,简单 PLD 在集成密度和性能方面的局限性也暴露出来,其寄存器、I/O 引脚、时钟资源的数目有限,没有内部互连,因此,包括 EPLD、CPLD 和 FPGA 在内的复杂 PLD 迅速发展起来,并向着高密度、高速度、低功耗,以及结构体系更灵活、适用范围更宽广的方向发展。

5. 可擦除可编程逻辑器件(EPLD)

可擦除可编程逻辑器件(EPLD, Erasable PLD)是 20 世纪 80 年代中期 Altera 公司推出的基于 UVEPROM 和 CMOS 技术的 PLD,后来发展到采用 E²CMOS 工艺制作的 PLD。EPLD 基本逻辑单元是宏单元。宏单元由可编程的与或阵列、可编程寄存器和可编程 I/O 三部分组成。从某种意义上讲 EPLD 是改进的 GAL,它在 GAL 基础上大量增加输出宏单元的数目,提供更大的与阵列,灵活性较 GAL 有较大改善,集成密度大幅度提高,内部连线相对固定,延时小,有利于器件在高频率下工作,但内部互连能力十分弱。世界著名的半导体器件公司,如 Altera、Xilinx、AMD、Lattice 等,均有 EPLD 产品,但结构差异较大。

6. 在线可编程(ISP)技术

20 世纪 80 年代末 Lattice 公司提出在线可编程(ISP, in System Programmability)技术,之后出现了 CPLD 器件。CPLD 是在 EPLD 的基础上发展起来的,采用 E²CMOS 工艺制作,与 EPLD 相比,增加了内部连线,对逻辑宏单元和 I/O 单元也有重大的改进。CPLD 至少包含三种结构:可编程逻辑宏单元、可编程 I/O 单元和可编程内部连线。部分 CPLD 器件内部还集成了 RAM、FIFO 或双口 RAM 等存储器,以适应 DSP 应用设计的要求。其典型器件有 Altera 公司的 MAX7000 系列,Xilinx 公司的 7000 和 9500 系列,Lattice 公司的 PLSI/ispLSI 系列和 AMD 公司的

MACH 系列。

7. 现场可编程门阵列(FPGA)器件

1985 年, Xilinx 公司提出可编程概念, 同时推出世界上第一片现场可编程门阵列 FPGA 器件。它是一种新型的高密度 PLD, 采用 CMOS-SRAM 工艺制作。其内部由许多独立的可编程逻辑模块(CLB)组成, 逻辑块之间可以灵活地相互连接。FPGA 的结构一般分为三部分: 可编程逻辑块、可编程 I/O 模块和可编程内部连线。CLB 的功能很强, 不仅能够实现逻辑函数, 还可以配置成 RAM 等复杂的形式。配置数据存放在片内的 SRAM 或者熔丝图上, 基于 SRAM 的 FPGA 器件工作前需要从芯片外部加载配置数据。配置数据可以存储在片外的 EPROM 或者计算机上, 设计人员可以控制加载过程, 在现场修改器件的逻辑功能, 即所谓现场可编程。FPGA 出现后受到电子设计工程师的普遍欢迎, 发展十分迅速。Xilinx、Altera 和 Actel 等公司都提供高性能的 FPGA 芯片。

8. 高密度 PLD

20 世纪 90 年代后, 高密度 PLD 在生产工艺、器件的编程和测试技术等方面都有了飞速发展。目前, PLD 的集成度非常高, 如 Altera 公司的 APEX II 系列器件, 采用 $0.15\ \mu\text{m}$ 工艺, 各层都是铜金属布线, 其中 EP2A90 的密度可达 400 万典型门, 可为用户提供 1 140 个 I/O 引脚, 1 GB/s 数据速率。

9. SOPC 器件

20 世纪末出现了 SOPC 器件, SOPC 是现代电子技术和电子系统设计的汇聚点和最新发展方向, 它将普通 EDA 技术、计算机系统、嵌入式系统、工业自动化控制系统、DSP 及无线电等融为一体, 涵盖了嵌入式系统设计技术的全部内容。SOPC 结合了 SOC、PLD 和 FPGA 的优点, 集成了硬核或软核 CPU、DSP、存储器、外围 I/O 及可编程逻辑, 用户可以利用 SOPC 平台自行设计各种高速、高性能的 DSP 处理器或特定功能的 CPU 处理器, 从而使电子系统设计进入了一个全新的模式。在应用的灵活性和价格上, SOPC 具有极大的优势, 被称为“半导体产业的未来”。Xilinx 公司和 Altera 公司的新一代 FPGA 集成了中央处理器(CPU)或数字处理器(DSP)内核, 在一片 FPGA 上进行软硬件协同设计, 为实现 SOPC 提供了强大的硬件支持。

世界各著名半导体器件公司, 如 Altera、Xilinx、Lattice、Actel 和 AMD 公司等, 均可提供不同类型的 CPLD、FPGA 产品。众多公司的竞争, 促进了可编程集成电路技术的提高, 使其性能不断改善, 产品日益丰富, 价格逐步下降。可以预计, 可编程逻辑器件将在结构、密度、功能、速度和性能等方面得到进一步发展, 结合 EDA 技术, PLD 将在现代电子系统设计中得到非常广泛的应用。

1.2.2 PLD 的主要特点

(1) 大规模。PLD 的逻辑规模已达数百万门, 近 10 万逻辑宏单元, 可以将一个复杂的电路系统, 如包括一个或多个嵌入式系统处理器、各类通信接口、控制模块和 DSP 模块等, 装在一个芯片, 即能满足片上系统 SOC 设计。

(2) 低功耗。由 Lattice 公司最新推出的 ispMACH4000z 系列 CPLD 达到了前所未有的低功耗性能, 静态电流 $20\ \mu\text{A}$, 以至于被称为 0 功耗器件, 而其他性能, 如速度、规模、接口特性等仍然保持了很好的指标。

(3) 模拟可编程。各种应用 EDA 工具软件设计、ISP 方式编程下载的模拟可编程及模数

混合可编程器件不断出现,最具代表性的器件是 Lattice 公司的 ispPAC 系列器件。

(4) 含多种专用端口和附加功能模块的 FPGA。例如, Lattice 公司的 ORT、ORSO 系列器件, Altera 公司的 Stratix、Cyclone、APEX 等系列器件,除内嵌大量的 ESB(嵌入式系统块)外,还含有嵌入的锁相环模块(用于时钟发生和管理)、嵌入式微处理核等。此外,系列器件还嵌有丰富的 DSP 模块。

1.3 PLD 和其他技术的比较

数字系统中会用到多种集成电路,如微处理器、DSP、存储器、半导体厂商提供的专用集成电路、用户自己设计的专用集成电路和 PLD 等。用户选择哪一种往往取决于成本、实现的难易程度、速度、可靠性与偏好等多种因素。

1.3.1 PLD 和 ASIC 的比较

ASIC 可分为数字 ASIC 和模拟 ASIC,数字 ASIC 又有全定制和半定制两种。PLD 和 ASIC 相比主要具有以下几个优点:

(1) PLD 具有在线升级能力。ASIC 一旦设计完成,就不能更改,所以 ASIC 不具有 PLD 的在线升级能力。

(2) PLD 设计周期短。ASIC 的设计不仅要像 PLD 的设计一样进行逻辑验证、时序分析等工作,还要进行版图、位置和互连线设计,因此设计周期长。

(3) PLD 启动的成本小。做一个 ASIC 至少需要几万片的量级,而 PLD 不存在这个问题,故而启动成本小。

1.3.2 PLD 和微处理器、DSP 的比较

PLD 和微处理器、DSP 都具有可编程的能力,所以它们在数字系统中的使用都非常广泛。但 PLD 和微处理器在设计上有几点不同之处:

(1) 通过 PLD 或 ASIC 实现的功能会比通过微处理器控制快得多。其中原因之一是一般微处理器是数字系统的核心,它通常要处理很多的任务,对于其中一个任务只能分给一定的时间片,而用 PLD 和 ASIC 实现一般是专门化的;另外一个原因是微处理器是通用的硬件结构,通过指令完成相应的逻辑功能,而 PLD 和 ASIC 可以设计专门的硬件结构。

(2) PLD 的设计方法和设计思路与微处理器、DSP 的设计有很大的差别。

1.4 硬件描述语言概述

硬件描述语言(HDL, Hardware Description Language)是 EDA 技术的重要组成部分,常用的硬件描述语言有 VHDL、Verilog 和 ABEL 语言。VHDL 语言是电子设计主流硬件的描述语言,英文全名是 VHSIC(Very High Speed Integrated Circuit),1983 年由美国国防部(DOD)发起创建,Verilog 语言起源于集成电路设计,ABEL 语言则来源于可编程逻辑器件的设计。

1.4.1 VHDL、Verilog 和 ABEL 三种语言的对比

下面从使用角度对 VHDL、Verilog 和 ABEL 三种语言进行对比。

1. 逻辑描述层次

VHDL 语言是一种高级描述语言,适用于行为级和 RTL 级的描述,最适于描述电路的行为;Verilog 语言和 ABEL 语言是一种较低级的描述语言,适用于 RTL 级和门电路级的描述,最适合描述门电路级。

2. 设计要求

使用 VHDL 进行电子系统设计时可以不了解电路的内部结构,设计者所做的工作较少;使用 Verilog 语言和 ABEL 语言进行电子系统设计时需了解电路的详细结构,设计者需要做大量的工作。

3. 综合过程

任何一种语言源程序,最终都要转换成门电路级才能被布线器或适配器所接受。因此,VHDL 语言源程序的综合通常要经过行为级—RTL 级—门电路级的转化,几乎不能直接控制门电路的生成。而 Verilog 语言和 ABEL 语言源程序的综合过程较为简单,即通过 RTL 级—门电路级的转化,易于控制电路资源。

4. 对综合器的要求

VHDL 语言描述层次较高,不易控制底层电路,因而对综合器的性能要求较高,Verilog 语言和 ABEL 语言对综合器的要求较低。

1.4.2 VHDL 语言

VHDL 语言由电子电气工程师学会(IEEE, Institute of Electrical and Electronics Engineers)进一步发展,并在 1987 年作为“IEEE 标准 1076”发布。从此,VHDL 成为硬件描述语言的业界标准之一。自 IEEE 公布了 VHDL 的标准版本之后,各 EDA 公司相继推出了自己的 VHDL 设计环境,或宣布自己的设计工具支持 VHDL。此后,VHDL 在电子设计领域得到了广泛应用,并逐步取代了原有的非标准硬件描述语言。

VHDL 作为一个规范语言和建模语言,随着 VHDL 的标准化,出现了一些支持该语言的行为仿真器。由于创建 VHDL 的最初目标是用于标准文档的建立和电路功能模拟,其基本想法是在高层次上描述系统和元件的行为。但到了 20 世纪 90 年代初,人们发现,VHDL 不仅可以作为系统模拟的建模工具,而且可以作为电路系统的设计工具;可以利用软件工具将 VHDL 源码自动地转化为文本方式表达的基本逻辑元件连接图,即网表文件。这种方法显然对于电路自动设计是一个极大的推进。很快,电子设计领域出现了第一个软件设计工具,即 VHDL 逻辑综合器,它把标准 VHDL 的部分语句描述转化为具体电路实现的网表文件。

1993 年,IEEE 对 VHDL 进行了修订,从更高的抽象层次和系统描述能力上扩展了 VHDL 的内容,公布了新版本的 VHDL,即 IEEE 标准的 1076—1993 版本。现在,VHDL 和 Verilog 作为 IEEE 的工业标准硬件描述语言,得到众多 EDA 公司的支持,在电子工程领域,已成为事实上的通用硬件描述语言。

VHDL 语言具有很强的电路描述和建模能力,能从多个层次对数字系统进行建模和描述,从而大大简化了硬件设计任务,提高了设计效率和可靠性。

用 VHDL 完成一个确定的设计,可以利用 EDA 工具进行逻辑综合和优化,并自动把 VHDL 描述设计转变成门级网表(根据不同的实现芯片)。这种方式突破了门级设计的瓶颈,极大地减少了电路设计的时间和可能发生的错误,降低了开发成本。利用 EDA 工具的逻辑优化功能,可以自动地把一个综合后的设计变成一个更小、更高速的电路系统。反过来,设计者还可以容易地从综合和优化的电路获得设计信息,返回去更新修改 VHDL 设计描述,使之更加完善。

VHDL 具有与具体硬件电路和设计平台无关的特性,并且具有良好的电路行为描述和系统描述的能力,同时在语言易读性和层次化、结构化设计方面,表现了强大的生命力和应用潜力。因此,VHDL 在支持各种模式的设计方法、自上而下与自下而上或混合方法方面,在面对当今许多电子产品的生命周期缩短、需要多次重新设计以溶入最新技术、改变工艺等方面都表现了良好的适应性。利用 VHDL 进行电子系统设计,其中一个很大的优点是设计者可以专心致力于其功能的实现,而不需要对不影响功能的与工艺有关的因素花费过多的时间和精力。

1.5 EDA 与传统电子设计方法

1.5.1 EDA 设计方法概述

1. 传统的“自下而上”设计方法

传统的电子设计技术通常是“自下而上”的,即首先确定构成系统的底层的电路模块或元件的结构和功能,然后根据主系统的功能要求,将它们组合成更大的功能块,使它们的结构和功能满足上层系统的要求。依此流程,逐步向上递推,直至完成整个目标系统的设计。对于一般的电子系统的设计,使用“自下而上”的设计方法,必须首先决定使用的器件类别和规格,如 74 系列的器件、某种 RAM 和 ROM、某类 CPU 或单片机,以及某些专用功能芯片等;然后是构成多个功能模块,如数据采集控制模块、信号处理模块、数据交换和接口模块等,直至最后利用它们完成整个系统的设计。这样的设计方法如同一砖一瓦地建造金字塔,不仅效率低、成本高,而且容易出错。

对于 ASIC 设计,则是根据系统的功能要求,首先从绘制硅片版图开始,逐级向上完成版图级、门级、RTL 级、行为级、功能级,直至系统级的设计。在这个过程中,任何一级发生问题,通常都不得不返工重来。

“自下而上”设计方法的特点是必须首先关注并致力于解决系统底层硬件的可获得性,以及它们的功能特性方面的诸多细节问题;在整个逐级设计和测试过程中,必须始终顾及具体目标器件的技术细节。在这个设计过程中的任一时刻,底层目标器件的更换,或某些技术参数不满足总体要求,或缺货,或由于市场竞争的变化,临时提出降低系统成本、提高运行速度等不可预测的外部因素,都将可能使前面的工作前功尽弃。由此可见,在某些情况下,“自下而上”的设计方法是一种低效、低可靠性、费时费力且成本高昂的设计方法。

2. EDA 的“自上而下”设计方法

EDA 技术为我们提供了一种“自上而下”的全新设计方法。这种设计方法首先从系统设计入手,在顶层进行功能方框图的划分和结构设计。在方框图一级进行仿真、纠错,并用硬件描述语言对高层次的系统行为进行描述,在系统一级进行验证;然后用综合优化工具生成具体