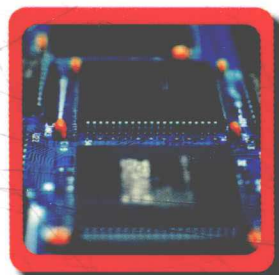




全国高等院校仪器仪表及自动化类“十三五”规划教材

可编程逻辑器件 与VHDL设计

◎ 靳 鸿 主编 ◎ 郭华玲 沈大伟 副主编



采用先建立知识框架再进行内容填充的介绍方式
以功能为牵引，分析不同风格和语句实现的区别
内容实践性强，知识点与实际科研项目相结合

中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

全国高等院校仪器仪表及自动化类“十三五”规划教材

可编程逻辑器件与 VHDL 设计

靳 鸿 主 编

郭华玲 沈大伟 副主编

赵正杰 孟令军 崔建峰
郭文凤 王 燕 刘 喆 参 编



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以存储测试系统的控制模块为设计对象,在介绍 VHDL 相关基础知识的前提下,详细讲解了如何采用 VHDL 进行控制模块关键子模块的设计方法。在对 VHDL 语言的数据对象、类型、数据结构及基本语句进行描述的基础上,着重将以上基础内容与实际的控制模块设计实现相结合,在不断强化基础的同时给出了 VHDL 在工程中的应用实例,对如何根据功能要求进行设计也有相关论述。本书内容丰富,实践性强,章节之间注重知识整体性,对应用 VHDL 进行系统设计有较强的指导和参考作用。

全书共 11 章,第 1~4 章是关于测试系统控制模块设计的基础理论,第 5~7 章是 VHDL 语言的基础理论,第 8~11 章是各控制模块的 VHDL 设计与实现方法及数字电路中常见的设计方法。

本书可作为高等院校测控技术与仪器、机械工程及自动化等专业的本科生和研究生教材,也可以供从事电子仪器设计和调试工作的相关工程技术人员自学和参考。

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

可编程逻辑器件与 VHDL 设计/靳鸿主编. —北京:电子工业出版社,2017.3

全国高等院校仪器仪表及自动化类“十三五”规划教材

ISBN 978-7-121-30775-1

I. ①可… II. ①靳… III. ①可编程序逻辑阵列—高等学校—教材②硬件描述语言—程序设计—高等学校—教材 IV. ①TP332.1②TP312

中国版本图书馆 CIP 数据核字(2016)第 322894 号

策划编辑:郭穗娟

责任编辑:郭穗娟 特约编辑:顾慧芳

印 刷:涿州市京南印刷厂

装 订:涿州市京南印刷厂

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:11.5 字数:294 千字

版 次:2017 年 3 月第 1 版

印 次:2017 年 3 月第 1 次印刷

定 价:45.00 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)88254888,88258888。

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。

本书咨询方式:(010)88254502,guosj@phei.com.cn。

前言

随着计算机和大规模集成电路制造技术的迅猛发展,现代复杂数字逻辑系统正向小型化、低功耗和高集成度方向发展。具有可重复使用、可移植性好、系统设计效率高等优点的可编程逻辑器逐渐成为电子系统设计的主流。硬件描述语言 VHDL (Very-High-Speed Integrated Circuit Hardware Description Language) 具有很强的行为描述能力,是进行可编程逻辑器件设计的主要工具,随着系统复杂性和智能化的不断提高, VHDL 已成为不可替代的语言工具,将在通信、工业、航空、航天等领域担负重任。

本书从系统对 VHDL 的需求入手,在明确 VHDL 设计目的的基础上,结合此语言的相关知识,建立 VHDL 的设计思路。进行程序设计时,强调思路是设计的灵魂,先有思路才能很好地构思整个程序的结构,优化其算法。本书以控制模块的功能需求为背景,在明确了需要什么的基础上,提出如何满足需要的问题。这样,采用 VHDL 进行程序设计时,可以先对程序建立总体的认识,再对如何构建程序进行思考,以功能为牵引,分析不同风格和语句实现的区别,掌握最优的 VHDL 设计方案。书中以信息获取系统控制模块设计为主线,除了扩展各具体模块的应用特点外,还考虑了各模块间的相互关系,一方面加强对语言的认识,另一方面将理论与实践进行结合,充分考虑工程实际的应用背景,建立理论学习与实际使用的衔接。

本书是作者从事十多年教学经验的积累和科研结果的体现,实践性强;知识点与实际科研项目相结合;采用先建立知识框架再进行内容填充的介绍方式,便于读者的学习和理解。希望本书,能够使读者了解 VHDL,建立 VHDL 程序设计的思路,掌握其设计的步骤和基本方法,对实践起到一定的指导作用。

本书共 11 章。第 1 章是绪论;第 2~4 章是对测试系统控制模块的相关描述,包括系统组成及控制模块的主要功能,控制模块的设计方法,基于 VHDL 的控制模块设计流程等;第 5~7 章着重介绍 VHDL 语言的基础理论及常用的语句;第 8~10 章是各控制模块的 VHDL 设计与实现方法;第 11 章介绍了数字电路常用器件以及数字电路开发常用的设计方法。

本书第 1、9 章由中北大学靳鸿编写;第 2 章由中北大学孟令军编写;第 3 章由中北大学王燕编写;第 4 章由沈阳理工大学装备工程学院郭文凤编写;第 5 章由中北大学郭华玲编写;第 6 章由中北大学赵正杰编写;第 7、8 章由中北大学沈大伟编写;第 10 章由中北大学崔建峰编写;第 11 章由太原理工大学刘喆编写。靳鸿、郭华玲、沈大伟负责统稿。

全体编者在此书的编写过程中都尽心尽力,但因水平有限,书中难免存在不足或疏漏之处,恳请广大读者批评指正,不胜感激。

编 者
2017 年 1 月

目 录

第 1 章 绪论	1
1.1 集成技术与可编程逻辑器件	1
1.1.1 可编程逻辑器件	1
1.1.2 CPLD 和 FPGA	2
1.2 电子系统设计与 VHDL	3
1.2.1 传统系统的设计方法	3
1.2.2 VHDL 与“自顶向下”的设计方法	4
1.3 EDA、VHDL 及其应用	5
1.3.1 EDA	5
1.3.2 VHDL 特点	6
1.3.3 VHDL 设计流程及应用	7
第 2 章 测试系统构成及控制模块主要功能	13
2.1 测试系统的基本组成	13
2.1.1 系统模型	13
2.1.2 系统基本组成	13
2.2 控制模块的实现形式	14
2.2.1 基于可编程逻辑器件的设计与实现	14
2.2.2 基于单片机的设计与实现	17
2.3 控制模块主要控制功能	19
2.3.1 ADC 控制	19
2.3.2 存储器的控制	21
2.3.3 接口的控制	24
习题	25
第 3 章 控制模块设计方法	26
3.1 数字逻辑电路设计方法概述	26
3.1.1 通用逻辑器件设计方法	26
3.1.2 ASIC 及可编程逻辑器件设计方法	29
3.2 控制模块的状态设计	33
3.2.1 状态图及其组成	33
3.2.2 控制模块状态图设计	34

3.3	系统功能模块划分与接口	37
3.3.1	模块划分原则	37
3.3.2	功能模块划分	38
3.3.3	常用接口与总线	38
	习题	42
第 4 章	基于 VHDL 的控制模块设计流程	43
4.1	VHDL 设计一般流程	43
4.1.1	VHDL 实际流程	43
4.1.2	仿真软件	44
4.2	设计输入与功能仿真	45
4.2.1	指定设计项目名称	45
4.2.2	创建新的设计文件	45
4.2.3	VHDL 程序设计	46
4.2.4	功能仿真	46
4.3	项目编译与时序仿真	49
4.3.1	编译过程	49
4.3.2	编译器组成及说明	49
4.3.3	编译相关参数选取与设置	50
4.3.4	编译文件	52
4.3.5	时序仿真	52
4.4	器件下载编程和配置	53
	习题	54
第 5 章	VHDL 基础	55
5.1	硬件描述语言概述	55
5.2	VHDL 的数据对象	55
5.2.1	常量	56
5.2.2	变量	57
5.2.3	信号	58
5.3	VHDL 的数据类型	59
5.3.1	标准的数据类型	60
5.3.2	标准逻辑位数据类型	62
5.3.3	用户自定义数据类型	63
5.4	VHDL 的运算符	69
5.4.1	逻辑运算符	69
5.4.2	关系运算符	70
5.4.3	算术运算符	70
5.4.4	并置运算符	71
5.5	VHDL 的程序结构	72
5.5.1	库及程序包	73

5.5.2 实体	74
5.5.3 结构体	75
5.5.4 配置	76
习题	76
第 6 章 VHDL 语句	78
6.1 VHDL 程序结构	78
6.1.1 VHDL 的特点	78
6.1.2 VHDL 程序结构	79
6.2 顺序语句	82
6.2.1 赋值语句	82
6.2.2 IF 语句	83
6.2.3 CASE 语句	85
6.2.4 LOOP 语句	87
6.2.5 WAIT 语句	89
6.3 并行语句	90
6.3.1 进程语句	91
6.3.2 块语句	92
6.3.3 并行赋值语句	94
6.3.4 元件例化语句	97
习题	99
第 7 章 基于 VHDL 的状态机设计	101
7.1 状态机设计基础	101
7.1.1 状态机的分类	101
7.1.2 状态机的描述方法	102
7.1.3 状态机的设计步骤	102
7.2 NAND Flash 块擦除模块状态机设计	103
习题	107
第 8 章 A/D 控制模块的 VHDL 设计与实现	108
8.1 A/D 概述	108
8.2 采样定理	110
8.2.1 时域采样定理	110
8.2.2 频域采样定理	111
8.3 并行 A/D	111
8.3.1 典型并行 A/D——AD7492 概述	111
8.3.2 并行 A/D 控制命令	113
8.4 串行 A/D	116
8.4.1 典型串行 A/D——AD7274 概述	116
8.4.2 串行 A/D 控制命令	117

习题	122
第 9 章 存储器控制模块的 VHDL 设计与实现	124
9.1 存储器分类及使用特点	124
9.1.1 SRAM 存储器	124
9.1.2 Flash 存储器	125
9.1.3 铁电存储器	125
9.2 SRAM 存储器及其控制	125
9.2.1 SRAM 基本结构	125
9.2.2 SRAM 基本操作与 VHDL 设计	126
9.3 Flash 存储器概述	130
9.3.1 Flash 的基本结构	130
9.3.2 NAND Flash 访问方法	132
9.4 Flash 存储器控制	133
9.4.1 Flash 擦除	133
9.4.2 Flash 无效块检测	136
9.4.3 Flash 页编程	139
9.4.4 Flash 读操作	142
习题	146
第 10 章 异步串行通信 (UART) 模块设计	147
10.1 UART 协议简介	147
10.1.1 UART 接口标准	147
10.1.2 UART 通信协议	148
10.2 UART 协议控制器 FPGA 实现	148
10.2.1 UART 接口实现原理与方案	149
10.2.2 波特率时钟生成模块设计	150
10.2.3 数据接收/发送逻辑模块设计	150
10.2.4 数据奇偶校验模块设计	156
10.2.5 串并转换模块设计	157
10.2.6 数据接收/发送 FIFO 模块设计	158
10.3 测试仿真与设计调试注意事项	159
10.3.1 测试仿真	159
10.3.2 设计调试注意事项	160
习题	161
第 11 章 数字电路开发常用设计方法	162
11.1 毛刺现象及消除方法	162
11.2 几种逻辑器件及信号清零方法	163
11.2.1 触发器及锁存器	163
11.2.2 信号清零方法	164

11.3 数字电路中的同步设计.....	165
11.4 数字电路时延电路产生及用法.....	167
11.5 数字电路中的时钟设计.....	167
11.5.1 全局时钟	168
11.5.2 门控制时钟	168
11.5.3 多级逻辑时钟	168
11.5.4 行波时钟	168
11.5.5 多时钟系统	169
习题.....	171
参考文献.....	172

第1章 绪论

系统是由若干相互作用和相互依赖的事物组合而成的,具有特定功能的整体。其中,通常将由电子元器件或部件组成的能够产生、传输、采集或处理电信号及信息的客观实体称为电子系统。

电子系统设计的发展是与两个因素的产生、发展密切相关的,这两个因素的每一次飞跃都会影响系统设计的革命性变革。它们就是集成技术和电子设计自动化(Electronic Design Automation, EDA)技术。

1.1 集成技术与可编程逻辑器件

集成技术的发展包括模拟集成、数字集成和混合集成技术的发展。其中数字技术的发展比较成熟,成为集成技术的主流。

1.1.1 可编程逻辑器件

随着微电子技术与工艺的发展,数字集成电路由早期的电子管、晶体管经历了小规模集成电路、中规模集成电路、大规模集成电路、超大规模集成电路、具有特定功能的专用集成电路 ASIC(Application Specific Integrated Circuit),到现在出现了片上系统 SoC(System on Chip),集成度和功能不断增大和增强。在 20 世纪 80 年代初,随着微电子技术的高速发展,特别是 ASIC 的市场需求,出现了可编程逻辑器件 PLD(Programmable Logic Device),其中应用最广泛的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),目前已广泛应用于通信、电子、电力、军事、仪器仪表、影视等各个高科技研发领域和产品中。

20 世纪 80 年代中期 Altera 公司推出了一种新型的可擦除、可编程逻辑器件 EPLD(Erasable Programmable Logic Device),它采用 CMOS 和 UVEPROM 工艺制作,集成度比 PAL 和 GAL 高得多,设计也更加灵活,但内部互连能力比较弱。20 世纪 80 年代末 Lattice 公司提出了在系统可编程技术以后,相继出现了一系列具备在系统可编程能力的复杂可编程逻辑器件 CPLD。CPLD 是在 EPLD 的基础上发展起来的,它采用 E^2 CMOS 工艺制作,增加了内部连线,改进了内部结构体系,因而比 EPLD 性能更好,设计更加灵活,其发展也非常迅速。1985 年 Xilinx 公司首家推出了 FPGA 器件,它是一种新型的高密度 PLD,采用 CMOS-SRAM 工艺制

作, 其结构和阵列型 PLD 不同, 内部由许多独立的可编程逻辑模块组成, 逻辑块之间可以灵活地相互连接, 具有密度高、编程速度快、设计灵活和可再配置设计能力等许多优点。FPGA 出现后立即受到世界范围内电子设计工程师的普遍欢迎, 并得到迅速发展。

1.1.2 CPLD 和 FPGA

CPLD 和 FPGA 都是可编程逻辑器件, 它们是在 PAL、GAL 等逻辑器件的基础上发展起来的。同以往的 PAL、GAL 等比较, CPLD 和 FPGA 的规模更大, 可以替代几十甚至几千块通用 IC 芯片。这样的 CPLD 和 FPGA 实际上就是一个子系统部件, 因此受到了世界范围内电子工程设计人员的广泛关注和普遍欢迎。

1. CPLD

CPLD 器件的结构是一种与阵列可编程、或阵列固定的与或阵列形式。PAL、GAL 都采用这种形式, 但 CPLD 同它们相比, 增加了内部连线, 对逻辑宏单元和 I/O 单元也有重大改进。一般情况下, CPLD 器件中包含三种结构: 可编程逻辑宏单元、可编程 I/O 单元、可编程内部连线。部分 CPLD 器件内还集成了 RAM、FIFO 或双口 RAM 等存储器, 以适应 DSP 应用设计的要求。

CPLD 器件具有同 FPGA 器件相似的集成度和易用性, 在速度上还有一定的优势, 因此, 在可编程逻辑器件技术的竞争中它与 FPGA 并驾齐驱, 成为两支领导可编程器件技术发展的力量。

2. FPGA

FPGA 是新一代面向用户的可编程逻辑器件, 它的功能密度远远超过其他 PLD 器件, 一块 FPGA 可以替代 100~200 片标准器件或者 20~40 片 GAL 器件, 其 I/O 引脚数多达 100 余条。所以一片 FPGA 芯片可以替代多个逻辑功能复杂的逻辑部件, 或者一个小型数字系统。自 FPGA 问世以来, 它已在许多领域获得了广泛的应用。

逻辑单元型结构 LCA (Logic Cell Array) 的 FPGA 由三部分组成, 即逻辑单元阵列 CLB (Configurable Logic Block)、I/O 单元、互联资源。这种类型结构的特点是通过 CLB 编程实现逻辑功能; 通过对 I/O 单元编程确定输入或输出结构; 通过对互联资源编程实现 CLB 之间、CLB 与 I/O 单元之间、I/O 单元之间的互联关系, 从而实现用户所需要的逻辑功能。PAL 结构扩展型 FPGA 则是在 PLA 基础上加以改进和扩展, 大幅度增加了寄存器数量和 I/O 引脚数, 增设了可编程互联资源, 改善了互联模式, 改进了阵列结构, 使得芯片的利用率大大提高。

3. 两者的区别

CPLD 与 FPGA 在价格、性能、逻辑规模和封装 (包括 EDA 软件性能) 等方面各有千秋, 面对不同的开发项目, 使用者应该作出最佳的选择, 表 1-1 是对 CPLD/FPGA 在各个方面的比较。

随着电子技术的高速发展, 今天的 CPLD 和 FPGA 器件在集成度、功能和性能 (速度及可靠性) 方面已经能够满足大多数场合的使用要求。用 CPLD、FPGA 等大规模可编程逻辑器件取代传统的标准集成电路、接口电路和专用集成电路已成为技术发展的必然趋势。

表 1-1 CPLD 与 FPGA 的比较

比较点 \ 器件	FPGA	CPLD
结构工艺	SRAM	E ² PROM 或 Flash
基本结构	LUT 查找表	乘积项
Pin-pin 时延	不定	确定
配置存储器	需外挂 ROM	不需
保密性能	无保密性	可加密
工作电压	3.3V 或 2.5V	5V
编程工艺	通过 PC 并行口或专用编程器或单片机进行配置	ISP 在线编程
应用	主要针对要求不是很高，设计编辑较复杂的系统，适合于完成时序较多的逻辑电路	主要针对速度要求高，但设计逻辑又不是很复杂的系统，适合于完成各类算数和组合逻辑

1.2 电子系统设计与 VHDL

如今，现场可编程器件 FPGA 和复杂可编程逻辑器件 CPLD 已广泛应用于通信、工业自动化、智能仪表、图像处理、计算机领域，大有取代中、小规模集成电路之趋势，成为电子系统设计（本文简称为“系统设计”）的主流。

1.2.1 传统系统的设计方法

- 以标准集成电路为对象的传统系统设计方法一般按照以下步骤进行：
- （1）根据系统对硬件的要求，详细编制技术规格书，并画出系统控制流图。
 - （2）根据技术规格书和系统控制流图，对系统的功能进行分化，合理地划分功能模块，并画出系统功能框图。
 - （3）进行各功能模块的细化和电路设计。
 - （4）电路版图 PCB（Printed Circuit Board）设计。
 - （5）各个模块的功能调试。
 - （6）各个模块的硬件电路连接并调试。
 - （7）整个系统的硬件电路调试。
- 若其中某个步骤出现问题，可能需要重新进行 PCB 板的设计，并重新进行电路板的焊接和调试步骤，如此反复，直到整个系统完成为止。这种方法也称为“自底向上” (Bottom-Up) 的系统设计方法。
- 传统的电路设计方法，系统仿真和调试要在系统的硬件完成后才能进行，因此存在的问题只有在后期才能发现，一旦考虑不周，就要重新设计，使得设计的费用和周期增大。另外，由于设计文件是电原理图，如果设计的系统比较复杂，那么其原理图就要成千上万张，给存档、阅读和修改都带来了不便。
- 除此之外，这种设计方法在进行底层设计时，缺乏对整个系统总体性能的把握、效率低，

如果在整个系统完成后发现性能还需改进,修改起来就比较困难,不能适应系统规模与复杂度提高的发展趋势。

1.2.2 VHDL 与“自顶向下”的设计方法

以若干张原理图构成设计文件的传统电路设计方法随着电路系统的复杂性、智能性的发展,所需的系统也逐渐庞大,所需的图纸成千上万,这么多的原理图不仅在设计上存在不便,在归档、阅读、修改等方面也有许多问题。于是人们寻求一种新的设计方法用于解决这样的问题。

1. 硬件描述语言

借鉴软件编程的思想,人们希望把复杂的电子电路用文字文件方式描述并保存下来,方便他人了解电路内容,就诞生了最初的硬件描述语言。

从 20 世纪 60 年代开始,为了解决大规模复杂集成电路的设计问题,许多 EDA 厂商和科研机构就建立和使用着自己的电路硬件描述语言,如 Data I/O 公司的 ABEL—HDL、Altera 公司的 AHDL、Microsim 公司的 DSL,等等。这些硬件描述语言各具特色,普遍收到了优于传统方法的实际效果,语言本身也在应用中不断地发展和完善,逐步成为描述硬件电路的重要手段。

2. VHDL

20 世纪 80 年代初美国国防部为其超高速集成电路计划(VHSIC)提出了硬件描述语言 VHDL (Very-High-Speed Integrated Circuit Hardware Description Language),作为该计划的标准 HDL 格式。VHDL 主要用于描述数字系统的结构、行为、功能和接口,在使用中,很好地体现了标准化的威力,因而逐步得到推广。目前在硬件描述语言中,较为常用的除 VHDL 外还有起源于集成电路的设计 Verilog HDL。

VHDL 具有强大的功能、覆盖面广、描述能力强,可用于从门级、电路级直至系统级的描述、仿真和综合。具有丰富的仿真语句和库函数,随时可对系统进行仿真模拟;与原理图设计相比 VHDL 有良好的可读性,容易被读者理解,易于文件的归档;VHDL 支持对大规模设计的分解和已有设计的再利用,可由多人、多项目组来共同承担和完成,标准化的规则和风格,为设计的再利用提供了有力的支持。

3. “自顶向下”的设计方法

随着硬件描述语言 HDL 和 EDA 工具的发展出现了一种(Top-Down)的全新设计方法,就是设计者以系统功能和性能为出发点,接着对系统进行功能划分,形成若干子系统,由子系统再细化为不同的功能模块,再向下到单元电路、元器件。

采用这种方法进行系统设计是从系统顶层开始的,利用仿真等手段,在设计初期就掌握所实现系统的性能状况,并可以做相应的设计方案调整。方案没有问题可向下继续进行,并随时可以根据需要加以调整,直到设计完成,这种设计方法是在上一级方案满足设计要求的前提下往下进行的,有力地保证了设计结果的正确性,缩短了设计周期,提高了设计的一次成功率。设计规模越大,这种设计方法的优势越明显。

这种设计方法的主要设计文件使用 HDL 语言编写源程序,用 HDL 语言编写的源程序作为归档文件有很多好处:一是资料量小,便于保存;二是可继承性好;三是阅读方便。可编程

逻辑器件的设计流程一般包括设计准备、设计输入、功能仿真、设计处理、时序仿真和器件编程和测试七个步骤。

1.3 EDA、VHDL 及其应用

可编程逻辑器件是逻辑器件家族中发展最快的一类器件,可编程逻辑器件的出现及其所具有的 VHDL 的设计方法使得其产品开发周期缩短、现场灵活性好、开发风险变小。随着工艺、技术及市场的不断发展,PLD 产品的价格将越来越便宜、集成度越来越高、速度越来越快,再加上其设计开发采用符合国际标准的、功能强大的通用性 EDA 工具,可编程逻辑器件的应用前景将愈来愈广阔,FPGA、CPLD 以其不可替代的地位,将越来越受到业内人士的关注。

1.3.1 EDA

随着可编程逻辑器件规模的不断扩大,对芯片功能的二次开发就越来越依赖于专用的手段和工具才能完成,EDA 顺应了这种需求,是人们利用集计算机图形学、拓扑逻辑学、计算数学、人工智能等多种计算机应用学科的最新成果开发而成的一整套软件工具,可进行芯片逻辑功能的设计、仿真、时序分析、逻辑综合等,极大地改善了开发环境。应用这种软件工具及其支持的硬件描述语言从事电子系统的设计,打破了软硬件之间最后的屏障,使软硬件工程师们有了真正的共同语言。

在 EDA 的设计过程中,除了硬件的行为和功能描述外,其他设计过程都可以用计算机来自动完成。它可以大大节省人力和物力,缩短研制周期,从而增强了设计的实时性,因而 EDA 设计方法得到了广泛的应用。

20 世纪 80 年代末至 90 年代初,EDA 技术发展的主要特点是采用自顶向下的设计方法,以最终实现可靠的硬件系统为目标,因而配备了系统设计自动化的全部工具。比如硬件描述语言平台 VHDL、Verilog HDL、ABEL - HDL 等各种软件工具提供了各种输入方法,主要有硬件描述语言文本输入法、原理图输入法和波形输入法等(见图 1.1 和图 1.2);以并行设计工程(CE)方式和系统级目标设计方法作为支持,并提供高性能的逻辑综合、优化和仿真模拟工具。

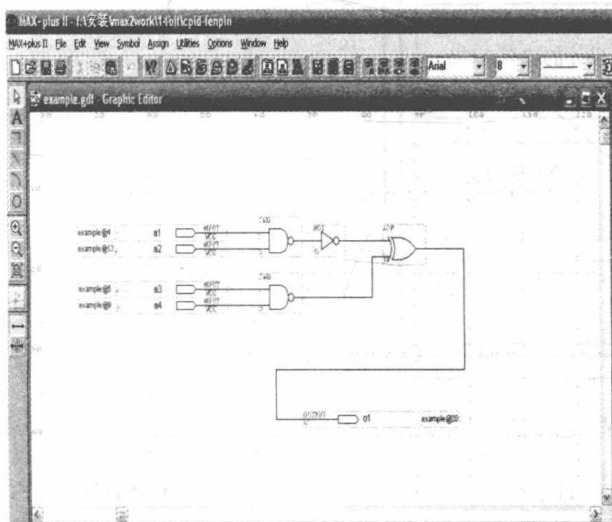


图 1.1 MAX+PLUS II 原理图输入界面

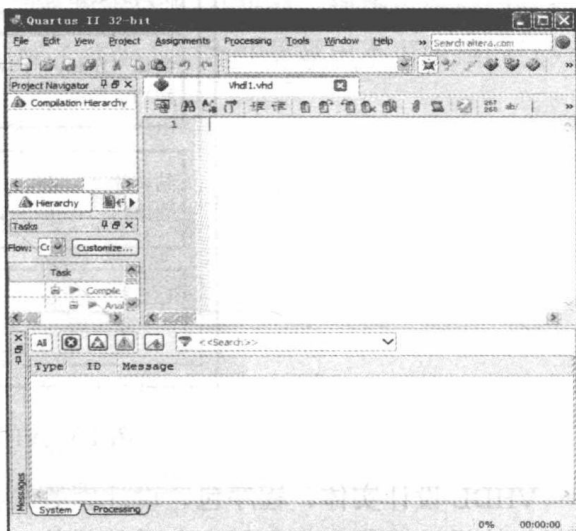


图 1.2 Quartus II VHDL 设计界面

硬件电路设计的软件化是电路设计的发展趋势，VHDL 是实现硬件电路设计软件化的重要语言工具，它实现了将数字系统的设计直接面向用户，根据系统的行为和功能要求，自上而下地完成相应的描述、综合、优化、仿真和验证，直到生成器件。

1.3.2 VHDL 特点

自 1962 年 Iverson 提出硬件描述语言 (HDL) 以来，出现了很多种硬件描述语言，但由于这些语言与硬件产品密切相关且语法不统一，造成了信息交流和设备维护的困难。VHDL 也是其中的一种硬件描述语言，原先由美国国防部制定，作为各合同商之间提交复杂电路设计文档的一种标准方案，1987 年被采纳为 IEEE1076 标准，并于 1993 年更新为 IEEE1164 标准。VHDL 用于描述、组织电路的结构行为，它克服了原理图输入在设计门级电路方面的局限性（一般认为瓶颈为 1 万左右，门级电路再增加，则测试和修改便难以进行）。

VHDL 语言具有很强的行为描述能力，可以实现各种级别（系统级、算法级、寄存器级、逻辑级、门级）的逻辑设计、仿真验证、时序分析、逻辑综合等。它所具有的丰富的仿真语句和库函数，使得在任何大系统的设计早期，就能检查设计系统功能的可行性，随时对设计进行仿真模拟。VHDL 采用基于库 (Library) 的设计方法。在设计过程中，可以建立各种可再次利用的模块，一个大规模的硬件电路设计往往不可能从门级电路开始一步步地进行设计，而是一些模块的累加。这些模块可以是一些标准库，也可以是预先设计或以前设计的模块，将这些模块存于库中，就可以在以后的设计中反复使用。这种设计方法可以大大地减少设计工作量，降低设计周期。

一个相对完整的 VHDL 设计由以下几个部分组成：库、程序包、实体、结构体、配置，如图 1.3 所示。

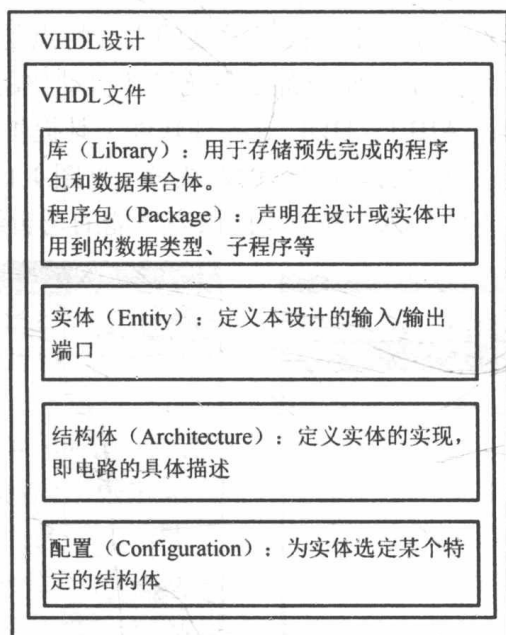


图 1.3 VHDL 程序结构框图

VHDL 设计实体、程序包、设计库等结构为设计任务分解和并行工作提供了有利的支持，便于加快设计流程和方便组合各功能模块，加快开发周期，降低开发成本。VHDL 具有相对

独立性,设计者没有必要熟悉硬件的结构,也不必了解最终设计实现的目标器件是什么,即可进行独立的设计,它良好的可移植性可兼容许多 EDA 软件平台和逻辑仿真与综合工具。

1.3.3 VHDL 设计流程及应用

VHDL 是用来描述从抽象到具体硬件级别的工业标准语言,并已成为一种通用的硬件设计交换媒介。计算机辅助工程软件的供应商已把 VHDL 作为其 CAD 或 EDA 软件输入与输出的标准,例如 SYNOPSYS、ALTERA、CA-DENCE、VIEWLOGIC 等 EDA 厂商均提供了 VHDL 的编译器,并在其仿真工具、综合工具和布图工具中提供了对 VHDL 的支持。VHDL 的设计流程如图 1.4 所示。

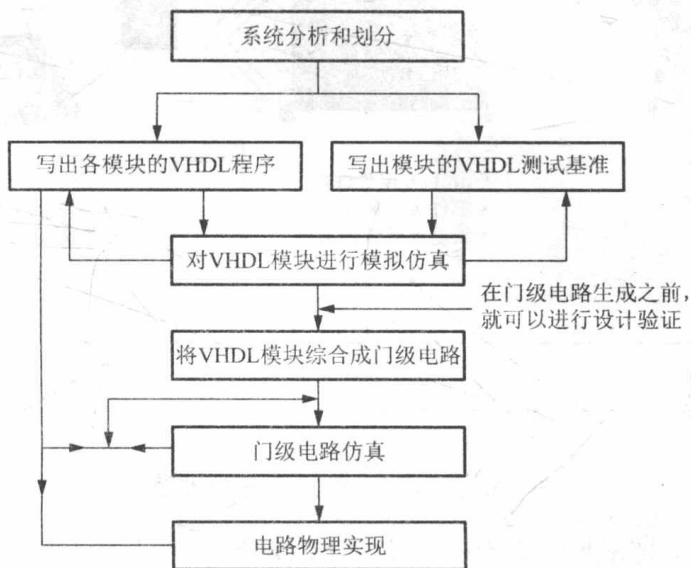


图 1.4 VHDL 的设计流程

在过去的半个世纪里,集成电路技术的进步不断刷新全球电子信息产业的形态,五光十色的新产品、新应用也改变了人类的生活方式。市场的需求使得电子产品的市场寿命周期日益缩短,与此同时,工艺技术的升级也让产品的开发成本呈几何级数上升。市场急需一种能够降低研发成本、缩短开发周期并具有设计灵活性的产品。在此背景下,FPGA(现场可编程门阵列)产业逐渐壮大,广泛应用在通信、工业、航空等领域(见图 1.5),并显露出不可阻挡的气势。

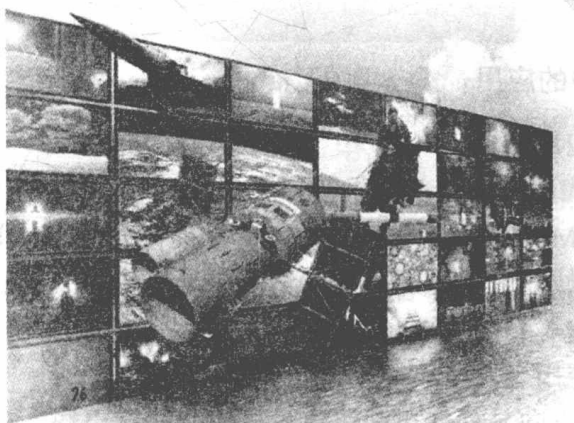


图 1.5 FPGA 在通信、工业、航空等领域应用

1. 在 ASIC 设计中的应用

可编程逻辑器件是在专用型集成电路（Application Specific Integrated Circuit, ASIC）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，设计完成后，如果不能满足要求，就得重新设计再进行验证。这样就使得设计开发周期变长，产品上市时间难以保证，大大增加了产品的开发费用。而 FPGA/CPLD 芯片是特殊的 ASIC 芯片，它们除具有 ASIC 的特点之外，还具有自身的优势。ASIC 向 FPGA 发展的过程如图 1.6 所示。

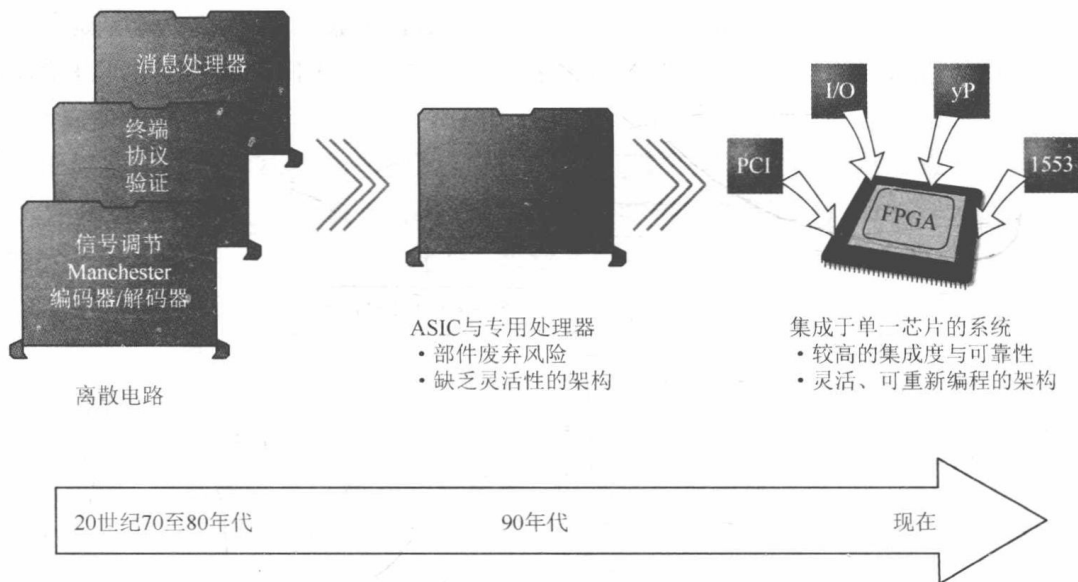


图 1.6 ASIC 向 FPGA 发展的过程

目前，ASIC 的容量越来越大，密度已达到平均每平方英寸 100 万个门电路。但随着密度的不断提高，芯片则受到引脚的限制，片上芯片虽然很多，但接入内核的引脚数目却是有限的。而选用 FPGA/CPLD 则不存在这样的限制，因为现在可达到的金属层数目增强了产品的优势，FPGA/CPLD 芯片的规模越来越大，其单片逻辑门数已达到上百万门，实现的功能也越来越强，同时可以实现系统集成。

另外，与 ASIC 相比，可编程逻辑器件研制周期较短，先期开发费用较低，也没有最少订购数量的限制，所有这一切简化了库存管理。随着每个门电路成本的降低和每个器件中门电路数量的增加，可编程逻辑器件正在大举打入传统的门阵列领域，并已有少量的打入了标准单元 ASIC 的领域。

2. 在电子技术领域中的应用

1) 在微机系统中的应用

FPGA/CPLD 可以取代现有的全部微机接口芯片，实现微机系统中的地址译码、总线控制、中断及 DMA 控制、DRAM 管理和 I/O 接口电路等功能。利用 CPLD 和 FPGA 可以把多个微机系统的功能集成在同一块芯片中，即进行所谓的“功能集成”。图 1.7 为基于 FPGA 的微机接口及应用综合实验平台。

2) 在通信领域中的应用

现代通信系统的发展方向是功能更强、体积更小、速度更快、功耗更低。FPGA/CPLD 在集成度、功能和速度上的优势正好满足通信系统的这些要求。所以现在无论是民用的移动电话、