

高等院校电子信息科学与工程规划教材

VHDL 程序设计教程

(第4版)

曾繁泰 曾祥云 编著



从D触发器模块到SOC芯片设计、仿真、测试与驱动



清华大学出版社

014035115

TP312VH-43
03-4

高等院校电子信息科学与工程规划教材

VHDL 程序设计教程

(第4版)

曾繁泰 曾祥云 编著



清华大学出版社

北 京



北航

C1714600

TP312VH-43
03-4

内 容 简 介

本书系统地阐述了 VHDL 硬件描述语言的基础知识、语法定义、编程技巧和使用方法。全书共 9 章，第 1 章为绪论，包括 VHDL 概述、语言设计环境、设计实现和教学建议；第 2 章为 VHDL 语言的词法基础；第 3 章为 VHDL 语言的程序结构；第 4 章为 VHDL 的语法基础；第 5 章为 VHDL 程序设计基础，第 4、5 章的 VHDL 程序片段完全以理解语法规范为目的；第 6 章为 VHDL 程序设计实践，本章的 VHDL 程序片段完全以理解逻辑电路结构为目的；第 7 章为 VHDL 仿真与测试，给出了 VHDL 测试程序、仿真程序的编写方法和片上内建自测试的概念；第 8 章为 VHDL 标准程序接口，列出了接口函数清单，给出了 C++ 联合 VHDL 设计、仿真、测试的实例；第 9 章为 IP 复用与 SoC 设计，详细介绍了 SoC 芯片研发流程和软件驱动开发环境。

本书面向高等院校本、专科的 VHDL 语言基础课程，可作为电子工程、信息工程、通信工程、物联网、工业自动化、计算机应用技术、仪器仪表、数字信号或图像处理等专业的授课教材或教学参考书，同时也可作为电子行业工程师的 VHDL 编程、FPGA 开发、SoC 设计的自学参考书。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。
版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目 (CIP) 数据

VHDL 程序设计教程/曾繁泰，曾祥云编著. —4 版. —北京：清华大学出版社，2014
高等院校电子信息科学与工程规划教材
ISBN 978-7-302-34891-7

I. ①V… II. ①曾… ②曾… III. ①硬件描述语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 311136 号

责任编辑：朱英彪

封面设计：刘 超

版式设计：文森时代

责任校对：王 云

责任印制：李红英

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者：北京富博印刷有限公司

装 订 者：北京市密云县京文制本装订厂

经 销：全国新华书店

开 本：185mm×260mm 印 张：19.75 字 数：479 千字

版 次：2001 年 1 月第 1 版 2014 年 3 月第 4 版 印 次：2014 年 3 月第 1 次印刷

印 数：1~3800

定 价：36.00 元

产品编号：053372-01

前 言

当前 VHDL 语言不断推出新的版本,EDA 工具不断完善,VHDL 语言的实现载体 FPGA 器件已经集成了百万逻辑门。同时,大学教育更加注重对学生创新能力与实践能力的培养。在这些因素的推动下,我们开始实施对《VHDL 程序设计教程(第3版)》的修订工作,删除了一些过时的内容,加入了 VHDL 2008 版增加的 VHPI 接口技术,在为读者提供更多 VHDL 新知识的同时,给出了大量新的设计和实验。为方便教学,书中大部分章节都安排了针对性较强的习题、实验和设计项目,使每一章的课堂教学效果都能及时通过实验得以强化。

本书根据高校课堂教学中学生的认知规律和实验教学中对学生动手能力的培养要求,以提高学生的设计能力和自主创新能力为目的,系统阐述了 VHDL 硬件描述语言的基础知识、语法定义、编程技巧和使用方法,在先进的 EDA 软件设计平台——Quartus II 13.0 和实现载体——STRATIX V 系列 FPGA 上实现了 VHDL 模块设计。通过本书的学习,读者能够理解 VHDL 语言设计,掌握 EDA 工程开发技术,熟悉 SoC 芯片研发方法。从 D 触发器设计到网络多核处理器设计,从易到难,从简单到复杂,跨度之大前所未有的;从硬件描述语言到软件编程语言,从硬件模块设计到软件驱动设计,从仿真到测试,范围之广实属罕见;从学生基础实验到尖端芯片研发,从设计语言的理论分析到工业芯片的生产测试,务实之风贯穿全篇。

为了尽可能缩短授课时数,优化教学效果,提高教学效率,本书详细介绍了 VHDL 相关语句语法,在给出完整并被验证过的 VHDL 描述的同时,还给出了综合后的 RTL 电路图及表现该电路系统功能的时序波形图。由此,通过一些简单、直观、典型的示例,将 VHDL 最核心、最基本的内容解释清楚,使读者能有效地掌握 VHDL 的主要内容。精通 VHDL 语言的语法规则和熟悉各种通用逻辑电路,是 VHDL 编程的两个必要条件。

现代电子工程和 EDA 技术的发展速度很快,VHDL 2008 版增加了软件接口规范,为 SoC 设计提供了软、硬件协同设计的条件。本书力图在 SoC 软件/硬件设计、仿真测试等方面都有所涉及,但限于作者知识面狭窄,本书还有诸多不足之处,希望业内专家、同行不吝斧正。

此外,由于本书篇幅所限,有些内容未能列入,包括诸多示例和实验、FPGA 实验平台更详细的使用方法,以及本书相关的 EDA 技术与 FPGA 开发的培训信息等,读者可浏览网址 www.SDDPM.com 进行了解。

本书由曾祥云修订第 1~5 章,并编写了第 8 章;曾繁泰编写了第 7 章和第 9 章,并通审了全部书稿。本书的修订、撰写得到了清华大学出版社朱英彪先生的大力支持,在此表示衷心的感谢。

作者的 E-mail 是 zft@sdu.edu.cn,欢迎读者来信交流。

编 者
于山东大学

目 录

第 1 章 绪论	1
1.1 VHDL 概述	1
1.1.1 硬件描述语言的诞生	1
1.1.2 HDL 语言的种类	2
1.1.3 VHDL 语言的特点	3
1.1.4 VHDL 语言的优势	4
1.2 VHDL 语言设计环境	4
1.2.1 设计工具	4
1.2.2 仿真工具	6
1.3 VHDL 语言设计实现	7
1.4 VHDL 语言课程教学建议	9
1.4.1 课时安排	9
1.4.2 实验	9
1.5 小结	11
1.6 习题	11
第 2 章 VHDL 语言的词法基础	12
2.1 对象	12
2.1.1 对象的声明与初始化	12
2.1.2 常量	12
2.1.3 变量	13
2.1.4 信号	14
2.1.5 文件	14
2.1.6 对象的隐式创建	15
2.2 词法元素	15
2.2.1 数字	15
2.2.2 字符、字符串及位串	16
2.2.3 分隔符	17
2.2.4 标识符和保留字	17
2.2.5 注释	19
2.3 数据类型	19
2.3.1 类型与子类型	20
2.3.2 标量类型	20



2.3.3	复合类型	23
2.3.4	存取类型	25
2.3.5	文件类型	25
2.3.6	隐含类型	25
2.3.7	IEEE 标准类型	26
2.3.8	类型转换	27
2.4	运算操作符	28
2.4.1	逻辑运算符	28
2.4.2	关系运算符	29
2.4.3	移位运算符	30
2.4.4	并置运算符	31
2.4.5	算术运算符	31
2.4.6	状态运算符	31
2.4.7	运算符的优先级	32
2.5	小结	32
2.6	习题	33
第3章	VHDL 语言的程序结构	34
3.1	实体	35
3.1.1	类型说明部分	36
3.1.2	端口说明部分	36
3.1.3	实体说明部分	38
3.1.4	实体语句部分	38
3.2	结构体	38
3.2.1	结构体名	39
3.2.2	结构体说明部分	39
3.2.3	结构体语句	40
3.2.4	进程语句	40
3.2.5	模块语句	45
3.3	子程序	49
3.3.1	过程语句	49
3.3.2	函数语句	51
3.3.3	RETURN 语句	53
3.4	VHDL 程序设计风格	54
3.4.1	结构体的行为描述法	54
3.4.2	结构体的数据流描述法	55
3.4.3	结构体的结构化描述法	56
3.5	小结	57
3.6	习题	57



第 4 章 VHDL 的语法基础.....	58
4.1 并行语句与顺序语句.....	58
4.2 变量赋值语句与信号代入语句.....	59
4.2.1 变量赋值语句.....	60
4.2.2 信号代入语句.....	61
4.3 条件分支语句与循环语句.....	64
4.3.1 IF 语句.....	64
4.3.2 CASE 语句.....	67
4.3.3 LOOP 语句.....	68
4.3.4 NULL 语句.....	71
4.4 元件与模块的复用.....	71
4.4.1 元件声明与调用.....	71
4.4.2 参数传递子句与参数映射.....	71
4.4.3 I/O 端口子句与端口映射.....	73
4.4.4 生成语句.....	74
4.5 调试.....	77
4.5.1 REPORT 语句.....	77
4.5.2 断言语句.....	77
4.6 小结.....	78
4.7 习题.....	78
第 5 章 VHDL 程序设计基础.....	79
5.1 层次化设计方法.....	79
5.2 元件库.....	79
5.2.1 库的概念及语法.....	79
5.2.2 库的分类.....	80
5.2.3 构造元件.....	81
5.2.4 程序打包.....	87
5.2.5 构建元件库.....	88
5.2.6 元件的调用.....	88
5.3 配置.....	89
5.3.1 默认连接和默认配置.....	90
5.3.2 块配置.....	92
5.3.3 元件的配置.....	94
5.3.4 配置中的端口映射与参数传递.....	96
5.3.5 多层配置.....	100
5.4 可见性.....	100
5.4.1 声明的可见性.....	100
5.4.2 USE 语句.....	101
5.4.3 变量共享.....	101



5.5	重载	102
5.5.1	函数重载	102
5.5.2	运算符重载	105
5.6	别名	106
5.7	程序包	107
5.7.1	程序包声明	108
5.7.2	程序包体	109
5.7.3	常用程序包	110
5.8	文本 I/O 操作	116
5.8.1	文本类型定义	116
5.8.2	文件访问过程定义	117
5.8.3	TEXTIO 程序包的语法	117
5.8.4	TEXTIO 程序包的内容	118
5.8.5	TEXTIO 程序包的调用	119
5.9	小结	121
5.10	习题	122
第 6 章	VHDL 程序设计实践	123
6.1	组合逻辑设计	123
6.1.1	门电路	123
6.1.2	编码器的设计	124
6.1.3	译码器的设计	126
6.1.4	加法器的设计	128
6.2	时序逻辑电路设计	129
6.2.1	时钟信号的 VHDL 描述方法	130
6.2.2	时序电路中复位信号 Reset 的 VHDL 描述方法	132
6.3	触发器的设计	133
6.3.1	D 触发器	133
6.3.2	带异步复位的 D 触发器	134
6.4	寄存器的设计	136
6.4.1	8 位通用寄存器	136
6.4.2	串入/串出移位寄存器	136
6.5	计数器的设计	138
6.5.1	同步计数器	138
6.5.2	可逆计数器	139
6.5.3	具有数据加载和进位控制接口的同步计数器	140
6.6	状态机的设计	142
6.6.1	Moore 型状态机的设计	143
6.6.2	状态机的建立过程	144
6.6.3	状态机的复位	145

6.6.4	状态机信号输出方法	150
6.6.5	Mealy 型状态机的设计	155
6.6.6	容错状态机的设计	155
6.7	微处理器设计	157
6.7.1	微处理器设计的任务	157
6.7.2	精简指令集计算机	157
6.7.3	URISC 处理器设计	160
6.7.4	4 位微处理器设计	164
6.7.5	构造微处理器基本元件	166
6.7.6	4 位微处理器的顶层设计	174
6.8	小结	175
6.9	习题	176
第 7 章	VHDL 仿真与测试	177
7.1	概述	177
7.1.1	ModelSim 的主要特点	177
7.1.2	ModelSim 的操作步骤	179
7.1.3	ModelSim 的操作实例	180
7.2	用 VHDL 编写 testbench	182
7.2.1	简单的 testbench	182
7.2.2	八进制计数器测试平台设计	184
7.3	激励信号	187
7.3.1	具有独立激励源的 testbench	188
7.3.2	用 TextIO 输入激励信号	190
7.4	用 EDA 工具生成 testbench	194
7.5	小结	196
7.6	习题	196
第 8 章	VHDL 标准程序接口	197
8.1	VHPI 概述	197
8.1.1	VHPI 的应用	197
8.1.2	VHPI 的功能	198
8.1.3	VHPI 命名规则	199
8.1.4	VHPI 句柄	199
8.2	类继承层次和 VHPI 信息模型	200
8.2.1	类继承层次	201
8.2.2	VHPI 信息模型	201
8.3	VHPI 访问函数	202
8.3.1	信息访问函数	202
8.3.2	属性访问函数	203



8.4	VHPI 工具运行	204
8.4.1	注册阶段	205
8.4.2	分析阶段	208
8.4.3	解释阶段	208
8.4.4	初始化阶段	209
8.4.5	仿真阶段	209
8.4.6	保存阶段	209
8.4.7	重启阶段	210
8.4.8	重置阶段	210
8.4.9	终止阶段	211
8.5	VHPI 回调	211
8.5.1	回调函数	212
8.5.2	回调方法与回调属性	213
8.5.3	回调原因	214
8.6	VHPI 数值访问和更新	221
8.6.1	数值结构和类型	221
8.6.2	读取对象数值	224
8.6.3	格式化	224
8.6.4	更新对象值	225
8.6.5	驱动事务调度	229
8.7	VHPI 应用举例	230
8.7.1	概述	230
8.7.2	构建 VHPI 应用	230
8.7.3	使用 VHPI	231
8.7.4	VHPI 实例	233
8.8	小结	235
8.9	习题	236
第9章	IP 复用与 SoC 设计	237
9.1	概述	237
9.2	IP 模块的设计	238
9.2.1	硬 IP 模块设计	239
9.2.2	软 IP 模块设计	240
9.2.3	IP 模块的封装	240
9.2.4	IP 模块的复用	241
9.3	SoC 芯片设计	242
9.3.1	SoC 的设计基础	242
9.3.2	SoC 的应用设计	244
9.3.3	SoC 的芯片实现	244
9.3.4	SoC 的芯片测试	245



9.4 基于 IP 的 SoC 设计 (1) ——嵌入式微处理器系统	246
9.5 基于 IP 的 SoC 设计 (2) ——网络多核处理器实现	255
9.6 基于 IP 的 SoC 设计 (3) ——万兆网络收发器仿真	265
9.7 小结	288
9.8 习题	288
参考文献	289
附录 A VHPI 类的继承层次	290
附录 B VHPI 函数参考	294

第1章 绪 论

1.1 VHDL 概述

硬件描述语言 (Hardware Description Language, HDL) 是电子系统硬件行为描述、结构描述、数据流描述的语言。3 种描述方法形成 3 种不同的设计风格。利用硬件描述语言, 可以进行数字电子系统 SoC、FPGA 和集成电路 ASIC 的设计。

国外硬件描述语言种类很多, 有的从 Pascal 发展而来, 有的从 C 语言发展而来。有些 HDL 成为 IEEE 标准, 但大部分则是企业标准。VHDL 来源于美国军方, 其他的硬件描述语言则来源于民间的公司。这些不同的语言传播到国内, 同样也产生了不同的影响。目前在我国比较有影响的有两种硬件描述语言——VHDL 语言和 Verilog HDL 语言, 均已成为 IEEE 标准语言。

电子设计自动化 (Electronic Design Automatic, EDA) 技术的基础是描述语言、设计工具和实现器件。三者的关系是: 设计师用硬件描述语言描绘出硬件的结构或行为, 用 EDA 设计工具将这些描述编译、综合、映射成与半导体工艺无关的硬件配置文件, 半导体器件 FPGA 则是这些硬件配置文件的实现载体。当 FPGA 器件加载, 配置上不同的文件时, 这个器件便具有了不同的功能。在这一系列的设计、综合、仿真、验证、配置的过程中, 现代电子设计方法贯穿于其中。

以 HDL 表达设计意图、以 FPGA 为硬件实现载体、以计算机为设计开发环境、以 EDA 软件为开发工具的现代电子设计方法是电子设计工程师要掌握的基本技能之一。本书从应用的角度向读者介绍 VHDL 编程技术, 让读者掌握 VHDL 编程方法, 为集成电路前端设计打下基础。

1.1.1 硬件描述语言的诞生

VHDL (Very High Speed Integrated Circuit Hardware Description Language) 即超高速集成电路硬件描述语言, 下面介绍其发展的技术根源和社会根源。

1. VHDL 发展的技术根源

在 VHDL 形成之前, 已有了许多程序设计语言, 如 Assembly、C、Pascal、Fortran、Prolog 等。这些语言运行在不同的硬件平台和不同的操作环境中, 它们适合于描述过程和算法, 不适合作硬件描述。在利用 EDA 工具进行电子设计时, 逻辑图、分立电子元件作为整个电子系统的设计越来越复杂, 已不适应设计要求。一款 EDA 工具, 需要一种硬件描述语言来作为它的工作语言。众多的 EDA 工具开发者各自推出了自己的硬件描述语言。

2. VHDL 发展的社会根源

美国国防部电子系统项目有众多的承包公司。由于各公司的技术路线不一致, 许多产



品不兼容,再加上使用各自的设计语言,使得甲公司的设计不能被乙公司重复利用,造成了信息交换和维护困难。美国政府为了降低开发费用,避免重复设计,由国防部为超高速集成电路提供一种硬件描述语言 VHDL,期望它功能强大、语法严密、可读性好。政府要求各公司的合同都用该语言来描述,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、能够作为工业标准的硬件描述语言。1983 年第三季度,由 IBM 公司、TI 公司、Inter metrics 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,于 1987 年 12 月通过标准审查,VHDL 1.0 版本宣布实施,即 IEEE STD 1076-1987。从此以后,美国国防部实施新的技术标准,要求电子系统开发者的合同文件一律采用 VHDL 文档,即第一个官方 VHDL 标准得到推广、实施和普及。

1993 年,经过重新修订,发布 VHDL 2.0 版本,从而形成新的标准,即 IEEE STD 1076-1993。

2006 年,VHDL 发布 VHDL 3.0 版本,除了与老版本完全兼容外,新版本还提供了许多扩展功能,使得编写和管理 VHDL 代码更加容易。一些关键的改动包括将子标准(1164、1076.2、1076.3)并入 1076 主标准,增加了一些运算符,条件生成语句的语法更加灵活,编入了 VHPI(C/C++语言接口)和 PSL(产品规范等级)。这些改进提升了 VHDL 代码的组合能力,使测试平台更灵活,并且使 VHDL 在系统层面上的应用更加广泛。

2008 年 8 月,VHDL 4.0 版本发布,解决了 3.0 版本中出现的多个问题,包括增强的类属性。2009 年 1 月,IEEE 公布了 VHDL 4.0 的标准版本,最新的 VHDL 标准 IEEE 1076-2008 开始实施。

1.1.2 HDL 语言的种类

在国外,HDL 语言已有上百种,许多高等学校、科研单位、EDA 公司都有自己的 HDL 语言,在此简要介绍一些较有影响的 HDL 语言。

Candence 公司是一家著名的 EDA 公司,财力雄厚。该公司的 Verilog HDL 于 1983 年由 Gate Way Design Automatic(GDA)公司的 Phil Moorby 首创。他在 1984—1985 年间成功设计了 Verilog-XL 仿真器,于 1986 年提出了快速门级仿真的 XL 算法,使 Verilog HDL 语言变得更加丰富和完善,从而受到了 EDA 工具设计公司的青睐。1989 年,Candence 公司购买了 GDA 公司,Verilog HDL 语言从此变为 Candence 公司的“私有财产”,成为 Candence 公司 EDA 设计环境上的硬件描述语言。经过 Candence 公司的努力,Verilog HDL 于 1995 年成为 IEEE 标准,也是民间公司第一个硬件描述语言标准,即 Verilog HDL 1364-1995。由于 Verilog HDL 语言从 C 语言发展而来,所以有 C 语言基础的设计人员能够较快入门。

Altera 公司是一家半导体器件设计公司,其 CPLD 器件在世界市场占主导地位。该公司不仅是硬件生产厂商,也是 EDA 工具开发商,它的 EDA 工具 Quartus II 由于人机界面友好、易于使用、性能优良,而受到 FPGA、CPLD 器件设计人员的欢迎。运行在 Quartus II 环境下的 AHDL 语言具有 C 语言的设计风格,易学易用,得到众多用户的认可。

HDL 语言来自不同地方,由不同语言演变而来,因此为了各平台之间能够相互转换,



推出了 EDIF (Electronic Design Interchange Format)。EDIF 不是一种语言,其主要作用是在不同数据格式的 EDA 工具之间交换设计数据,为用户提供方便。

1.1.3 VHDL 语言的特点

VHDL 主要用于描述数字系统的结构、行为、功能和接口。除了含有许多具有硬件特征的语句外,VHDL 的语言形式、描述风格以及语法类似于一般的计算机高级语言。VHDL 的程序结构特点是将一项工程设计(可以是一个元件、一个电路模块或一个系统)分成实体(或称可视部分及端口)和结构体(或称不可视部分)。在对一个设计实体定义了外部界面后,一旦其结构体开发完成,其他的设计就可以直接调用这个实体。这种将设计项目分成实体和结构体的概念是 VHDL 系统设计的基本点。与其他硬件描述语言相比,VHDL 具有以下特点。

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、明确的源代码来描述复杂的逻辑控制。它具有多层次的设计描述功能,层层细化,最后可直接生成电路级描述。VHDL 支持时序电路、组合逻辑电路的设计,这是其他硬件描述语言所不能比拟的。VHDL 还支持各种设计方法,既支持自底向上的设计,又支持自顶向下的设计;既支持模块化设计,又支持层次化设计。

2. 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 EDA 工具都支持 VHDL,这为 VHDL 的进一步推广和广泛应用奠定了基础。在硬件电路设计过程中,主要的设计文件都是用 VHDL 编写的源代码,而由于 VHDL 语法严谨、结构清晰、参数化设计等,使其易读、易于修改和设计。

3. 完善的系统描述能力和底层描述能力

VHDL 具有多层次的设计描述功能,既可以描述系统级电路,又可以描述门级电路。而描述既可以采用行为描述、寄存器传输描述或结构描述,也可以采用三者混合的混合级描述。VHDL 支持惯性延迟和传输延迟描述,因此可以准确地建立硬件电路时序模型。VHDL 支持预定义的和自定义的数据类型,给硬件描述带来较大的自由度,使设计人员能够方便地创建高层次的系统模型。

4. 独立于器件的设计、与工艺无关

设计人员用 VHDL 进行设计时,不需要考虑制造器件的工艺,因此可以集中精力进行功能设计。当设计描述完成后,可以选用不同的器件工艺来实现其功能。

5. 很强的移植能力

VHDL 是一种标准化的硬件描述语言,同一个设计描述可以被不同的工具所支持,使得设计描述的移植成为可能。

6. 易于共享和复用

VHDL 采用基于库 (Library) 的设计方法,建立各种可复用的模块。这些模块可以预



先设计或使用以前设计中的存档模块,将这些模块存放到库中,就可以在以后的设计中进行复用,可以使设计成果在设计人员之间进行交流和共享,减少硬件电路重复设计,提高设计效率。

1.1.4 VHDL 语言的优势

VHDL 语言通常具有如下优势:

(1) 与其他的硬件描述语言相比, VHDL 具有更强的行为描述能力,从而决定了它成为系统设计领域最佳的硬件描述语言。强大的行为描述能力是避开具体的器件结构,从逻辑行为上描述和设计大规模电子系统的重要保证。

(2) VHDL 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仿真验证设计系统的功能可行性,可随时对设计项目进行仿真模拟。

(3) VHDL 语句的行为描述能力和程序结构,决定了它支持大规模设计的分解和已有设计的再利用。大规模系统高效、高速地完成,必须有多人甚至多个研发组并行工作,才能满足上市时间的要求。

(4) 对于用 VHDL 完成的设计,可以利用 EDA 工具进行逻辑综合和优化,并自动地把 VHDL 描述设计转变成门级网表。

(5) VHDL 对设计的描述具有相对独立性,设计者可以不懂硬件的结构,也不必关心最终设计实现的目标器件是什么,就可以开展独立的电路功能、系统行为设计。

1.2 VHDL 语言设计环境

1.2.1 设计工具

VHDL 语言的设计工具是集成电路设计领域的各种 EDA 工具。EDA 工程是计算机科学与技术高度发展衍生的一门崭新的电子设计自动化新学科。科学研究的方法包括实验、分析、综合、归纳等,都在 EDA 工程方法学中得到体现。EDA 工程方法学是以计算机为硬件平台,以 EDA 软件工具为工作环境的现代电子设计方法学。EDA 工具设计软件开发需要计算机软件工程专家、半导体工艺专家、电路系统专家合作完成。EDA 工具的研发是知识密集型产业,EDA 工程研究的最终成果是借助于大量功能丰富的软件工具开发出来的,提供了集成电路设计的基础条件。

EDA 工具种类繁多,不同的设计环节可以使用不同的设计工具。同一设计功能的 EDA 工具由不同的 EDA 工具开发商提供,质量也有所不同。按设计阶段,EDA 工具可分为前端设计工具和后端设计工具;按设计电路的性质,可分为数字电路设计工具和模拟电路设计工具;按设计功能,可分为综合工具、仿真工具、时序分析工具、布局布线工具、电磁兼容分析工具、功耗分析工具等。前端设计有与制造工艺无关的 IP 库,后端设计则有与设计规则有关的工艺库。这些形形色色的 EDA 工具在集成电路设计中分别起到不同的作用。从根据市场需求进行设计的功能定义、设计输入、逻辑综合、系统划分、功能仿真、布图



规划,到根据器件工艺设计规则进行的版图设计(布局、布线)、寄生参数提取(时序分析)、版图后仿真,将这些设计步骤按设计顺序组织起来,就构成了集成电路设计流程,如图 1.1 所示。

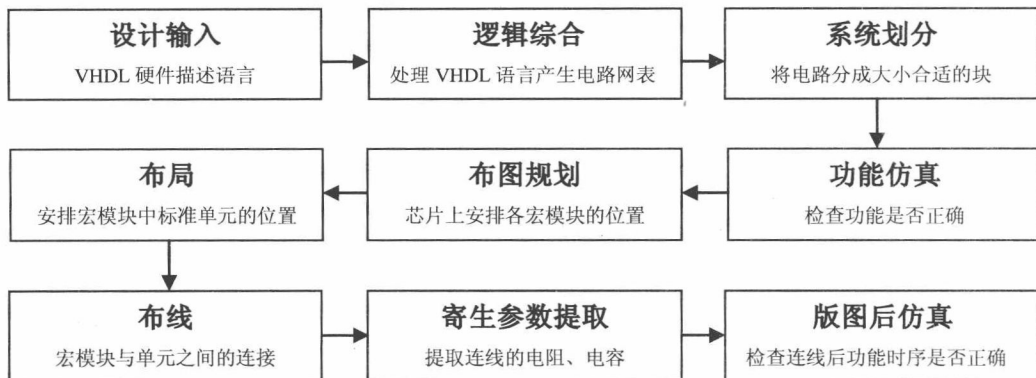


图 1.1 用 VHDL 语言设计集成电路的流程

将这些设计环节中所用到的设计工具组织起来,就构成了集成电路设计环境,如图 1.2 所示。



图 1.2 集成电路设计环境

VHDL 语言输入界面、编辑界面、编译工具、器件库、函数库的组合称为 VHDL 设计环



境。许多 EDA 工具均提供了 VHDL 设计环境, 如 Altera 公司的 Quartus II, 其界面如图 1.3 所示。

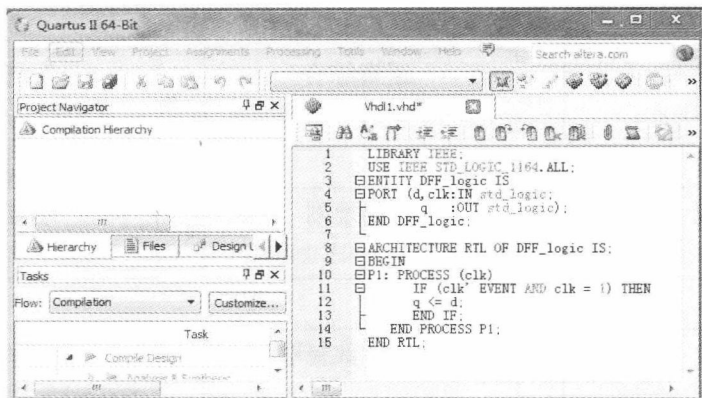


图 1.3 利用 Quartus 工具的 VHDL 语言设计环境

Xilinx 公司的 ISE、Synopsys 公司的 Compiler 等也都可以用来进行 VHDL 设计。用这些 EDA 工具设计 VHDL 语言程序, 编译生成的文件具有不同的用途: Quartus II 工具、ISE 工具面向 FPGA 设计, Synopsys 的工具则面向 ASIC 集成电路设计。

用户只需选择一款 EDA 工具安装在计算机中, 然后按 EDA 工具的操作说明, 即可开展 VHDL 语言的编写、编辑和编译工作。

1.2.2 仿真工具

使用 EDA 工具的最大优点是可以利用仿真工具来验证设计是否正确。各公司的 EDA 工具都有仿真功能, Cadence 公司的 Verilog-XL (见图 1.4)、NC-Verilog 用于 Verilog 仿真, leapfrog 用于 VHDL 仿真, Analog Artist 用于模拟电路仿真。

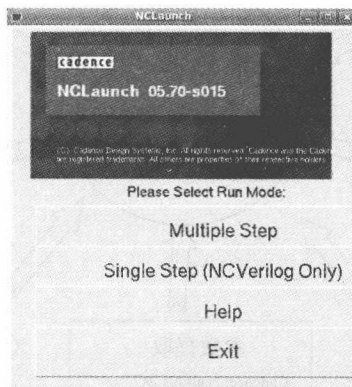


图 1.4 Verilog-XL 仿真工具入口界面

Mentor 公司的 ModelSim 是业界最优秀的 HDL 语言仿真软件, 它提供了友好的仿真环境, 具有强大的调试能力, 并提供了一个丰富、易用的图形用户界面和调试环境, 是业界唯一单内核支持 VHDL 和 Verilog 混合仿真的 PC 版 HDL 语言仿真器。ModelSim 采用直接