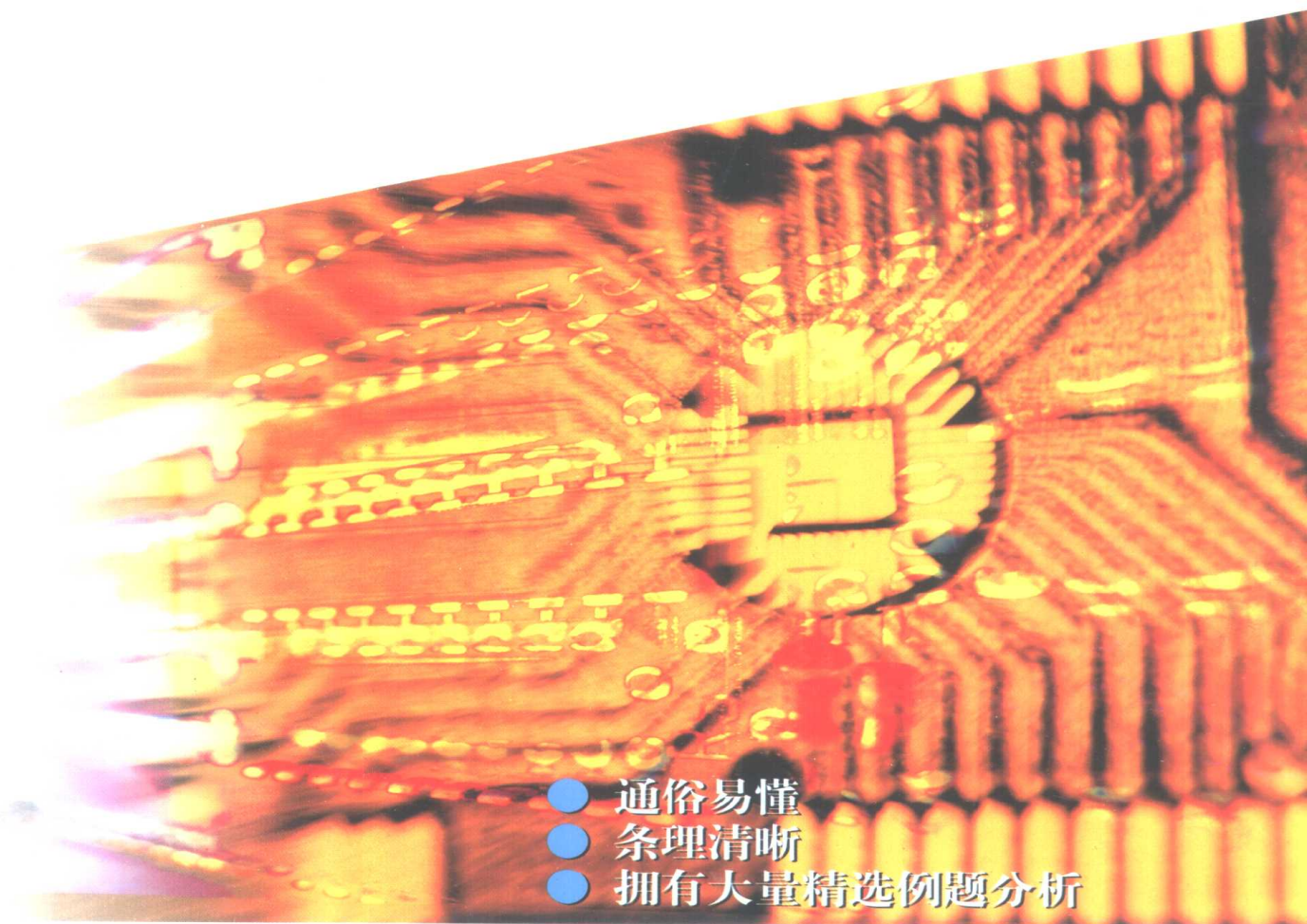


高等院校电子技术教材 (2)

集成电路设计 VHDL 教程

赵俊超等 编写



- 通俗易懂
- 条理清晰
- 拥有大量精选例题分析



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

高等院校电子技术教材

集成电路设计 VHDL 教程

赵俊超等 编写

本版特点:

作者运用 219 个程序实例

详尽介绍了使用 VHDL 语言进行 EDA 描述
数控知识系统、全面, 实例分析标准、完整

读者对象:

高校电子、电气、自动控制等专业师生

电子类研究生和科研人员

微电子类从业人员和工程师的参考书



北京希望电子出版社

Beijing Hope Electronic Press

www.bhp.com.cn

内 容 简 介

本书是作者多年使用 VHDL 进行大规模电子自动化 (EDA) 集成电路设计和教学经验的总结。作者共使用了 **219** 个实例程序 (其中 153 个完整程序) 来讨论 VHDL 用于硬件系统设计的各个方面。本书还引入了很多当前硬件设计领域流行的应用, 所以既具有实用性, 也具有时效性。

VHDL 是 IEEE 标准之一, 并且已经成为我国高校电子、电气和微电子专业高年级学生的必修课程。该书的宗旨是帮助硬件设计工程师学习如何用 VHDL 进行设计建模。对硬件系统设计的每个层次的 VHDL 描述加以指导, 包括从设计方法到门级设计、实现等方面, 告诉设计者如何简洁、高效、正确地写硬件系统设计的 VHDL 描述。

全书由 18 章组成, 内容包括 VHDL 程序设计基础、程序的基本结构; VHDL 数据、属性、表达式和主要描述语句; 信号与延迟; VHDL 模型的基本结构、子程序和连接配置; 基本组合道路和时序电路的 VHDL 模型; 有限状态机; 系统仿真; 逻辑综合与设计实现; 以及系统设计实例等。

本书主要面向高校电子、电气、自动控制等专业师生, 以及具有一定逻辑电路设计基础与程序设计经验的电子类研究生和科研人员。既可作为系统学习硬件设计的教材, 也可作为微电子类设计从业人员和工程师的参考书。

有关该书的技术咨询, 请与作者赵俊超 (william-zhang@vip.sina.com) 联系。

说明: 有关书中实例的源代码, 请从 www.x_br.com 下载 3891.zip。

系 列 书 名: 高等院校电子技术教材 (2)

书 名: 集成电路设计 VHDL 教程

总 策 划: 北京希望电子出版社

文 本 著 者: 赵俊超等 编写

责 任 编 辑: 但明天

出版、发行者: 北京希望电子出版社

地 址: 北京市海淀区知春路 63 号卫星大厦三层 (100080)

网 址: www.bhp.com.cn E-mail: lwm@bhp.com.cn

电 话: 010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921, 82610344 (发行)

010-82675588-202 (门市) 010-82675588-501, 82675588-201 (编辑部)

销 售: 各地新华书店、软件连锁店

排 版: 希望图书输出中心 全卫

文 本 印 刷 者: 北京东升印刷厂印刷

开 本 / 规 格: 787 毫米×1092 毫米 16 开本 28 印张 652 千字

版 次 / 印 次: 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

印 数: 1-4000 册

本 版 号: ISBN 7-900118-23-3

定 价: 35.00 元

说明: 凡我社产品如有缺损, 可执相关凭证与本社调换。

前言

随着电子技术的不断发展与进步,集成电路的设计方法学也在不断地更新。时至今日,传统的手工设计过程已经被先进的电子设计自动化(EDA, Electronic Design Automation)工具所替代。只有以硬件描述语言和逻辑综合为基础的自顶向下的电路设计方法才能满足日趋复杂的集成电路系统设计需求,才能缩短设计周期以满足社会对集成电路系统日益紧迫的需求。熟悉并掌握这些现代化设计理念和 EDA 工具,已成为集成电路和电子系统设计人员必备的一项基本素质。

现代的 EDA 设计工具,不无例外地以硬件描述语言(HDL, Hardware Description Language)作为其设计的输入。因此,掌握 EDA 工具首先必须学会 HDL 编程。

VHDL 是美国电气和电子工程师协会制定的标准硬件描述语言(IEEE 标准 1076),是世界上第一个标准化的 HDL 语言,它可用于数字电路与系统的描述、仿真和自动设计。美国国防部要求自 1988 年 9 月 30 日起,美国所有军方研制的专用集成电路(ASIC)必须用 VHDL 描述。同时,欧美大的集成电路设计生产厂家,如 Motorola 等都已经开始使用 VHDL。

本书的宗旨是帮助硬件设计工程师学习如何用 VHDL 更好地设计建模。本书对硬件系统设计的每个层次的 VHDL 描述都将加以指导,包括从设计方法到门级设计实现等各个方面的内容。本书不希望展示每种可能用到的 VHDL 结构,而是力图告诉设计者如何简洁、高效并且正确地编写硬件系统设计的 VHDL 描述。

本书按两个部分组织内容:第 1 到 9 章为基础篇,主要介绍了 VHDL 的基础知识,第 10-18 章为提高篇,介绍了从设计方法到设计工具等各层面的相关技术。

第 1 章是引言,介绍了 HDL 的基本概念。涉及到 VHDL 语言的产生、发展,VHDL 语言的特点,以及开发环境。

第 2 章 VHDL 语言程序的基本结构。VHDL 语言程序由实体和结构体组成,本章详细介绍了类属和端口说明、端口模式以及实体说明部分和实体语句部分。另外还介绍了结构体的命名、定义语句和并行处理语句,以及结构体的子结构设计方法。最后用比较的方法介绍了四种描述风格:行为描述、数据流描述、结构描述和混合描述。

第 3 章 VHDL 语言的数据格式和表达式。本章详细介绍了 VHDL 语言的四类数据对象和四类数据类型,还介绍了不同类型的数据对象之间的类型转换方法以及 VHDL 语言中的词法单元。最后,介绍了 VHDL 语言的表达式与运算符,以及运算符的优先级别。

第 4 章介绍了 VHDL 语言中的两类描述语句:顺序语句和并行语句。顺序语句部分包括变量赋值语句、信号赋值语句、IF 语句、CASE 语句、LOOP 语句、NEXT 语句、EXIT 语句、断言语句、过程调用语句、RETURN 语句、NULL 语句和 REPORT 语句。并行语句部分包括:进程语句、WAIT 语句、BLOCK 语句、并行过程调用语句、并行断言语句、并行信号赋值语句、信号代入语句、元件例化语句、生成语句、参数传递语句和端口映射语句。本章最后还介绍了 VHDL 语言中的命名规则和注解的标注。

第 5 章介绍了 VHDL 语言预定义属性。一共有五类属性:数值类属性、函数类属性、信号类属性、类型类属性和范围类属性。此外,本章还介绍了 ATTRIBUTE 语句,最后总

结了检查信号建立时间和保持时间的方法。

第6章的主要内容是信号与延迟。首先介绍了信号驱动源,然后是传输延迟和惯性延迟,接下来介绍了信号驱动源上延迟的作用,最后引出了 VHDL 中模拟周期和 Δ 延迟这两个很重要的概念。

第7章 VHDL 语言模型的基本结构。本章介绍了大型任务的层次化设计,重点介绍了层次化设计不同阶段中用到的设计库、程序包等,最后还介绍了元件例化的内容。

第8章详细介绍了 VHDL 描述大型设计的配置方法。在本章首先介绍了用默认连接、默认配置来把结构体连接到实体上。在配置说明部分,涉及到两个层次上的配置,即低级配置和实体-结构体对的配置;在这部分,还介绍了有关端口映射的内容。

第9章讨论了 VHDL 语言中各种子程序和它们的用途。本章内容包括函数和过程两大部分,详细介绍了过程与函数定义和使用上的异同。最后还介绍了子程序重载和运算符重载。

第10章介绍了一些有关数字电路设计的理论以及其他一些相关的内容。本章内容是复杂数字系统设计的基础,首先介绍了电路设计中的重要概念,然后介绍了可编程逻辑器件的基础知识以及开发过程。在本章后面部分,介绍了集成电路的设计与综合、自顶向下与自底向上的设计方式等设计理论,还重点讨论了三种设计方法:组合逻辑电路设计、时序逻辑电路设计和数字集成系统的行为综合。

第11章介绍了基本组合逻辑电路的 VHDL 模型的建模方法。本章内容包括:逻辑门电路、三态缓冲器、多路选择器、译码器、编码器、比较器、移位器、加法器、乘法器、求补器、算术逻辑单元 ALU 和可编程逻辑阵列 PLA 等。

第12章基本时序逻辑电路的 VHDL 模型。主要介绍了数字系统中常用的基本时序逻辑电路,包括:锁存器、触发器、寄存器以及计数器的概念、特点和分类,并给出典型时序逻辑电路的 VHDL 模型,着重区分同步时序逻辑电路和异步时序逻辑电路。

第13章详细介绍有限状态机的概念。本章首先介绍了有限状态机的概念与分类,然后具体论述了状态转换图和状态转换表的建立方法和流程。在此基础上,详细讨论了有限状态机的 VHDL 建模方法,有限状态机状态编码方式,并结合具体实例详细介绍了有限状态机的五种 VHDL 建模风格,并且给出了详尽的 VHDL 代码。

第14章从系统设计角度的系统层次化设计方法和自顶向下的系统设计方法角度,主要介绍了系统设计中的层次划分和层次化 VHDL 建模以及系统设计树的生成和系统原理块流程图的生成,并以与 80C51 指令兼容微处理器为例详细介绍了系统自顶向下的设计流程。

第15章主要介绍用 VHDL 描述的数字系统的行为仿真方法。本章分别采用实例说明了组合电路系统和时序电路系统的行为仿真方法和测试文件的生成方法,并且给出了仿真结果。最后总结出了两种基本的测试程序生成方式:表格测试程序设计和 I/O 文件测试程序设计。

第16章从逻辑综合的概念、原理入手,介绍了逻辑综合过程。本章以 Synopsys 公司的逻辑综合工具 Design Compiler 为例讨论了逻辑综合库的设置和综合约束条件的设置。然后着重介绍了 CPLD 和 FPGA 基本原理和结构,并在此基础上给出若干实例,详细讨论了面向 CPLD 的设计实现技术和面向 FPGA 的设计实现技术。在讨论设计实现时,本

章以 Cypress 公司的 CPLD 和 FPGA 为例, 以 Warp 综合工具为设计实现的软件平台。

第 17 章介绍 Altera 公司的软件系统 MAX+Plus II。本章首先介绍了 MAX+Plus II 软件系统的特点以及安装方法, 然后详细介绍了使用 MAX+Plus II 软件系统实现设计的流程, 对于每个流程都给出了详细的步骤和操作方法, 并配以图示, 便于脱离电脑和 MAX+Plus II 环境学习。

第 18 章向读者介绍 Xilinx Foundation Series ISE 3.1i 系统软件的特点及其提供的各种工具的基本使用方法。本章首先介绍了 Xilinx Foundation Series ISE 3.1i 系统软件提供的各种应用软件工具的功能和使用方法, 然后以一个设计实例来详细介绍各种软件的简单使用步骤。

本书是作者多年使用 VHDL 进行大规模集成电路设计的经验总结, 从清晰的概念入手, 涵盖了 VHDL 用于硬件系统设计的方方面面, 可以作为系统学习硬件设计的教材。

本书的使用, 主要是面向具有一定的逻辑电路设计基础与程序设计经验的电子类高年级本科生或研究生, 本书也可以作为电子类设计从业人员的工程参考书。对于已有一定基础的读者可以跳过基础篇直接到提高篇, 选择性地阅读。

本书主要作者: 赵俊超、陈维良、张威。参加本书编写工作的有孙海涛、刘军、丁斌、王敏、刘永刚、崔黎明、吴奕、李涛、杨超、栾东美、张檀、高硕、陈健、李德文、赵钊、邢建文、董良、成维、王卉、王剑、许宇斌、宋华、张振、王韬、吴超、杨波。由于篇幅有限, 时间仓促, 错误和疏漏之处在所难免, 恳请广大读者批评指正。

作者

2002 年于清华大学

名师执笔、实践第一、教与学并用



CX-3277
定价: 21.00 元
ISBN:030058143



CX-2681
定价: 16.00 元
ISBN:030062981



CX-2682
定价: 24.00 元
ISBN:030064690



CX-2745
定价: 19.00 元
ISBN:030064674



CX-2746
定价: 14.00 元
ISBN:030064682



CX-2747
定价: 23.00 元
ISBN:030066553



CX-2934
定价: 15.00 元
ISBN:030058151



CX-3250
定价: 26.00 元
ISBN:03006304x



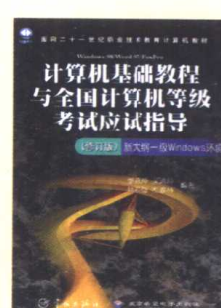
CX-2749
定价: 23.00 元
ISBN:030063023



CX-2750
定价: 23.00 元
ISBN:801442768



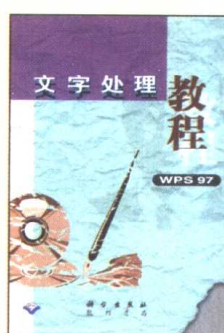
CX-2752
定价: 25.00 元
ISBN:80144275X



CX-2751
定价: 24.00 元
ISBN:801441540



CX-2683
定价: 15.00 元
ISBN:030064623



CX-2684
定价: 18.50 元
ISBN:030071344



CX-2755
定价: 27.00 元
ISBN:030066537



CX-2756
定价: 19.00 元
ISBN:030066529



CX-2757
定价: 16.00 元
ISBN:030062965



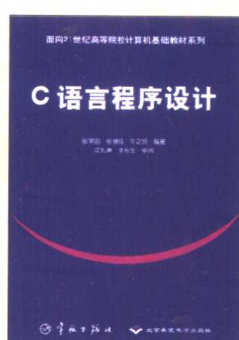
CX-2758
定价: 18.00 元
ISBN:030059972



CX-3813
定价: 20.00 元
ISBN:801444329



CX-3814
定价: 16.00 元
ISBN:801444337



CX-3815
定价: 21.00 元
ISBN:801444345



CX-3816
定价: 21.00 元
ISBN:801444353



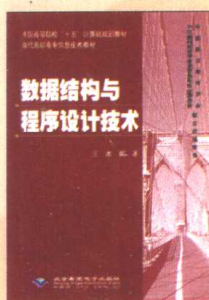
CX-3817
定价: 23.00 元
ISBN:801444361



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀知春路 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱北京希望电子出版社 (邮编 100080)
联系电话: 010-82675588(总机) 传真: 010-62520573

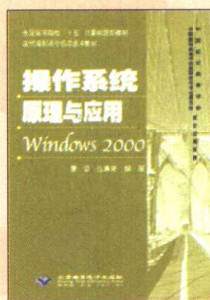
名师执笔、实践第一、教与学并用



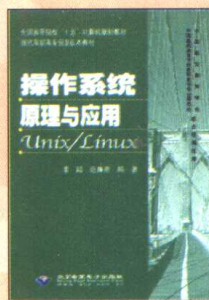
CX-3821
定价: 23.00 元
ISBN: 7900101594



CX-3822
定价: 20.00 元
ISBN: 7900101608



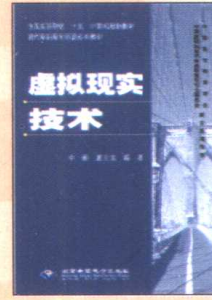
CX-3823
定价: 23.00 元
ISBN: 7900101616



CX-3824
定价: 20.00 元
ISBN: 7900101624



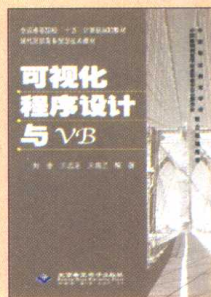
CX-3825
定价: 23.00 元
ISBN: 7900101632



CX-3826
定价: 21.00 元
ISBN: 7900101640



CX-3827
定价: 20.00 元
ISBN: 7900101659



CX-3828
定价: 24.00 元
ISBN: 7900101780



CX-3832
定价: 25.00 元
ISBN: 7900101772



CX-3745
定价: 25.00 元(估)
ISBN: 900101667



CX-3803
定价: 20.00 元(估)
ISBN: 900101675



CX-3704
定价: 20.00 元(估)
ISBN: 900101683



CX-3805
定价: 22.00 元(估)
ISBN: 900101691



CX-3806
定价: 32.00 元(估)
ISBN: 900101705



CX-3807
定价: 26.00 元(估)
ISBN: 900101713



CX-3808
定价: 23.00 元(估)
ISBN: 900101721



CX-3809
定价: 21.00 元(估)
ISBN: 90010173X



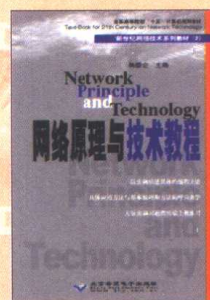
CX-3810
定价: 23.00 元(估)
ISBN: 900101748



CX-3811
定价: 22.00 元(估)
ISBN: 900101756



CX-3812
定价: 21.00 元(估)
ISBN: 900101764



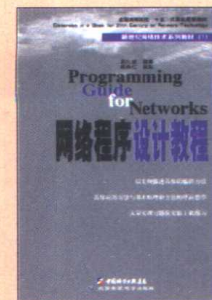
CX-3818
定价: 27.00 元
ISBN: 7900101500



CX-3819
定价: 29.00 元
ISBN: 7900101519



CX-3820
定价: 22.00 元
ISBN: 7900101527



CX-3612
定价: 23.00 元
ISBN: 7900088156



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀迎春路 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱北京希望电子出版社 (邮编 100080)
联系电话: 010-82675588(总机) 传真: 010-62520573

目 录

第 1 章 引言	1	3.3.2 复合类型	24
1.1 硬件描述语言	1	3.3.3 存取类型	26
1.2 VHDL 语言的产生及发展	2	3.3.4 文件类型	26
1.3 VHDL 语言的特点	3	3.3.5 其他类型	27
1.4 VHDL 语言的开发环境	3	3.3.6 子类型	28
1.5 小结	5	3.4 类型转换	29
第 2 章 VHDL 程序基本结构	6	3.4.1 用类型标记实现类型转换	29
2.1 VHDL 程序的基本单元	6	3.4.2 用户创建的类型转换	30
2.2 设计实体	7	3.4.3 数据类型的限定	31
2.2.1 类属和端口说明	8	3.4.4 IEEE 标准数据类型	31
2.2.2 端口模式	9	3.5 词法单元	32
2.2.3 实体说明部分	9	3.5.1 注释	32
2.2.4 实体语句部分	10	3.5.2 数字	32
2.3 设计结构体	10	3.5.3 字符	33
2.3.1 结构体命名	11	3.5.4 字符串	33
2.3.2 定义语句	11	3.5.5 位串	33
2.3.3 并行处理语句	11	3.6 VHDL 表达式与运算符	33
2.3.4 结构体的子结构设计方法	12	3.6.1 逻辑运算符	34
2.4 描述风格	12	3.6.2 算术运算符	34
2.4.1 行为描述	13	3.6.3 关系运算符	35
2.4.2 数据流描述	14	3.6.4 并置运算符	36
2.4.3 结构描述	14	3.6.5 运算符的优先级	36
2.4.4 混合描述	16	3.7 小结	37
2.5 小结	17	第 4 章 VHDL 主要描述语句	38
第 3 章 VHDL 数据和表达式	18	4.1 顺序语句	38
3.1 VHDL 标识符	18	4.1.1 变量赋值语句	38
3.1.1 短标识符	18	4.1.2 信号赋值语句	39
3.1.2 扩展标识符	18	4.1.3 IF 语句	39
3.2 VHDL 数据对象	19	4.1.4 CASE 语句	40
3.2.1 常量	20	4.1.5 LOOP 语句	41
3.2.2 变量	20	4.1.6 NEXT 语句	43
3.2.3 信号	21	4.1.7 EXIT 语句	43
3.2.4 文件	22	4.1.8 断言语句	44
3.3 VHDL 数据类型	22	4.1.9 过程调用语句	44
3.3.1 标量类型	23	4.1.10 RETURN 语句	45

4.1.11 NULL 语句.....	45	作用.....	90
4.1.12 REPORT 语句.....	45	6.3.2 信号驱动源上惯性延迟的作用	91
4.2 并行语句.....	46	6.3.3 信号驱动源上阈值惯性延迟的	
4.2.1 进程语句.....	46	作用.....	93
4.2.2 WAIT 语句.....	48	6.4 模拟周期.....	93
4.2.3 BLOCK 语句.....	50	6.5 δ 延迟.....	95
4.2.4 并行过程调用语句.....	52	6.6 小结.....	97
4.2.5 并行断言语句.....	53	第 7 章 VHDL 模型的基本结构.....	98
4.2.6 并行信号赋值语句.....	54	7.1 设计层次.....	98
4.2.7 信号代入语句.....	55	7.2 库.....	98
4.2.8 元件例化语句.....	57	7.2.1 STD 库.....	99
4.2.9 生成语句.....	59	7.2.2 WORK 库.....	99
4.2.10 参数传递语句.....	60	7.2.3 资源库.....	99
4.2.11 端口映射语句.....	62	7.2.4 USE 子句.....	99
4.3 命名规则和注解标注.....	62	7.3 程序包.....	100
4.4 小结.....	63	7.3.1 STANDARD 程序包.....	102
第 5 章 属性.....	64	7.3.2 TEXTIO 程序包.....	103
5.1 预定义属性.....	64	7.3.3 Std_Logic-1164 程序包.....	109
5.2 数值类属性.....	64	7.3.4 Numeric_Std 程序包.....	112
5.2.1 数值类属性.....	64	7.3.5 Numeric_Bit 程序包.....	112
5.2.2 数值类数组属性.....	66	7.4 元件例化.....	112
5.2.3 数值类块属性.....	68	7.4.1 用户构造.....	112
5.3 函数类属性.....	69	7.4.2 调用元件.....	119
5.3.1 函数类属性.....	69	7.5 小结.....	121
5.3.2 函数数组属性.....	71	第 8 章 配置.....	122
5.3.3 函数信号属性.....	73	8.1 默认连接.....	122
5.4 信号类属性.....	75	8.2 默认配置.....	122
5.5 类型类属性.....	80	8.3 配置说明.....	124
5.6 范围类属性.....	81	8.3.1 元件配置.....	125
5.7 ATTRIBUTE 语句.....	82	8.3.2 低级配置.....	126
5.8 用属性检查建立和保持时间.....	83	8.3.3 实体-结构体对的配置.....	127
5.9 小结.....	85	8.3.4 端口映射.....	128
第 6 章 信号与延迟.....	86	8.4 映射实体.....	129
6.1 信号驱动源.....	86	8.5 配置中的类属.....	132
6.2 延迟.....	87	8.5.1 结构体中的类属值.....	134
6.2.1 惯性延迟.....	88	8.5.2 配置中的类属.....	136
6.2.2 传输延迟.....	89	8.6 块的配置.....	139
6.3 信号驱动源上的延迟作用.....	90	8.7 结构体的配置.....	141
6.3.1 信号驱动源上传输延迟的		8.8 小结.....	141

第 9 章 子程序	142	11.6 比较器	203
9.1 过程	142	11.7 移位器	204
9.2 函数	144	11.8 运算器	205
9.2.1 转换函数	146	11.8.1 加法器 (Adder)	205
9.2.2 决断函数	151	11.8.2 乘法器 (Multiplier)	207
9.3 子程序重载	157	11.8.3 求补器	210
9.3.1 子程序重载	157	11.9 算术逻辑单元 ALU	210
9.3.2 运算符重载	158	11.10 可编程逻辑阵列 PLA	217
9.4 小结	161	11.11 小结	220
第 10 章 VHDL 程序设计基础	162	第 12 章 基本时序电路的 VHDL 模型	
10.1 设计中的重要概念	162		221
10.1.1 组合逻辑与时序逻辑	162	12.1 锁存器	221
10.1.2 锁存器、触发器和寄存器	164	12.1.1 电平锁存器	221
10.1.3 存储器	168	12.1.2 同步锁存器	222
10.2 可编程逻辑器件	169	12.1.3 异步锁存器	223
10.2.1 PLD 概述	169	12.2 触发器	224
10.2.2 CPLD 和 FPGA	171	12.2.1 D 触发器	224
10.2.3 开发 PLD	173	12.2.2 T 触发器	229
10.3 相关的设计理论	175	12.2.3 JK 触发器	232
10.3.1 集成电路的设计与综合	175	12.3 寄存器	235
10.3.2 层次化设计方式	177	12.3.1 通用寄存器	235
10.4 相关的设计方法	178	12.3.2 移位寄存器	236
10.4.1 组合电路设计	178	12.4 计数器	239
10.4.2 时序电路设计	186	12.4.1 同步计数器	239
10.4.3 数字集成系统的行为综合	191	12.4.2 异步计数器	240
10.5 小结	193	12.5 小结	241
第 11 章 基本组合电路的 VHDL 模型		第 13 章 有限状态机	242
.....	194	13.1 有限状态机概述	242
11.1 逻辑门	194	13.1.1 Moore 状态机功能描述	242
11.1.1 与门 (AND Gate)	194	13.1.2 Mealy 状态机功能描述	243
11.1.2 或门 (OR Gate)	195	13.1.3 有限状态机的选择	243
11.1.3 反相器 (Inverter)	196	13.2 构造状态表	243
11.1.4 与非门 (NAND Gate)	197	13.2.1 建立状态转换图	244
11.1.5 或非门 (NOR Gate)	198	13.2.2 建立状态转换表	246
11.1.6 异或门 (XOR Gate)	199	13.3 有限状态机的复位	247
11.2 缓冲器	200	13.3.1 有限状态机的同步复位	247
11.3 选择器	200	13.3.2 有限状态机的异步复位	248
11.4 译码器	201	13.4 建立有限状态机的 VHDL	
11.5 编码器	203	模型	248

13.4.1 有限状态机的描述风格.....	248	实例.....	363
13.4.2 有限状态机的描述实例.....	249	16.3 小结.....	369
13.5 有限状态机状态编码方式.....	259	第 17 章 MAX+PlusII 使用入门.....	370
13.5.1 组合译码式.....	259	17.1 MAX+PlusII 特点.....	370
13.5.2 一位有效式.....	261	17.2 MAX+PlusII 的安装.....	371
13.6 小结.....	263	17.2.1 推荐的系统配置.....	371
第 14 章 系统设计实例.....	264	17.2.2 MAX+Plus II 的安装.....	372
14.1 存储器设计.....	264	17.3 使用 MAX+Plus II 软件系统的	
14.1.1 只读存储器 ROM.....	264	设计流程.....	373
14.1.2 随机读取存储器 RAM.....	265	17.3.1 设计输入.....	374
14.1.3 堆栈(stack).....	268	17.3.2 设计处理.....	375
14.2 系统层次化设计.....	270	17.3.3 设计校验.....	376
14.2.1 系统层次设计方法.....	270	17.3.4 设计编程.....	376
14.2.2 系统模块划分方法.....	271	17.3.5 设计流程.....	377
14.2.3 系统层次化设计实例.....	271	17.4 逻辑设计的输入方法.....	378
14.3 自顶向下的系统设计方法.....	286	17.4.1 图形设计输入.....	378
14.3.1 Top-Down 分析方法.....	286	17.4.2 文本设计输入.....	383
14.3.2 OC51 微处理器设计实例.....	287	17.4.3 创建顶层图形设计文件.....	383
14.4 小结.....	312	17.4.4 层次显示.....	383
第 15 章 系统仿真.....	313	17.5 编译设计项目.....	384
15.1 系统仿真概述.....	313	17.5.1 准备编译.....	384
15.2 组合电路系统仿真.....	314	17.5.2 编译选项设置.....	385
15.3 时序电路系统仿真.....	316	17.5.3 运行编译器.....	389
15.4 测试激励设计方式.....	322	17.5.4 在底层图编辑器中观察试配	
15.5 小结.....	326	结果.....	389
第 16 章 逻辑综合与设计实现.....	327	17.5.5 引脚锁定.....	389
16.1 逻辑综合.....	327	17.6 设计项目仿真.....	390
16.1.1 逻辑综合的原理.....	327	17.7 定时分析.....	391
16.1.2 设置技术库.....	329	17.8 器件编程.....	394
16.1.3 逻辑综合的约束条件.....	330	17.9 小结.....	396
16.2 设计实现.....	331	第 18 章 Xilinx Foundation Series	
16.2.1 设计实现概述.....	332	ISE 3.1i 简介.....	397
16.2.2 面向 CPLD 器件的实现		18.1 Xilinx Foundation Series ISE 3.1i	
概述.....	332	特点介绍.....	397
16.2.3 面向 CPLD 器件的实现		18.2 Xilinx Foundation Series ISE 3.1i	
实例.....	333	的安装.....	398
16.2.4 面向 FPGA 器件的实现		18.3 Xilinx Foundation Series ISE 3.1i	
概述.....	361	软件系统工具综述.....	401
16.2.5 面向 FPGA 器件的实现		18.3.1 软件启动.....	401

18.3.2 项目管理	401	18.4.1 VHDL 设计输入	408
18.3.3 设计输入	403	18.4.2 模块功能验证	410
18.3.4 逻辑综合	403	18.4.3 顶层电路逻辑图输入	413
18.3.5 设计约束	405	18.4.4 设计实现	418
18.3.6 功能仿真	405	18.4.5 时序分析	418
18.3.7 设计实现	405	18.5 小结	421
18.3.8 报告文件	406	附录 A VHDL 保留字	422
18.3.9 时序仿真	407	附录 B VHDL 语法总结	425
18.3.10 器件编程	408	附录 C VHDL 英汉名词对照表	428
18.4 Xilinx Foundation Series ISE 3.1i		附录 D 部分 PLD 厂家主流芯片介绍 ..	437
使用实例	408		

第1章 引言

随着电子设计技术的飞速发展,专用集成电路 ASIC 和用户现场可编程门阵列(FPGA)的复杂度越来越高,数字通信、工业自动化控制等领域所用的数字电路及系统的复杂程度也越来越高。设计这样复杂的电路及系统已不再是简单的个人劳动,而需要综合许多专家的经验 and 知识才能够完成。在数字逻辑设计领域,迫切需要一种共同的工业标准来统一对数字逻辑电路及系统的描述,把专家们设计的各种常用数字逻辑电路和系统部件建成宏单元(Macrocell)或软核(Soft-Core)库供设计者引用,以减轻重复劳动,提高工作效率。VHDL 和 Verilog HDL 这两种工业标准的产生顺应了历史的潮流,因而得到了迅速的发展。

1.1 硬件描述语言

硬件描述语言 HDL 是一种用形式化方法来描述数字电路和设计数字逻辑系统的语言。数字逻辑电路设计者可以利用这种语言来描述自己的设计思想,然后利用电子设计自动化 EDA 工具进行仿真,再自动综合到门级电路,最后利用 ASIC 或 FPGA 实现其功能。目前,这种被称为高层次设计(High-Level-Design)的方法已经被广泛采用。

硬件描述语言发展至今已有二十年的历史,并成功地应用于设计的各个阶段:仿真、验证和综合等。到 80 年代,已经出现了上百种硬件描述语言,它们对设计自动化起到了极大的促进作用。但是,这些语言一般各自面向特定的设计领域与层次,而且众多的语言使用户无所适从,因此,急需一种面向设计的多领域、多层次并得到普遍认同的标准硬件描述语言。进入 80 年代后期,硬件描述语言向着标准化的方向发展。最终,VHDL 和 Verilog HDL 语言适应了这种趋势的要求,先后成为 IEEE 标准。

比起传统的原理图设计方法来说,HDL 有许多优点,主要有如下几点:

首先,用 HDL 设计电路能够获得非常抽象的描述。设计者不用选择特定的制造工艺就能写出电路的寄存器传输级(RTL, Register Transfer Level)描述。逻辑综合工具能自动将 RTL 描述转换成任何一种制造工艺。如果出现了新工艺,设计者不用重新设计电路。当使用新工艺时,他们需要做的只是简单地把电路的 RTL 描述输入到逻辑综合工具中,就能产生一个新的门级网表。逻辑综合工具将针对新工艺自动地进行电路面积和时序的优化。

其次,用 HDL 描述电路设计,在设计的前期就可以完成电路功能级的验证。由于设计者工作在 RTL 级,他们可以不断地优化和修改 RTL 描述,直到满足所需要的功能为止。这时能够发现并改进设计中的绝大部分错误。在设计晚期的门级网表或物理版图中出现功能性错误的概率已经非常小,这样可以非常显著地缩短设计周期。

此外,用 HDL 设计电路类似于计算机编程。带有注解的文字性描述更有利于电路的开发与调试。比起门级原理图来说,HDL 还提供了非常简明的设计表达。在非常复杂的设计中,门级原理图几乎是不可理解的。

最近 10 年来,HDL 在逻辑电路设计中已经得到了广泛应用。工程管理者不再面临在设计中是否使用硬件描述语言的困境,相反,他们关心的是选用哪种语言能更好地与设计

环境相结合。使用硬件描述语言,设计者能够更好地从功能和行为上表述自己的设计,而且还可以加上详细的注解,以便在以后的设计中重复使用。在具体设计之前,通过抽象的功能描述,可以找到更灵活的系统结构,并发现设计的瓶颈所在。

1.2 VHDL 语言的产生及发展

VHDL (Very High Speed Integrated Circuit Hardware Description Language, 超高速集成电路硬件描述语言)发展的社会根源是这样的:美国国防部电子系统项目有众多的承包公司,各公司技术线路不一致,许多产品不兼容,他们使用各自的设计语言,使得承包公司甲的设计不能被承包公司乙重复利用,这就造成了信息交换困难和设计维护困难。为了解决这个问题,降低开发费用,避免重复设计,美国国防部为他们的超高速集成电路计划 VHSIC (very high speed integrated circuit) 提出了一种硬件描述语言 VHDL (VHSIC description language),这个任务交给了 TI 公司、IBM 公司和 Intermetrics 公司。大家期望 VHDL 功能强大、严格、可读性好。政府要求各公司的合同都用它来描述,以避免产生歧义。

由政府牵头,VHDL 工作小组于 1981 年 6 月成立,提出了一个满足电子设计各种要求的能够作为工业标准的 HDL。1983 年第三季度,由 TI 公司、IBM 公司和 Intermetrics 公司签约,组成开发小组,工作任务是提出语言版本和开发软件环境。1986 年 IEEE 标准化组织开始工作,讨论 VHDL 语言标准,历时一年多。1987 年 12 月 IEEE 接受 VHDL 为标准 HDL,这就是今天我们所了解的 IEEE Std 1076-1987 [LRM87]。1993 年 VHDL 重新修订,增加了一些功能,新的标准版本记作 IEEE Std 1076-1993 [LRM93]。严格地说,VHDL'93 和 VHDL'87 并不完全兼容(例如,增加了一些保留字并删去了某些属性),但是,对 VHDL'87 的源码只做了少许简单的修改就可以成为合法的 VHDL'93 代码[BFMR93]。

1988 年 9 月 30 日之后,美国国防部实施新的技术标准,要求电子系统开发商的合同文件一律采用 VHDL 文档,于是第一个官方 VHDL 标准得到推广、实施和普及。VHDL 逐渐演变为工业标准的过程,如表 1.1 所示。

表 1.1 VHDL 语言的发展

年	第一季度	第二季度	第三季度	第四季度
1983	美国国防部提出要求	和 TI、IBM 及 Intermetrics 公司签约	开始工作	
1984	V2.0		V5.0	V6.0
1985			V7.2	IEEE 开始标准化工作
1986	VHDL 分析和标准化小组第一次会议	语言参考手册第一次手稿		语言参考手册第二次手稿 1076/A
1987	评审 1076/A	语言参考手册 1076/B	反对票的出现	IEEE Std 1076-1987 通过
1992				再次投票通过
1993		第二次投票通过		IEEE Std 1076-1993 通过

1.3 VHDL 语言的特点

VHDL 和 Verilog HDL 语言都是用于逻辑设计的硬件描述语言,并且都成为 IEEE 标准。它们作为描述硬件电路的语言,具有一些共同的特点:能形式化地抽象表示电路的结构和行为;支持逻辑设计层次与领域的描述;可借用高级语言的精巧结构来简化电路的描述;具有电路仿真与验证机制以保证设计的正确性;支持电路描述由高层到低层的综合转换;硬件描述与实现工艺无关;便于文档管理;易于理解和设计再利用。

但是 VHDL 和 Verilog HDL 又各有其特点。我们知道,电路设计的表示可分为不同的层次,一般有 5 个抽象层次:系统级(System Level)、算法级(Algorithmic Level)、寄存器传输级(Register Transfer Level)、门级(Gate Level)和电路级(Circuit Level),VHDL 语言的抽象建模范围可以覆盖从最抽象的系统级一直到最精确的门级,而 Verilog HDL 语言的描述范围则一直可以到电路级。和 Verilog HDL 语言相比,VHDL 语言在门级电路描述方面不如 Verilog HDL 语言,但在系统级抽象方面要比 Verilog HDL 语言强,所以对于大型、特大型的系統级设计,使用 VHDL 语言最为合适。

总之,采用 VHDL 语言设计复杂数字电路的方法具有很多优点,下面将简要介绍。

VHDL 语言的设计技术齐全、方法灵活、支持广泛。VHDL 语言可以支持自顶向下(Top Down)和基于库(Library-Based)的设计方法,而且还支持同步电路、异步电路、FPGA 以及其他随机电路的设计。其范围之广是其它 HDL 语言所不能比拟的。另外,由于 VHDL 语言早在 1987 年 12 月已作为 IEEE Std 1076-1987 标准公开发布。因此,目前大多数 EDA 工具几乎在不同程度上都支持 VHDL 语言,这样为 VHDL 语言进一步推广和应用创造了良好的环境。由于 VHDL 语言已作为工业标准,这样设计成果便于复用和交流,反过来就能进一步推动 VHDL 语言的推广及完善。还有,VHDL 语言的语法比较严格,给阅读和使用都带来了极大的好处。

VHDL 语言的系統硬件描述能力很强。VHDL 语言具有多层次描述系統硬件功能的能力,可以从系統级到门级电路,而且高层次的行为描述可以与低层次的 RTL 描述混合使用。VHDL 在描述数字系統时,可以使用前后一致的语义和语法跨越多个层次,并且使用跨越多个级别的混合描述模拟该系统。因此,可以对由高层次行为描述的子系统及低层次详细实现子系统所组成的系統进行模拟。

需要注意的是,VHDL 语言并不具有描述模拟电路的能力。虽然某些研究的成果以实例表明,VHDL 的描述能力可以扩展到电路级[HaSt91],但是在这个级别上,VHDL 语言不是一种理想的语言。把 VHDL 语言的描述能力扩展到描述模拟电路由 IEEE 的 1076.1 小组进行。该小组正在设计一种新语言,以便既能描述模拟电路又能描述数模混合电路。这种新语言和我们提到的描述数字电路的 VHDL'93 非常接近。

1.4 VHDL 语言的开发环境

VHDL 语言可以在多种 EDA 工具设计环境中运行,硬件平台是工作站或高档微机。有许多公司都提供 VHDL 工具,其中包括 VHDL 的创始者(例如 Intermetrics 公司,该公司的 VHDL 部分已经并入 Cadence 公司),也包括一些大型的 EDA 公司(例如 Cadence, Mentor

Graphics, ViewLogic, Synopsys 等), 还包括一些专门领域的公司(例如 CLSI, Model-Technology, Vantage 等)。VHDL 模拟器和综合器是商业软件, 可以在多种工作平台上运行(VHDL 描述不是 100% 可以被综合的, 不同公司提供的工具软件在描述风格和构造方面会有不同的指南和限制, 满足这些要求的 VHDL 描述才能被该公司提供的工具所综合)。

欧洲有一个促进 IP 核、集成电路及微机械系统技术发展的组织 EUROPACTICE, 它的网址是 <http://www.te.rl.ac.uk/europactice>, 访问这个站点可以得到 54 种设计工具(参考 2000 年 7 月 21 日发行的光盘 CD-ROM version 11.0 中的数据), 其中有一些和 VHDL 有关, 下面将简要介绍这部分工具。

1. Cadence

该公司宣称他们的 VHDL 模拟器(Leapfrog)是工业界最快的 VHDL 模拟器, 并且和逻辑综合工具的 Synergy 系列紧密相连。

2. Synopsys

该公司提供 VHDL 系统模拟器(VSS Expert)和 VHDL 编译器。该模拟器有一条命令, 可以告知程序中的某一行执行了多少次。留意观察这些数字, 就可以知道该模型或测试基准的哪一部分执行的次数最少。根据设计者的输入数据以及设计者目前所处的设计阶段, Synopsys 的 Design Power 可以快速给出比较精确的功耗反馈信息。在设计的早期, Design Power 根据设计者对电路状态转换频率的粗略估计, 可以快速给出功耗反馈信息, 设计者利用此信息可以对设计指标作出折衷选择。在设计的后期即门级优化阶段, Design Power 利用门级模拟器提供的精确的电路状态转换数据, 可以给出精确的功耗反馈信息。Synopsys 的 Design Analyzer 是一个强大的综合工具, 可以把 VHDL 源代码综合成多种格式的门级网表。

3. Mentor Graphics

该公司提供 VHDL 模拟器 QuickVHDL 和 QuickSim II, 以及 VHDL 综合工具 Autologic VHDL。

4. Model-Technology

该公司提供个人计算机 Windows 环境下运行的 VHDL 模拟器(V-System/Windows), 它也有在工作站上运行的版本。

5. Altera

Altera 公司的 PLSM-VHDL(适用于个人计算机)和 PLSM-VHDL WS(适用于 SUN4 和 HP 9000 工作站)是 VHDL 的输入工具, 支持该公司的 Classic, Max 5000, Max 7000, Flex 8000 等可编程器系列。Altera 还提供和 Mentor-Graphics, Cadence 以及 Synopsys 工具的接口和库。

此外, VEDA 和 Xilinx 公司也支持 VHDL 设计输入。VEDA 有一个叫做 VHDL Path 的工具, 它可以计算语句、分支、条件、路径和信号的作用范围, 并给出统计报告。