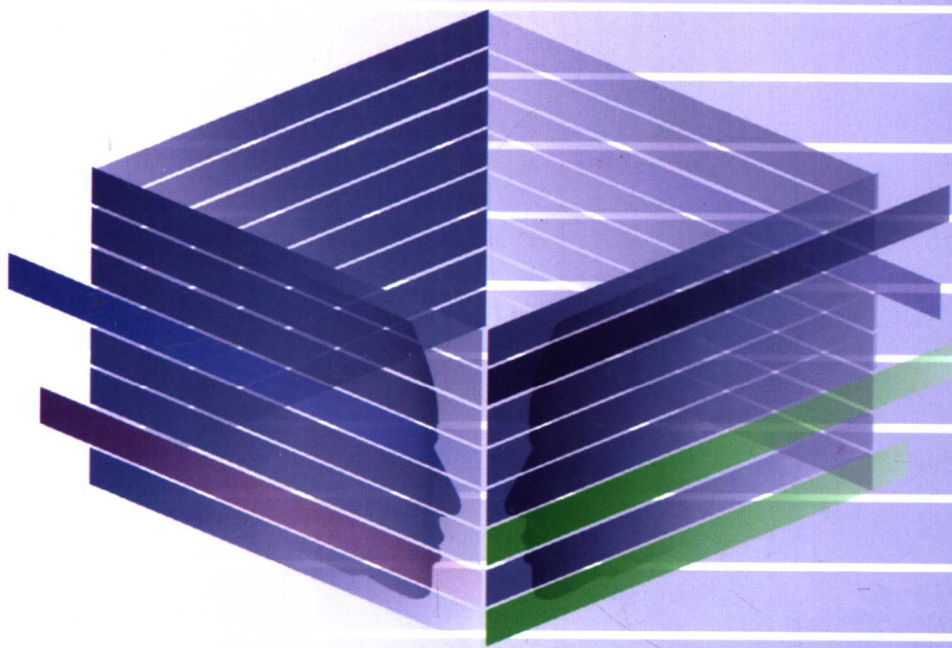


硬件描述语言 VHDL教程

(基础篇·提高篇)

姜雪松 刘东升 编著

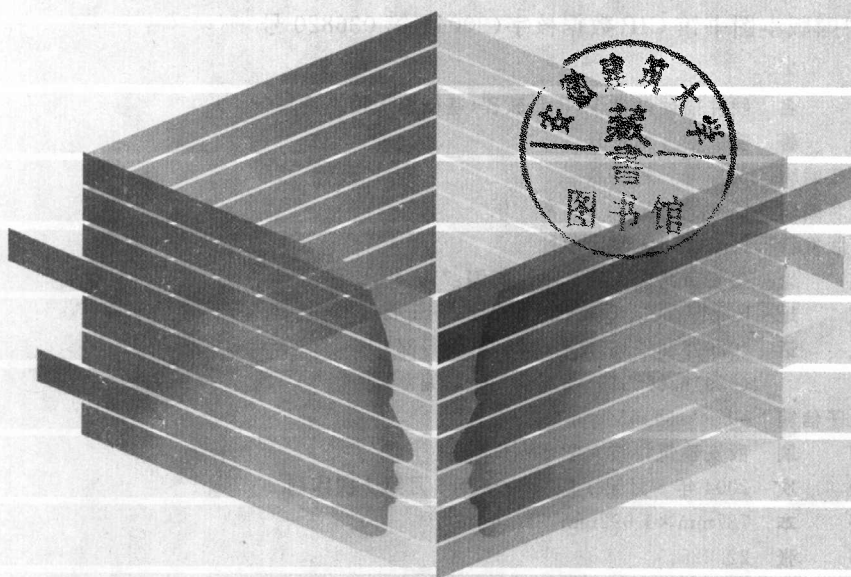


西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

硬件描述语言 VHDL教程

(基础篇·提高篇)

姜雪松 刘东升 编著



西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

内 容 提 要

在硬件电子电路设计领域中,设计自动化工具已经被广大电子工程师所接受,它必将取代人工设计方法,成为主要的设计手段。电子设计自动化的重要特征就是要求采用形式化语言来描述硬件电子电路的功能,即使用硬件描述语言来描述硬件电子电路。VHDL 语言是国际标准化硬件描述语言,在电子系统自动化设计中已经十分流行,而且已经成为主要的硬件描述工具。如今在电子系统设计领域中,它已成为广大技术人员必须掌握的一种语言。

本书分为上、下两册,上册包含基础篇和提高篇,适合作为高校电子信息类专业本科教材;下册包含应用篇、部分习题答案、附录,适合作为补充教材或教学参考书。其中,基础篇介绍了 VHDL 语言的基础知识,主要目的是为初学者打下一个良好的基础;提高篇介绍了 VHDL 语言的高级知识和目前在电子电路设计中常用电路结构的 VHDL 描述,目的是使读者掌握用 VHDL 语言设计简单逻辑电路的基本方法;应用篇从一些大型实例出发,介绍用 VHDL 语言设计大型复杂电路的流程和在设计过程中所用到的设计技巧,并且简要介绍了可编程逻辑器件的基础知识和 MAX+plus II 系统软件的基本使用。书中列举的大量实例都经过计算机上的 MAX+plus II 软件或 SUN SPARC 工作站上的 Synopsys 软件的验证。

本书的特点是全面系统、易读易懂、由浅入深,能够使读者逐步掌握 VHDL 语言。本书不仅注重基础知识的介绍,而且力求向读者系统地讲解 VHDL 的实际应用。它既可作为高等学校计算机和电子工程专业的研究生、本科生的教材和教学参考书,也可作为广大电子电路设计工程师、ASIC 设计人员和系统设计人员的参考书。

图书在版编目(CIP)数据

硬件描述语言 VHDL 教程(基础篇·提高篇)/姜雪松,刘东升编著.

—西安:西安交通大学出版社,2004.6

ISBN 7-5605-1815-X

I. 硬... II. ①姜... ②刘... III. 硬件描述语言,VHDL—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 036820 号

书 名 硬件描述语言 VHDL 教程(基础篇·提高篇)

编 著 姜雪松 刘东升

责任编辑 贺峰涛 屈晓燕

文字编辑 黄 娟 吴 飞 常 乐

出版发行 西安交通大学出版社

地 址 西安市兴庆南路 25 号(邮编:710049)

网 址 <http://unit.xjtu.edu.cn/unit/jtupress>

电 话 (029)82668357 82667874(发行部)

(029)82668315 82669096(总编办)

电子信箱 eibooks@163.com

印 刷 西安建筑科技大学印刷厂

版 次 2004 年 6 月第 1 版 2004 年 6 月第 1 次印刷

开 本 787mm×1 092mm 1/16

印 张 22.625

字 数 548 千字

书 号 ISBN 7-5605-1815-X/TP·374

总 定 价 52.00 元 本册定价 32.00 元

版权所有 侵权必究

前 言

随着科学技术的迅猛发展,电子工业界经历了巨大的飞跃。集成电路的设计正朝着速度快、性能高、容量大、体积小和微功耗的方向发展,这种发展必将导致集成电路的设计规模日益增大,复杂程度日益增高。在这种情况下,沿用了十几年的传统硬件电路设计方法已不能满足需要,它已经远远落后于当今技术的发展。

因此我们必须采用一种新的硬件电路设计方法来代替传统的硬件电路设计方法,以满足大规模电路设计的需要。根据系统对硬件的行为和功能要求,自上而下地逐层完成相应的描述、模拟、综合、优化和布线,直到完成硬件设计。在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地完成,这种方法称为电子设计自动化(EDA)。它可以大大节省人力和物力、缩短研制周期,适用于小批量产品开发,也适用于大批量产品的样品研制,因而得到了广泛的应用。

电子设计自动化的重要特征就是要求采用形式化语言来描述硬件电子电路的功能,即采用硬件描述语言来描述硬件电子电路。目前,硬件描述语言 VHDL(Very High Speed Integrated Circuit(VHSIC)Hardware Description Language)作为 IEEE 标准所规范的硬件描述语言,随着各种 EDA 工具和集成电路厂商的普遍认同和推广,目前正在全球范围内先进国家的电子系统设计领域获得广泛应用。

随着 VHDL 语言的广泛应用,无论是电子设计工程师,还是高等院校的学生都迫切需要一本除了介绍 VHDL 基本概念和基本语法外,还能够从实际出发、着重介绍各种不同电路结构的描述方法的参考书。作者编写此书的目的就是从以上方面来满足广大读者的需要,以使读者能够全面掌握 VHDL 语言。

本书分为三个部分对 VHDL 语言进行全面介绍。

第一部分基础篇,主要介绍 VHDL 语言的基础知识,目的是为初学者打下一个良好的基础。该部分由第 1 章到第 6 章组成,重点介绍了 VHDL 概述,VHDL 的基本模型结构,VHDL 的基本元素,VHDL 的基本描述语句,VHDL 的库、程序包和配置的概念以及 VHDL 的预定义属性。

第二部分提高篇,介绍了 VHDL 语言的高级知识和目前在电子电路设计中常用电路结构的 VHDL 描述,目的是使读者掌握用 VHDL 语言设计简单逻辑电路的基本方法。这部分包括第 7 章到第 12 章;第 7 章和第 8 章重点介绍了重载的概念、VHDL 的综合等高级知识;第 9 章详细介绍了常用的数字逻辑电路的 VHDL 描述;第 10 章主要介绍了在硬件电路设计中广泛使用的状态机;第 11 章关于优化数据通路的一些方法;第 12 章则重点介绍了测试平台的概念以及建立测试平台的常用方法。

第三部分应用篇,从一些大型实例出发,介绍用 VHDL 语言设计大型复杂电路的流程和在设计过程中所用到的设计技巧,并且简要介绍了可编程逻辑器件的基础知识和 MAX+plus

II 软件的基本使用。这部分主要介绍了四个应用实例:第 13 章介绍了微波炉定时器芯片的设计,目的是使读者掌握设计大型数字电路的流程;第 14 章给出了一个数字频率计的设计,目的是进一步掌握设计大型电路的方法;第 15 章给出了计时器的设计,目的是加深对自顶向下的设计方法的理解;第 16 章给出了一个 32 位微处理器的例子,目的是为了锻炼读者阅读 VHDL 程序的能力。第 17 章介绍了可编程逻辑器件的基础知识,目的是使读者了解可编程逻辑器件的结构,以有助于今后的硬件设计。第 18 章以一个加法计数器的设计为例详细地介绍了 MAX+plus II 软件的使用。

考虑到大部分读者的需要,将上述内容分为上、下两册出版。其中上册包含基础篇和提高篇,适合作为高校电子信息类专业本科教材使用;下册包含应用篇、部分习题解答和附录等,适合作为补充教材或教学参考书。

作者在本书中列举的大量小例子许多都取自于实际硬件电路的设计,并且经过了计算机上的 MAX+plus II 软件或 SUN SPARC 工作站上的 Synopsys 软件的编译验证。

本书的特点是全面系统、易读易懂、由浅入深,能够使读者逐步掌握 VHDL 语言。本书不仅注重基础知识的介绍,而且力求向读者系统地讲解 VHDL 的实际应用。它既可作为高等学校计算机和电子工程专业的研究生、本科生的教材和教学参考书,也可作为广大电子电路设计工程师、ASIC 设计人员和系统设计人员的参考书。

作者在编写本书的过程中参考了许多关于 VHDL 语言的最新专著及文献,同时本书也包含着作者在使用 VHDL 语言设计硬件电路过程中的经验总结。郭健、蒋亮为本书的编写整理了大量的资料,罗卫国、周健成、赵鑫完成了全书的文字校对工作,姜立冬、林辉负责调试了书中的所有程序。在此,向所有为本书作出贡献的人们表示衷心的感谢!

由于作者的水平有限,书中难免存在错误和不足之处,望读者不吝批评指正。

作者

2004 年 1 月

目 录

基础篇

第 1 章 概述	(3)
1.1 硬件描述语言 VHDL	(3)
1.1.1 VHDL 的发展历史	(3)
1.1.2 VHDL 的特点	(4)
1.1.3 VHDL 的发展趋势	(7)
1.2 EDA 技术	(7)
1.2.1 EDA 的发展历史	(8)
1.2.2 EDA 的基本特征	(9)
1.2.3 EDA 的基本工具	(10)
1.2.4 EDA 工具的流程	(11)
习题	(12)
第 2 章 VHDL 的基本模型结构	(13)
2.1 设计实体	(13)
2.2 实体说明	(15)
2.2.1 类属说明	(15)
2.2.2 端口说明	(17)
2.2.3 实体说明部分	(18)
2.2.4 实体语句部分	(18)
2.3 结构体	(18)
2.3.1 结构体名	(19)
2.3.2 结构体说明语句	(19)
2.3.3 结构体功能描述语句	(19)
2.3.4 结构体的四种描述形式	(21)
2.4 结构体的三种子结构	(27)
2.4.1 块语句结构	(27)
2.4.2 进程语句结构	(29)
2.4.3 子程序结构	(32)
习题	(42)
第 3 章 VHDL 的基本元素	(45)
3.1 VHDL 的标识符与词法单元	(45)
3.1.1 标识符	(45)

3.1.2 词法单元·····	(46)
3.2 VHDL 的数据对象 ·····	(48)
3.2.1 常量·····	(48)
3.2.2 变量和信号·····	(49)
3.2.3 文件·····	(54)
3.2.4 信号的延迟·····	(56)
3.3 VHDL 的数据类型 ·····	(59)
3.3.1 标量类型·····	(60)
3.3.2 复合类型·····	(64)
3.3.3 子类型·····	(67)
3.3.4 存取类型和文件类型·····	(69)
3.3.5 数据类型的转换·····	(69)
3.4 VHDL 的运算符 ·····	(71)
3.4.1 逻辑运算符·····	(71)
3.4.2 算术运算符·····	(72)
3.4.3 关系运算符·····	(74)
3.4.4 并置运算符·····	(74)
3.4.5 运算符的优先级·····	(75)
习题 ·····	(76)
第4章 VHDL 的基本描述语句 ·····	(79)
4.1 并行语句·····	(79)
4.1.1 进程(process)语句 ·····	(80)
4.1.2 块(block)语句 ·····	(81)
4.1.3 并行信号赋值语句·····	(84)
4.1.4 并行过程调用语句·····	(89)
4.1.5 并行断言语句·····	(90)
4.1.6 类属(generic)语句 ·····	(92)
4.1.7 元件例化语句·····	(94)
4.1.8 生成(generate)语句 ·····	(95)
4.2 顺序语句 ·····	(100)
4.2.1 顺序赋值语句 ·····	(101)
4.2.2 wait 语句 ·····	(103)
4.2.3 if 语句 ·····	(105)
4.2.4 case 语句 ·····	(107)
4.2.5 loop 语句 ·····	(109)
4.2.6 next 语句 ·····	(112)
4.2.7 exit 语句 ·····	(112)
4.2.8 null 语句 ·····	(113)
4.2.9 return 语句 ·····	(113)

4.2.10 顺序断言语句	(113)
4.2.11 report 语句	(114)
习题	(115)
第5章 VHDL 的库、程序包和配置	(118)
5.1 VHDL 的库	(118)
5.1.1 库的使用	(118)
5.1.2 库的分类	(120)
5.2 VHDL 的程序包	(123)
5.2.1 程序包的基本结构	(123)
5.2.2 常见的程序包	(125)
5.3 VHDL 的配置	(126)
5.3.1 默认配置	(126)
5.3.2 元件配置	(129)
5.3.3 块配置	(134)
5.3.4 结构体配置	(137)
习题	(139)
第6章 VHDL 中属性的描述及定义语句	(140)
6.1 数值类属性	(141)
6.1.1 数据类型的数值属性	(141)
6.1.2 数组的数值属性	(142)
6.1.3 块的数值属性	(144)
6.2 函数类属性	(145)
6.2.1 数据类型的属性函数	(146)
6.2.2 数组的属性函数	(147)
6.2.3 信号的属性函数	(149)
6.3 信号类属性	(153)
6.4 数据类型类属性	(159)
6.5 数据区间类属性	(160)
6.6 用户自定义的属性	(161)
习题	(161)

提高篇

第7章 VHDL 中的重载	(165)
7.1 子程序重载	(165)
7.1.1 参数类型的重载	(166)
7.1.2 参数数目的重载	(172)
7.1.3 函数返回类型的重载	(176)
7.1.4 标准程序包中的重载现象	(180)
7.2 运算符重载	(184)

习题.....	(186)
第 8 章 综合	(187)
8.1 综合概述	(187)
8.1.1 RTL 级描述	(187)
8.1.2 约束	(191)
8.1.3 属性	(193)
8.1.4 工艺库	(194)
8.1.5 综合的步骤	(196)
8.2 VHDL 的综合	(197)
习题.....	(206)
第 9 章 简单数字逻辑电路的设计	(207)
9.1 数字系统的设计方法基础	(207)
9.1.1 数字系统的设计流程	(208)
9.1.2 设计的基本方法	(209)
9.1.3 设计的基本准则	(211)
9.2 基本逻辑门电路的设计	(213)
9.2.1 与门电路	(214)
9.2.2 或门电路	(216)
9.2.3 反相器	(219)
9.2.4 与非门电路	(220)
9.2.5 或非门电路	(223)
9.2.6 异或门电路	(226)
9.3 组合逻辑电路的设计	(228)
9.3.1 编码器	(228)
9.3.2 译码器	(232)
9.3.3 数据选择器和分配器	(241)
9.3.4 比较器	(248)
9.3.5 加法器	(250)
9.3.6 三态门及数据缓冲器	(252)
9.4 时序逻辑电路的设计	(255)
9.4.1 时钟信号的描述	(255)
9.4.2 触发器	(257)
9.4.3 寄存器	(268)
9.4.4 计数器	(277)
9.4.5 存储器	(287)
习题.....	(295)
第 10 章 状态机的设计	(296)
10.1 状态机概述	(296)
10.1.1 状态机的基本结构和功能	(296)

10.1.2	状态机的表示方法	(298)
10.1.3	状态机的设计步骤	(299)
10.2	Moore 型状态机的设计描述	(300)
10.2.1	状态机的建立过程	(300)
10.2.2	单进程状态机的设计方法	(302)
10.2.3	双进程状态机的设计方法	(303)
10.2.4	三进程状态机的设计方法	(309)
10.3	Moore 型状态机的复位	(311)
10.3.1	状态机的同步复位	(311)
10.3.2	状态机的异步复位	(315)
10.4	Moore 型状态机的信号输出方式	(316)
10.4.1	同步的信号输出方式	(316)
10.4.2	状态直接输出的方式	(318)
10.4.3	并行译码的信号输出方式	(321)
	习题	(323)
第 11 章	优化数据通路	(324)
11.1	流水线设计	(325)
11.1.1	流水线设计的基本概念	(325)
11.1.2	应用流水线设计的 DRAM 控制器	(326)
11.2	资源共享	(335)
11.2.1	流水线式行波进位加法器	(336)
11.2.2	预进位加法器	(338)
	习题	(342)
第 12 章	建立测试平台	(343)
12.1	概述	(343)
12.2	建立测试平台的方式	(344)
12.2.1	表格化方式	(344)
12.2.2	文件 I/O 方式	(348)
	习题	(351)

附:《硬件描述语言 VHDL 教程(应用篇·习题解答)》目录

应用篇

第 13 章 微波炉定时器芯片的设计

- 13.1 定时器芯片的功能描述
- 13.2 定时器芯片的模块划分
- 13.3 定时器芯片的模块设计
 - 13.3.1 控制状态机
 - 13.3.2 数据装入器
 - 13.3.3 定时器
- 13.4 定时器芯片的顶层设计

第 14 章 数字频率计的设计

- 14.1 数字频率计的功能描述
- 14.2 数字频率计的模块划分
- 14.3 数字频率计的模块设计
 - 14.3.1 时钟发生器
 - 14.3.2 高频计数器
 - 14.3.3 低频计数器
 - 14.3.4 倒数及 BCD 码转换器
 - 14.3.5 中央控制电路
- 14.4 数字频率计的顶层设计

第 15 章 计时器的设计

- 15.1 计时器的功能描述
- 15.2 计时器的模块划分
- 15.3 计时器的模块设计
 - 15.3.1 键输入模块
 - 15.3.2 时钟分频模块
 - 15.3.3 控制模块
 - 15.3.4 计时模块
 - 15.3.5 显示模块
- 15.4 计时器的顶层设计

第 16 章 32 位微处理器的设计

第 17 章 可编程逻辑器件基础

- 17.1 可编程逻辑器件的分类
 - 17.1.1 可编程逻辑器件的集成度分类
 - 17.1.2 其他分类方法
- 17.2 PLD 器件的基本结构
 - 17.2.1 与或阵列
 - 17.2.2 宏单元
 - 17.2.3 一个具体的应用实例
- 17.3 FPGA 器件的基本结构
 - 17.3.1 查找表型 FPGA 的结构
 - 17.3.2 多路开关型 FPGA 的结构
 - 17.3.3 多级与非门型 FPGA 的结构
 - 17.3.4 一个具体的应用实例
- 17.4 可编程逻辑器件的编程与测试技术
 - 17.4.1 在系统可编程技术
 - 17.4.2 边界扫描测试技术

第 18 章 MAX+plus II 的使用入门

- 18.1 MAX+plus II 概述
 - 18.1.1 MAX+plus II 简介
 - 18.1.2 MAX+plus II 的安装
 - 18.1.3 MAX+plus II 对 VHDL 的支持
 - 18.1.4 MAX+plus II 的组成部分
- 18.2 MAX+plus II 的使用
 - 18.2.1 设计输入
 - 18.2.2 设计实现
 - 18.2.3 设计校验
 - 18.2.4 器件编程

部分习题解答

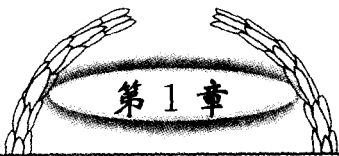
附录 A VHDL 的保留字

附录 B VHDL 的标准包集合文件

参考文献

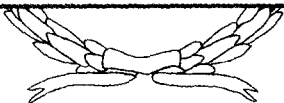
基础篇

- ◇ 第 1 章 概述
- ◇ 第 2 章 VHDL 的基本模型结构
- ◇ 第 3 章 VHDL 的基本元素
- ◇ 第 4 章 VHDL 的基本描述语句
- ◇ 第 5 章 VHDL 的库、程序包和配置
- ◇ 第 6 章 VHDL 中属性的描述及定义语句



第1章

概 述



随着科学技术的迅猛发展,电子工业界经历了巨大的飞跃。在电子电路设计领域中,速度快、性能高、容量大、体积小和微功耗成为集成电路设计的主要发展方向。为了适应这些新的需求,设计自动化工具已经被广大电子工程师所接受,它必将取代人工设计方法,成为主要的设计手段。目前,VHDL 已经成为各家 EDA 工具和集成电路厂商所普遍认同和共同推广的标准化硬件描述语言。因此,掌握 EDA 技术,学会用 VHDL 设计电子电路是每个硬件设计工程师必须掌握的一项基本技能。

本章我们将对硬件描述语言 VHDL 和 EDA 技术进行概括介绍,使读者对它们有一个初步的了解。



硬件描述语言 VHDL

1.1.1 VHDL 的发展历史

20 世纪 70 年代,随着对中小规模集成电路的大量需求,集成电路的设计正朝着速度快、性能高、容量大、体积小和微功耗的方向发展,而这种发展必将导致集成电路的设计规模日益增大、复杂程度日益增高和精度要求更加严格。在这种情况下,影响开发商的瓶颈问题就是其设计能力,这种状况实际上孕育着对现代设计方法和现代测试方法的普遍需求。沿用了十几年的传统硬件电路设计方法已经不能满足需要,它已经远远落后于当今技术的发展。因此,大规模高密度可编程逻辑器件和硬件描述语言 VHDL 成为解决这些问题的关键所在。

一个较优选的设计方式应该能够提高设计者的工作效率,并且能够使设计输入、设计理解 and 设计维护更加便利和快捷。这种设计方式要求能够满足以下几个条件:它应该是开放的、非专业的、工业用的、工业界所能普遍接受的标准;它允许设计在不同的 EDA 工具环境间移植,并且它的模块能够封装成独立单元重复使用;它支持阶层结构的复杂设计和从门级到系统级的设计描述。

因此我们必须采用一种新的硬件电路设计方法来代替传统的硬件电路设计方法,以满足

大规模集成电路设计的需要。根据系统对硬件的行为和功能要求,自上而下地逐层完成相应的描述、模拟、综合、优化和布线,直到完成硬件设计。在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地完成,这种方法称为电子设计自动化(EDA)。它可以大大节省人力和物力、缩短研制周期,适用于小批量产品开发,同时也适用于大批量产品的样品研制,因而得到了广泛的应用。

电子设计自动化要求用硬件描述语言(HDL)来进行硬件的行为和功能描述,因此众多软件公司研制开发了具有自己特色的硬件描述语言,这些硬件描述语言必将有很大的差异,而硬件设计工程师一旦选用了电子设计自动化工具,就会被束缚在这个硬件设计环境中,而不能在众多的电子设计自动化工具中选择一个最佳组合作为自己的最佳设计环境。因此,硬件工程师需要一种强大的面向设计的多层次、多领域并得到广大 EDA 厂商认同的标准化硬件描述语言。

美国国防部在 20 世纪 70 年代末和 80 年代初提出了 VHSIC(Very High Speed Integrated Circuit)计划,VHSIC 计划的目标是为下一代集成电路的生产、实现阶段性的工艺极限以及完成 10 万门级以上的设计,建立一项新的描述方法。1981 年,美国国防部提出了一种新的硬件描述语言,称为“超高速集成电路硬件描述语言”(VHSIC Hardware Description Language),简称 VHDL。当这个语言被首次开发出来时,其目标只是一个使电路文本化的一种标准,主要是为了使采用了文本描述的设计能够为其他人所理解,同时也用作模型语言,能采用软件进行模拟。VHDL 的结构和设计方法受到了 Ada 语言的影响,并吸收了其他硬件描述语言的优点。1986 年,IEEE 致力于 VHDL 的标准化工作,为此成立了 VHDL 标准化小组。经过了多次的修改与扩充,直到 1987 年 12 月 VHDL 才被接纳为 IEEE 1076 标准。1988 年,Milstd454 规定所有为美国国防部设计的 ASIC 产品必须采用 VHDL 来进行描述。1993 年,IEEE 1076 标准被修订,更新为新的 VHDL 标准 IEEE 1164。1996 年,IEEE1076.3 成为 VHDL 综合标准。

VHDL 成为标准以后,很快在世界各地得到了广泛应用,为电子设计自动化的普及和推广奠定了坚实的基础。1995 年我国国家技术监督局制定的《CAD 通用技术规范》推荐 VHDL 作为我国电子设计自动化硬件描述语言的国家标准。

1.1.2 VHDL 的特点

今天,VHDL 已经成为一个数字电路和系统的描述、建模、综合的工业国际标准,因此在电子产业界获得了广泛的应用。VHDL 能够成为标准并获得广泛应用,必然具有与其他硬件描述语言不同的地方,这恰恰是 VHDL 的优越性所在。下面我们对 VHDL 的特点进行介绍,首先来看看 VHDL 的优点:

(1) 功能强大、设计灵活

VHDL 具有功能强大的语言结构,可以用简洁明确的程序来描述复杂的逻辑功能。为了有效控制设计的实现,它还具有多层次的设计描述功能,支持设计库和可重复使用的元件生成;它支持阶层设计,层层细化,最后可直接生成电路级描述;VHDL 还支持同步电路、异步电路和随机电路的设计。

VHDL 的设计非常灵活,这主要源于 VHDL 支持各种设计方法。从设计流程上来说,它既支持自底向上的设计,也支持自顶向下的设计;从设计层次上说,它既支持模块化设计,也支

持层次化设计。

(2)强大的系统硬件描述能力

VHDL 具有多层次描述系统硬件功能的能力,可以从系统的数学模型直到门级电路。另外,高层次的行为描述可以与低层次的寄存器传输描述和结构描述混合使用。VHDL 能进行系统级的硬件描述,这是它的最突出优点。

VHDL 一个突出的特点是具有丰富的数据类型,既可以支持预定义的数据类型,也可以自己定义数据类型,这样便给硬件描述带来了较大的自由度。

同时 VHDL 具有子程序调用功能,对于已经完成的设计程序,我们可以通过修改子程序的方法来改变设计的规模和结构,这样使得对系统硬件的描述更加灵活。

(3)移植能力强

由于电子设计自动化技术的普及和推广,不同的 EDA 厂商纷纷推出自己的 EDA 开发工具。各个 EDA 开发工具的不同导致了模拟工具、开发工具和操作平台的不同,这样就会对一些硬件描述语言的使用产生了限制。而这对 VHDL 的使用没有丝毫的影响,现在几乎所有的 EDA 开发工具都支持 VHDL,这也正是 VHDL 广泛使用的重要原因。

用 VHDL 进行硬件电路设计时,VHDL 的可移植能力主要体现在:对于同一个设计描述,它可以从一个模拟工具移植到另一个模拟工具,从一个综合工具移植到另一个综合工具,从一个工作平台移植到另一个工作平台去执行。

如图 1.1 所示体现了 VHDL 程序的移植能力。从图中我们可以看出,某个 VHDL 程序可以用任一模拟工具进行模拟;进行综合时,也可以选择任一综合工具进行综合,体现了 VHDL 强大的移植能力。

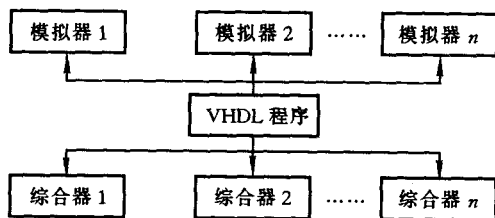


图 1.1 VHDL 程序不依赖于模拟器和综合器的移植能力

(4)VHDL 语法规则、标准,易于共享与复用

VHDL 语法规则、标准,可读性强。用 VHDL 书写的源文件既是程序,又是文档;既是技术工程人员进行设计成果交流的交流文件,也可作为合同签约者之间的合同文本。

VHDL 采用基于库的设计方法。这样在设计一个大规模集成电路的过程中,技术人员就不需要从门级电路开始一步步地进行设计,可以用原来设计好的模块直接进行累加。这些模块可以预先设计或者使用以前设计中的存档模块,将这些模块存放在库中,就可以在以后的设计中进行复用。不难看出,复用减小了硬件电路设计的工作量,缩短了开发周期。

由于 VHDL 是一种 IEEE 的工业标准硬件描述语言,具有严格的语法规则和统一的标准,因此它可以使设计成果在设计人员之间进行交流和共享。反过来,就可以进一步推动 VHDL 的发展和完善。

(5)支持广泛、易于修改

由于 VHDL 已经成为 IEEE 标准所规范的硬件描述语言,目前大多数的 EDA 工具几乎都支持 VHDL,这为 VHDL 的进一步推广和和广泛应用奠定了基础。目前许多公司都提供了 VHDL 开发工具,其中既包括一些像 Cadence, Mentor Graphics 和 Synopsys 等大型 EDA 公司,也包括一些专门领域的公司,例如 Model-Technology。

在用 VHDL 进行硬件电路设计的过程中,主要的设计文件是用 VHDL 编写的程序,因为 VHDL 遵循统一的标准和规范,所以易于修改。

(6)与工艺无关

当设计人员用 VHDL 进行硬件电路设计时,并没有涉及到与工艺有关的信息。当然,这样的信息是可以用 VHDL 来编写的,但在设计中我们往往不考虑与工艺有关的信息。当一个设计描述用 VHDL 模拟器和 VHDL 综合器进行完编译、模拟和综合后,就可以采用不同的映射工具将设计映射到不同的工艺上去。映射成不同的工艺,只需要改变相应的映射工具,而无需修改设计描述。

(7)易于 ASIC 移植

VHDL 效率高的重要体现之一就是如果技术工程人员的设计是被综合到一个 CPLD 或 FPGA 的话,则可以使设计的产品以最快的速度上市。当产品的数量达到相当的规模时,采用 VHDL 能够很容易地帮助技术工程人员实现转成 ASIC 的设计。有时用于 PLD 的程序可以直接用于 ASIC,并且由于 VHDL 是一种 IEEE 的工业标准硬件描述语言,所以用 VHDL 设计可以确保 ASIC 厂商生产高质量的器件产品。

(8)上市时间短,成本低

VHDL 和可编程逻辑很好地结合,可以大大提高数字单片化设计的实现速度。VHDL 使设计描述更加方便、快捷,可编程逻辑应用可以将产品设计的前期风险降至最低,并使设计的快速复制简单易行,同时多种综合工具都支持这种形式的设计。VHDL 和可编程逻辑的组合作为一种强有力的设计方式,将为设计者的产品上市带来创纪录的速度。

VHDL 作为 IEEE 的工业标准具有许多其他硬件描述语言所不具有的优点以外,它也存在一些缺点:

(1)不具有描述模拟电路的能力

VHDL 不具有描述模拟电路的能力,虽然研究结果证明 VHDL 可以扩展到电路级上,但在电路级上 VHDL 并不是一种理想的语言。目前 IEEE 的 1076.1 小组正在设计一种新的语言,这种语言能够描述模拟电路和数模混合电路。这个新语言将以 VHDL 为基础,并在此基础上增加描述模拟电路的扩展内容。

(2)综合工具生成的逻辑实现有时并不最佳

技术设计人员采用综合工具所生成的逻辑实现有时候并不能让人满意,因为优化的结果往往依赖于设计的目标。现在所用的综合工具采用一定的算法来对设计的实现进行控制,但是固定的算法并不能发现设计中的所有问题,这样就有可能导致综合工具生成的逻辑实现与技术人员希望的逻辑实现有一定的差距。

(3)EDA 工具的不同导致综合质量的不同

不同的 EDA 工具对同一 VHDL 描述进行综合,往往产生不同的综合质量,这是因为不同的 EDA 工具采用不同的算法所致。因此设计人员在设计的时候往往需要对不同 EDA 工具