电子设计自动化技术

李　平　编著

电子科技大学出版社

图书在版编目（CIP）数据

电子设计自动化技术 / 李平编著. —成都：电子科技大学出版社，2008. 5

高等学校教材

ISBN 978–7–81114–793–3

Ⅰ. 电… Ⅱ. 李… Ⅲ. 硬件描述语言，VHDL－程序设计－高等学校－教材 Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2008）第 049523 号

内 容 简 介

本书是编著者结合多年的电子设计自动化（EDA）教学实践和二十多年电子设计与芯片设计经验编著而成。该教材具有线条清晰、深入浅出、易学易懂的特点，使读者能够较快地理解并掌握基于硬件描述语言 VHDL 的 EDA 设计方法。

本书的主要内容包括：EDA 设计方法与技能、VHDL 程序的结构及其描述方式、VHDL 程序的编译与仿真、VHDL 语言规则、VHDL 的主要描述语句、VHDL 的设计共享、组合逻辑电路设计、时序逻辑电路设计、逻辑系统的状态机设计、VHDL 设计录入与仿真调试方法、VHDL 的 FPGA 实现方法、VHDL 的 ASIC 实现方法等。

本书适用于高等院校电子信息类专业的高年级本科生和研究生，也可供工程技术人员参考。

高等学校教材

电子设计自动化技术

李 平 编著

出 版： 电子科技大学出版社（成都市一环路东一段 159 号电子信息产业大厦 邮编：610051）
策划编辑： 郭蜀燕
责任编辑： 郭蜀燕 黄礼玲
主 页： www.uestcp.com.cn
电子邮箱： uestcp@uestcp.com.cn
发 行： 新华书店经销
印 刷： 成都蜀通印务有限责任公司
成品尺寸： 185 mm×260 mm **印张** 17.25 **字数** 420 千字
版 次： 2008 年 5 月第一版
印 次： 2008 年 5 月第一次印刷
书 号： ISBN 978–7–81114–793–3
定 价： 32.00 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话：028-83202463；本社邮购电话：028-83208003。
◆ 本书如有缺页、破损、装订错误，请寄回印刷厂调换。
◆ 课件下载在我社主页“下载专区”。

前　　言

电子设计自动化（EDA）方法的诞生和推广应用是 20 世纪 90 年代在数字电路设计方法上的一次革命性变革。随着步入 21 世纪，我们已经进入了单片集成系统（SoC，System on a Chip ）时代。

如果说电子管把人类带入了数字时代，那么晶体管就是固体化的电子管。集成电路的思路是用 10 只晶体管卖 1 只晶体管的价格。那么 SoC 的思路是什么？SoC 的思路就是把 10 颗大规模集成电路芯片（VLSI）当做一颗 VLSI 芯片销售。其本质与集成电路的思想相同，即：通过扩大应用取得经济效益，伴随而来的是技术进步的巨大社会效益。

从某种意义上来说，SoC 的出现，在学术和学科的影响已经大大超过了晶体管和集成电路。在晶体管和集成电路时代，电子系统技术和晶体管/集成电路技术基本上是割裂的、不相关的。而在 SoC 时代，人们不禁要问：SoC 到底是 S 还是 C？是学 S 还是学 C？SoC 模糊了电子系统与基础元器件的界线，S 的进步意味着优秀的芯片设计的出现，而一颗新型芯片的诞生则宣告又一次电子系统的升级。

EDA 技术是进入 SoC 殿堂的钥匙，不管是学 S 还是学 C，只有拥有这把钥匙，才能进入 SoC 的殿堂。

什么是 EDA 技术？简而言之，EDA 技术就是利用 EDA 软件把描述电路与系统的语言自动地转换成电路，然后再把电路自动转换成芯片。电路与系统的 EDA 设计方法是自顶向下的设计方法。它从协议、标准等高层次和抽象的概念开始电路与系统的设计，并按层次从高到低的方法进行验证和管理，能够设计共享和重复使用。这样大大提高了处理复杂电路与系统设计的能力，设计所需的周期也大幅度缩短。

描述电路与系统的语言称为硬件描述语言（HDL）。HDL 的使用使得电路与系统的设计内容和方式发生了巨大的改变。传统的、繁琐的布尔代数化简和卡洛图分析等设计方法逐渐被淘汰。电路与系统设计不再是从画电路图开始，而是从编写 HDL 代码开始。

HDL 的设计方式使数字电路的设计变得更加快捷和容易。掌握 HDL 是现代电路与系统设计工程师、集成电路（IC）设计工程师必备的专业技能，是掌握 EDA 设计方法的基础。

目前成为工业标准的硬件描述语言主要有两种，即 VHDL 和 Verilog HDL。20 世纪 80 年代末，它们在国外诞生并成为 IEEE 标准。它们在国内推广应用的时间比国外晚，但近些年来，硬件描述语言在国内的推广应用已经非常广泛。

以上两种硬件描述语言具有以下共同特点：能抽象表示电路的结构和行为，支持逻辑设计中层次描述；用简单的语句来描述复杂的电路；具有电路仿真与验证机制；支持电路描述由高层到低层的综合转换。

相比较而言，VHDL 在系统级抽象方面强于 Verilog HDL，而且采用 FPGA/CPLD 作为物理实现的设计者更加愿意采用 VHDL。目前主流的 EDA 工具均支持 VHDL。

鉴于 EDA 技术的重要性，“电子设计自动化技术”课程所教授的知识和技能已成为很多用人单位录用的必要条件，该课程已成为很多学生的“饭碗课程”，故本教材更加强调知识的完备性，安排了“VHDL 的 FPGA 实现”和“VHDL 的 ASIC 实现”等与实验相关的内

容。相信通过上述实验，学生可以更牢固地掌握课程的相关知识和技能。

EDA 的基础是 VHDL，作者希望读者在 VHDL 学习的过程中，既见树木又见森林，避免拘泥于具体 VHDL 语法规定的细枝末节中。本书首先通过对 VHDL 程序架构的清晰表述和对 VHDL 语句的合理归类，使得读者对 VHDL 的学习变得较为容易；其次本书抓住能力培养这个关键，通过 EDA 工具使用和面向物理实现的 VHDL 实践来锻炼学生的 EDA 设计能力。

本书每章之后配有一定的习题，主要是为了启发学生对该章内容的思考，帮助掌握所学内容，其难度较低。本书末还配有一定的课程测试题和历年的试题，目的是便于学生自我检验。书末所配 EDA 竞赛题具有一定难度，希望给学有余力的学生提供深入探究的思路和方向，体现学无止境。

编写本书的宗旨是为高等学校高年级本科生和研究生提供一本电子设计自动化（EDA）技术的专门教材，同时也为电路与系统工程师和集成电路设计工程师提供一本专业参考书。

本书编写过程中引用了许多学者的著作，在此向他们表示衷心的感谢。特别感谢电子科技大学出版社郭蜀燕为本书出版所做的工作。在本书的统编工作中杜涛、王靖、谢小东、阮爱武、胡滨、罗和平、廖永波、李夏、王劲等做了大量艰苦工作，在此深表谢意！

书中若有不妥之处，恳请广大读者提出宝贵意见和建议。

作　者

目　录

第一章　电子设计自动化（EDA）与硬件描述语言（HDL）.....1
1.1　TOP-DOWN 设计方法.....1
1.1.1　TOP-DOWN 设计的主要阶段.....1
1.1.2　TOP-DOWN 设计方法的特点.....2
1.1.3　TOP-DOWN 设计方法的优势.....4
1.1.4　FPGA/CPLD 与 ASIC 两种物理实现.....4
1.2　硬件描述语言（HDL）.....6
1.2.1　VHDL 的特点.....6
1.2.2　VHDL 的应用及 IP 核.....7
习题.....8
第二章　VHDL 程序的结构及其描述方式.....9
2.1　VHDL 程序的结构.....9
2.1.1　实体（ENTITY）.....10
2.1.2　构造体（ARCHITECTURE）.....12
2.1.3　VHDL 程序的基本构成格式.....14
2.2　VHDL 描述方式.....16
2.2.1　行为级（Behavior Level）描述.....16
2.2.2　寄存器传输级（RTL）描述.....19
2.2.3　结构级（Structural Level）描述.....21
习题.....22
第三章　VHDL 程序的编译与仿真.....23
3.1　EDA 仿真工具简介.....23
3.2　VHDL 程序的编译与仿真.....24
习题.....32
第四章　VHDL 语言规则.....33
4.1　标识符.....33
4.2　数据对象.....33
4.2.1　信号.....33
4.2.2　变量.....37
4.2.3　常量申明.....38
4.3　数据类型.....39
4.3.1　标准数据类型.....39
4.3.2　用户自定义数据类型.....41

4.3.3 用户自定义子类型 44
4.3.4 数据类型转换 45
4.4 操作符 46
习题 50
第五章 VHDL 的主要描述语句 51
5.1 概述 51
5.2 并行信号赋值语句 52
5.3 进程（PROCESS） 54
5.3.1 显式进程 54
5.3.2 隐式进程 58
5.3.3 进程的执行 59
5.4 顺序描述语句 64
5.4.1 IF 语句 64
5.4.2 CASE 语句 66
5.4.3 LOOP 循环语句 68
5.5 子程序（SUB-PROGRAM） 73
5.5.1 函数（FUNCTION） 74
5.5.2 过程（PROCEDURE） 75
5.6 块（BLOCK） 77
5.7 断言语句（ASSERT） 78
5.8 元件语句（COMPONENT） 79
5.8.1 元件申明 80
5.8.2 元件调用 80
5.8.3 元件配置 82
5.9 生成语句（GENERATE） 85
5.9.1 FOR-GENERATE 语句 85
5.9.2 IF-GENERATE 语句 86
习题 88
第六章 VHDL 的设计共享 89
6.1 程序包（PACKAGE） 89
6.2 库（LIBRARY） 90
6.3 配置（CONFIGURATION） 93
6.3.1 实体与构造体的连接配置 93
6.3.2 层与层的连接配置 96
习题 99
第七章 组合逻辑电路设计 102
7.1 组合逻辑电路 102
7.2 基本门电路 102

7.2.1　二输入与门 102
7.2.2　二输入或门 104
7.2.3　二输入与非门 105
7.2.4　二输入或非门 106
7.2.5　非门 107
7.2.6　二输入异或门 109
7.2.7　多输入门电路 110
7.3　编/译码器 112
7.3.1　3-8 译码器 112
7.3.2　优先级编码器 114
7.4　多路选择器 116
7.5　比较器 117
7.6　减法器 118
7.7　乘法器 119
7.8　交通信号灯监测电路 121
7.9　存储器 122
7.9.1　存储器的初始化 123
7.9.2　ROM 的描述 123
7.9.3　RAM 的描述 125
习题 127
第八章　时序逻辑电路设计 128
8.1　时序逻辑电路 128
8.2　时序逻辑设计基础 128
8.2.1　时钟信号描述 128
8.2.2　同步复位和异步复位 132
8.3　触发器 133
8.3.1　D 触发器 133
8.3.2　T 触发器 137
8.3.3　RS 触发器 139
8.4　寄存器 141
8.4.1　锁存寄存器 141
8.4.2　移位寄存器 141
8.5　计数器 144
8.5.1　同步计数器 144
8.5.2　同步可逆计数器 145
8.5.3　异步计数器 146
8.6　分频器 147
习题 149
第九章　VHDL 逻辑系统的状态机设计 150

9.1 有限状态机概述 150
9.2 有限状态机的描述方式 151
9.3 状态机的状态编码 153
9.4 状态机设计实例 155
9.4.1 Moore 型状态机设计 155
9.4.2 Mealy 型状态机设计 158
习题 159
第十章 VHDL 设计录入与仿真调试方法 161
10.1 设计录入 161
10.1.1 HDE（HDL 编辑器）录入方法 162
10.1.2 BDE（框图编辑器）录入方法 167
10.1.3 FSM（状态图编辑器）录入方法 174
10.2 仿真 183
10.2.1 图形化界面手动加激励仿真 183
10.2.2 编写宏文件（*.do）加激励仿真 187
10.2.3 编写测试向量（Test Bench）加激励仿真 191
习题 197
第十一章 VHDL 的 FPGA 实现 198
11.1 概述 198
11.2 VHDL 的 FPGA 逻辑综合 199
11.3 VHDL 的 FPGA 物理实现 204
习题 207
第十二章 VHDL 的 ASIC 实现 208
12.1 概述 208
12.2 VHDL 的 ASIC 逻辑综合 210
12.3 ASIC 自动布局布线 220
习题 241
附录 A 《电子设计自动化技术》课程测试题 242
附录 B 历年试题 247
附录 C 全国 EDA 大赛试题集锦 256
附录 D VHDL 保留字 263
附录 E EDA 工具软件一览表 264
附录 F 部分 FPGA 厂家名录 267
参考文献 268

第一章　电子设计自动化（EDA）与硬件描述语言（HDL）

本章读导　通过本章学习，理解什么是正向设计方法，HDL 在 EDA 设计中的作用，VHDL 的特点，VHDL 的应用；应当形成这样的概念：VHDL 是 EDA 设计的基础，学习 VHDL 的目的是掌握一种电路与系统设计的先进设计方法。

1.1　TOP-DOWN 设计方法

电子设计自动化（EDA）一般采用自顶向下（TOP-DOWN）的设计方法，通常也称为正向设计，它是针对传统的自底向上（BOTTOM-UP）的设计方法而提出来的。

自底向上的设计方法是从已存在的单元出发进行电路或系统设计。例如，对已有的芯片进行解剖、照相、逻辑提取、仿真验证并最后照原样绘制版图。这种方法对模拟集成电路和较小规模数字电路设计，不失为一种可行的办法。而且这种方法对于学习别人的成功经验，特别是在集成电路布局布线方面的经验有一定的好处。但是，随着集成电路规模的扩大，特别是对于深亚微米 5 层以上金属布线的超大规模集成电路，这种方法正在失去其效能。一方面，由于对集成电路芯片进行解剖、分析的难度和成本越来越高；另一方面，解剖、分析失败的风险也愈来愈大。随着我国集成电路布局布线保护条例的实施，这种方法还会在形成产品时遇到知识产权方面的问题。

所谓自顶向下的设计过程是指从系统硬件的高层次抽象描述向底层物理描述的一系列转换过程。自顶向下的设计由功能级、行为级描述开始；寄存器传输（RTL）级描述为第一个中间结果；再将 RTL 级描述由逻辑综合得到网表（Net-list）或电路图；利用 EDA 工具将网表自动转换成目标文件下载到现场可编程门阵列（FPGA，Field Programmable Gate Array）/复杂可编程逻辑器件（CPLD，Complex Programmable Logic Device）或通过自动布局布线设计成专用集成电路（ASIC，Application Specification Integrated Circuit），从而得到电路与系统的物理实现。相对自底向上的设计而言，自顶向下的设计对底层单元的选取更加灵活，因此有利于更好地发挥设计人员的创造性，从而形成具有自主知识产权的芯片产品。

1.1.1　TOP-DOWN 设计的主要阶段

TOP-DOWN 设计分为行为级描述、寄存器传输（RTL）级描述、逻辑综合和物理实现四个阶段，如图 1.1 所示。

第一阶段是行为级描述。行为级描述实质上是对整个系统的数学模型的描述。在行为级描述阶段，并不考虑实际的操作和算法用什么方法实现，考虑更多的是采用什么样的系

统框架使设计达到系统设计规划书（设计目的）的要求。在这一阶段，典型描述方法是有限状态机、控制流图、数据流图和控制数据流图。

第二阶段是寄存器传输（RTL）级描述。RTL 描述也被称为数据流级描述，顾名思义，在这一阶段要考虑具体的功能模块中从输入端到输出端的数据运算通路，数据的运算通路的最基本的设计单元是寄存器、计数器、多路选择器、算术逻辑单元（ALU）等。要想通过综合工具得到硬件的具体实现，必须将抽象程度很高的行为级描述的 VHDL 程序改写为 RTL 级描述的 VHDL 程序。

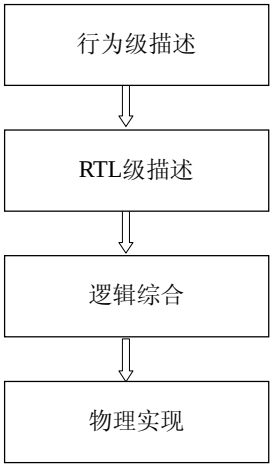


图 1.1　TOP-DOWN 设计层次

在完成编写 RTL 方式描述的 VHDL 源程序以后，再用仿真工具软件对程序进行仿真。如果这一步的仿真通过了，那么就可以利用逻辑综合工具软件进行综合了。

第三阶段是逻辑综合。逻辑综合这一阶段是利用逻辑综合工具软件，将高层次的与工艺无关的 RTL 方式描述的 HDL 源程序转换成低层次的与特定工艺相关的用基本逻辑元件表示的文件（门级网表）。在逻辑门设计时，电路的基本单元通常是与门、或门、异或门、倒相器、D 触发器、锁存器等逻辑单元。如果需要，可以将逻辑综合的结果以原理图方式输出。为了分析整个设计的性能，需要使用仿真工具软件、时序分析工具完成门级仿真和时序验证。

第四阶段为物理实现。物理实现分为 FPGA/CPLD 和 ASIC 两种。FPGA/CPLD 物理实现在元件库文件支持下，ASIC 物理实现在工艺库文件支持下，利用 EDA 工具，将门级网表作为输入，进行自动布局布线最终生成 FPGA/CPLD 的目标文件或 ASIC 的版图文件。接下来，还需对布局布线后的结果进行寄生参数提取和后仿真，以便验证布局布线后，寄生参数（寄生电阻、电容等）的存在对所设计电路功能的影响。如果前仿真和后仿真结果一致，则由 FPGA/CPLD 目标文件下载到 FPGA/CPLD 得到设计结果；或由 ASIC 的版图（GDS-II）文件送交制版，由代工厂（Foundry）利用光刻掩膜版对硅片进行一系列加工得到设计结果。如果由于寄生效应使后仿真与前仿真不一致，则必须对前端设计进行必要的修改，然后重新进行后仿真，直到前仿真和后仿真结果一致。

1.1.2　TOP-DOWN 设计方法的特点

完整的 TOP-DOWN 设计流程如图 1.2 所示，TOP-DOWN 设计从总体行为设计、寄存器传输（RTL）级描述，经过逻辑综合得到网表，最终得到物理实现为止。四个阶段都要分别进行仿真，即行为级仿真、RTL 仿真、门级仿真和后仿真。四级仿真贯穿系统硬件设计全过程。上一阶段的仿真结果为下一阶段的仿真提供有用信息。因此，在系统设计早期就能发现设计中存在的问题，并尽可能在早期设计阶段就解决这些问题。这是 TOP-DOWN 设计方法的突出特点之一。

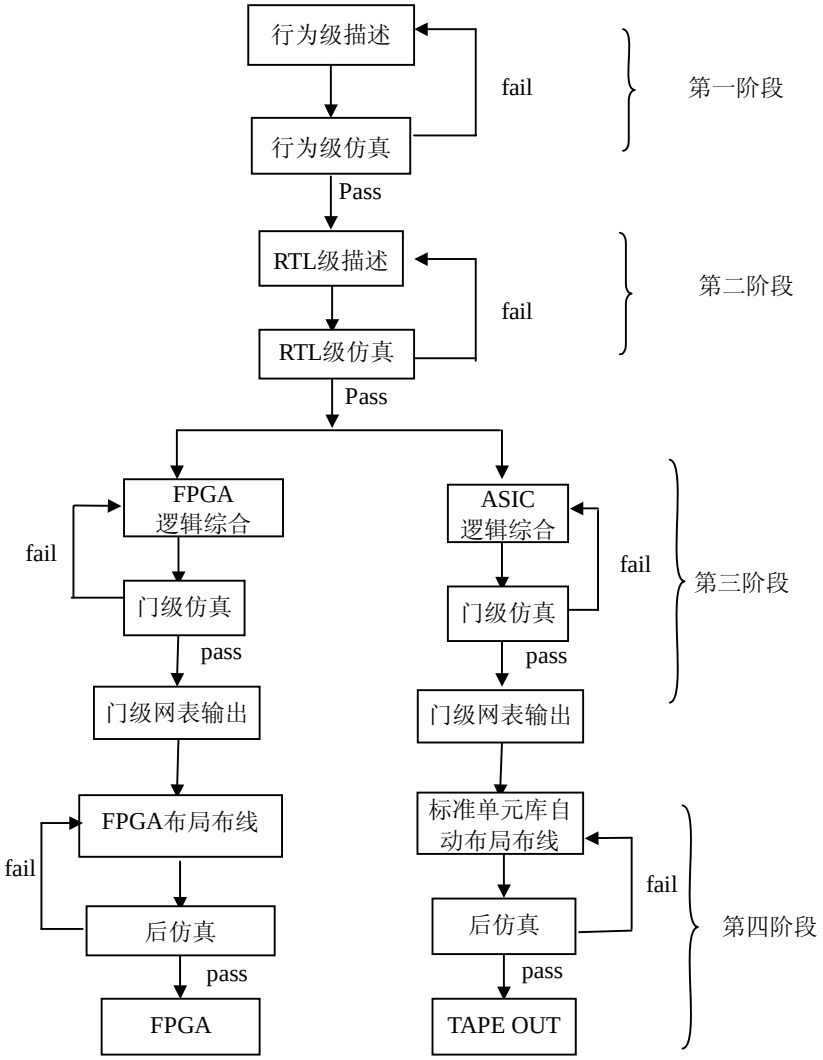


图 1.2　TOP-DOWN 设计流程

进行一项新的电路与系统设计，设计伊始，极有可能在行为级仿真时就不能通过，这说明设计者对所要设计的电路与系统的功能理解有误，必须重新认识该电路与系统。经反复修改、调试，直到行为级仿真通过，验证了设计者对电路与系统的认识的正确性。此后，RTL 级仿真可能通不过，此时，问题只可能出现在对信号流向、时序的认识方面，而不用再怀疑对系统的认识是否正确。同理，此后若门级仿真通不过，则只需查看与门级延时等相关的问题。最后，后仿真通不过时，只排查寄生参数带来的延时问题即可。

这正是自顶向下设计与自底向上设计的不同之处。可以想象，当采用自底向上设计时，所面临的问题将包括：对系统的认识、信号流走向、时序问题、门级延时、寄生参数等所有问题。当电路与系统变得越来越复杂时，这些问题的解决会变得十分困难，甚至无法解决。这正是自顶向下设计方法日益成为主流设计方法的重要原因。

TOP-DOWN 设计的特点之二是自动化。如在上述 TOP-DOWN 设计的第三阶段，由 EDA

工具自动完成综合。又如在上述 TOP-DOWN 设计的第四阶段，由 EDA 工具自动烧录 FPGA/CPLD 或进行自动布局布线产生集成电路版图。现代的 EDA 设计一般均采用了如上所述的 TOP-DOWN 设计方法。TOP-DOWN 设计方法的特点和优势其实直接体现于其设计步骤中。

1.1.3 TOP-DOWN 设计方法的优势

由于 TOP-DOWN 设计方法的上述特点导致了 TOP-DOWN 设计具有以下显著优势：

- 在系统设计早期发现设计中存在的问题，提高了设计的一次成功率。
- 大大缩短系统设计的周期，减少系统开发所耗用的时间。
- 易于系统划分和项目管理，使几十万门乃至几百万门的大规模复杂数字电路的设计成为可能。
- 设计效率提高，可减少设计人员。
- 通过设计共享，避免重复设计。

1.1.4 FPGA/CPLD 与 ASIC 两种物理实现

EDA 设计的内容最终需要由 FPGA/CPLD 或 ASIC 来实现。下面分别介绍 FPGA/CPLD 与 ASIC 两种物理实现的特点。

FPGA 和 CPLD 是由加工厂完成了集成电路工艺制造和封装测试的可编程集成电路成品。可编程（Programmable）的含义是设计工程师自己就可以通过 EDA 软件将门级网表转为目标文件，再将目标文件通过烧录器烧录至 FPGA/CPLD 上，改变其内部配置从而实现自己所需要的功能。

1．FPGA/CPLD 的优点

① 资金投入小、风险小

FPGA 芯片在出厂之前都做过百分之百的测试，不需要设计人员承担投片风险和费用，设计人员只需在自己的实验室里就可以通过相关的软硬件环境来完成芯片的最终功能设计。所以，开发 FPGA／CPLD 的资金投入小，节省了许多潜在的花费。

② 开发周期与面市时间（Time to Market）短

系统工程师可以反复地编程、擦除、使用芯片或者在外围电路不动的情况下用不同的配置文件就可实现其不同的功能。所以，用 FPGA/CPLD 试制样片，能以最快的速度改进系统设计，验证系统功能，从而达到缩短开发周期的目的。

③ 改动灵活

FPGA/CPLD 软件包中有各种输入工具、仿真工具、编程器及烧录器等全线产品，电路设计人员在很短的时间内就可完成电路的输入、编译、优化、仿真，乃至最后芯片的制作。当电路有少量改动时，更能显示出 FPGA／CPLD 的优势。

④ 易学易用

电路设计人员使用 FPGA／CPLD 进行电路设计时，不需要具备专门的集成电路深层次的知识。FPGA／CPLD 软件易学易用，可以使设计人员更能集中精力进行系统设计、电路设计。

2．FPGA/CPLD 的缺点

① 芯片成本高、价格贵

FPGA／CPLD 器件可写、可擦、可编程的特点也使其付出了代价，即它们的芯片成本高、价格贵，一般不适合在整机产品中大规模使用。

但是，在通信产品中有例外。有两方面原因使然，其一，由于通信产品的利润率较高，可以接受 FPGA/CPLD 的较高成本；其二，由于通信产品的协议经常升级换代，FPGA/CPLD 可以方便地重新写入。

② 嵌入电路困难

不能由电路与系统工程师嵌入传感电路、模拟电路或功率输出电路。

3．ASIC 的优缺点

专用集成电路 ASIC 是针对特定用户的集成电路。在 EDA 设计中不能像 FPGA/CPLD 那样由设计者通过硬件编程加以实现。必须由加工厂根据版图制作掩膜版（Mask，又称光刻版），并进行一系列集成电路工艺加工（简称流片或工艺流片）得到芯片（Chip），Chip 经封装测试后成为最终的 ASIC 产品。设计者的 VHDL 程序经综合后生成网表，通过 EDA 工具产生 ASIC 的版图，按照版图加工出的芯片具有设计者所要求的功能。对 ASIC 设计而言，EDA 过程从 VHDL 描述开始，到产生版图数据为止。

1）ASIC 的优点

ASIC 的优点正好是 FPGA／CPLD 的缺点。

① 价格优势

由于采用半导体平面工艺，几十只与几万只 ASIC 芯片的研制、生产成本几乎是一样的，因此它特别适合大规模使用。用量愈大，价格愈低，在大规模使用时 ASIC 具有价格优势。

② 频率优势（性能优势）

ASIC 具有频率优势。在半导体工艺条件相同前提下，ASIC 芯片的工作频率高于 FPGA 芯片。

③ 嵌入优势

ASIC 可以由电路与系统工程师嵌入传感电路、模拟电路或功率输出等外围电路。这些电路加上核心的数字电路，如微控制器（MCU）核，则可构成片上系统（SoC，System on a Chip）。

值得指出的是：采用 ASIC 实现方式并不是像看上去那样困难和价格昂贵。版图数据文件（GDS-II）可以送交多项目晶圆（MPW，Multi Project Wafer）组织，两个月后即可获得加工好的几十只 ASIC 样品。更重要的是 Mask 和工艺流片的费用经过多个用户分摊后大大降低。

2）ASIC 的缺点

ASIC 的缺点正好对应 FPGA／CPLD 的优点。

① 资金投入大。如果不采用 MPW 加工方式，Mask 和工艺流片费用都是昂贵的。

② 研发、流片都具有较高的失败风险，而 FPGA／CPLD 完全没有这种风险。

③ 面市时间较长。ASIC 设计完成后，一般还需要两个月左右的加工时间，而 FPGA／CPLD 设计完成后只需现场烧录就可以得到设计结果。

④ 改动困难。一旦改动，意味着新的 Mask 制版费、工艺流片费用和失败风险。

需要提到的是：随着 ASIC 芯片加工工艺最小线宽的不断缩小，光刻版费用迅速增加。一套光刻版的制作费用随工艺最小线宽变化如表 1.1 所示。

表 1.1　光刻版制作费用一览表

工艺最小线宽	0.5μm	0.35μm	0.18μm	0.13μm	0.09μm
一套 Mask 费用（单位：万元人民币）	10～15	30～40	60～70	110～130	约 400

从以上分析可以看出，FPGA/CPLD 和 ASIC 各有优势和不足。目前，两者均在努力扩大自己的优势，FPGA 厂商正努力在 FPGA 中嵌入 ASIC 形式的 IP 核；另一方面，ASIC 厂商也正在采用各种可重构技术，包括嵌入 FPGA 技术，以期增加产品的灵活性和适应性。

1.2　硬件描述语言（HDL）

TOP-DOWN 设计方法的发展主要归功于硬件描述语言（HDL，Hardware Description Language）的成熟和电子设计自动化（EDA）工具的进步。

硬件指所要设计的电路与系统。这种称谓本身揭示了 HDL 所描述的内容最终是要通过硬件来实现。

硬件的描述方式是多种多样的。例如集成电路的版图是对集成电路的一种描述，它真实、物理地给出了芯片内的结构详情；又如大家所熟悉的电路图也是一种对电路或系统的描述，它确切地描绘出了一个硬件的电路结构。

目前人们所说的 HDL，通常特指电子技术高层设计阶段中所采用的硬件描述语言。这样的 HDL 有如下特点：

1．HDL 以行为级描述见长，它能从比较抽象的角度描述电子实体的行为，能够进行系统早期仿真；

2．HDL 能够进行结构化描述，它能从具体的角度描述电子实体的结构，便于存档，便于共享；

3．HDL 具备了从比较抽象到比较具体的多个层面上对电子实体进行混合描述的能力，降低了硬件电路设计难度；

4．HDL 的生命力在于用它描述的实体的程序，既能被仿真，又能被综合。通过仿真（simulation）,可验证设计的正确性；通过综合（synthesis），抽象的设计描述将自动地自上而下转化为实在的物理设计——逻辑图、电路图，直至 FPGA 或 ASIC 版图。

硬件描述语言是高层次、自动化设计的起点和基础。

1.2.1　VHDL 的特点

目前世界上存在许多 HDL，其中最流行的为 VHDL 和 Verilog HDL 两种。

VHDL 的头一个字母 V 代表 Very High Speed Integrated Circuit，即 VHSIC，所以 VHDL 最初是超高速集成电路硬件描述语言。其诞生的背景是美国的 VHSIC 计划，即 1980～1986 年，美国国防部组织的以国防电子系统的要求为目标的集成电路研究计划。

人们发现 VHDL 能够满足各种数字电路设计要求，可以作为一种通用的硬件描述语言

工业标准。国际电气与电子工程协会 IEEE 参与了对它的标准化。经广泛征求意见，融合其他 HDL 的优点，1987 年 12 月由 IEEE 正式推出了版号为 IEEE—1076 的第一个 VHDL 工业标准版本，1993 年推出更新版 VHDL 工业标准版本。

VHDL 具有以下显著优点：

1．通用性好，支持面广。因为它是工业标准，故受到普遍支持，凡大型 EDA 软件都推出支持 VHDL 的设计环境。因此用 VHDL 描述的设计文件，可采用不同的设计工具。

2．重用性好。VHDL 的描述与具体工艺无关，因而适用面宽。VHDL 的设计模块便于在不同设计场合重复使用。

3．可靠性好。VHDL 文件兼技术文档与实体设计于一身，可读性好，既是技术说明，又是设计实现，这就保证了二者的一致性。

4．与 Verilog HDL 相比较，VHDL 以行为级描述见长，即以抽象的角度描述电子实体的行为的能力更强。

客观地讲，上述 4 条 VHDL 优点中的前 3 点基本上是 HDL 的共同优点，而其中第 4 条是 VHDL 的特殊的优点。需要指出的是：在学习和使用 VHDL 时应当特别关注并利用其抽象描述电子实体行为能力强的特点。VHDL 抽象描述电子实体行为能力强意味着人们可以利用它来较迅速地获得对电路与系统的正确描述。

1.2.2　VHDL 的应用及 IP 核

VHDL 所描述的内容最终是要通过硬件实现的。传统的 VHDL 的应用方式是获得两种物理实现之一，即 FPGA／CPLD 或 ASIC。但是，目前 VHDL 的应用具有一种新的形式，即由 VHDL 形成知识产权核或称 IP 核（Intellectual Property Core）。IP 核逐渐以商品的形式出现在市场上，而且正以每年 30%以上的速度增长，VHDL 在这方面的应用愈来愈多。

比如，FPGA／CPLD 厂商在 FPGA／CPLD 中预先嵌入 IP 核，如微控制器（MCU），数字信号处理器（DSP），典型模拟电路，模/数（A/D）转换器，数/模（D/A）转换器等，为 FPGA／CPLD 增加了功能。FPGA／CPLD 正在向可编程片上系统（SoPC）方向发展。

ASIC 则强调采用 IP 复用（IP Reuse）设计方法来缩短其面市时间，降低其投片试制风险。

上述两个方向的技术发展都与新兴的 IP 核技术有关。IP 核可分为软核、固核、硬核三种。

软核（Soft IP Core）是指前面所说的 EDA 芯片设计的第二阶段，即寄存器级（RTL 级）的设计结果，且经过了 RTL 级仿真验证。

固核（Firm IP Core）是指我们前面所说的 EDA 芯片设计的第四阶段的结果，即经过了 FPGA 实物验证的设计结果。

硬核（Hard IP Core）同样是所说的 EDA 芯片设计的第四阶段的结果，但特别针对经过了 ASIC 工艺验证的设计结果。

很显然，从 FPGA 的角度看固核最有价值。而从 ASIC 角度看则硬核最有价值，因为要将固核转化成硬核具有我们已提到过的投片风险。软核则由于它与工艺和器件均无关，具有高度的灵活性，从而具有其独特的价值。作为电路与系统或 IC 设计工程师和研究人员，必须具备读懂和研发软核、固核的能力。

软核通过仿真加以验证，固核则通过 FPGA/CPLD 加以硬件验证。在大规模集成电路（VLSI）设计过程中，常常在完成软件仿真以后，用 FPGA/CPLD 硬件测试作为 ASIC 硬件仿真手段或功能验证手段。这是因为 FPGA/CPLD 内部已经考虑了与 ASIC 内部相似的门级延时和连线延时。

在工程实践中，往往存在这样的情况，即在电子产品数量较少的初期阶段（一般是在产品数量小于 1 万只时），采用 FPGA/CPLD，而当电子产品数超过 10 万只以后，则考虑将所使用的 FPGA/CPLD 转化成 ASIC。从 IP 的角度看，就是将固核转化成硬核，简称为“固核硬化”。固核硬化仍然是存在一定的投片失败的风险的。这是因为 FPGA/CPLD 与 ASIC 内部的门级延时和连线延时只是相似，不是相等。固核硬化的设计，需要使用本书第十二章介绍的自动布局布线 APR 设计流程。

与上述“固核硬化”对应，在集成电路行业，还存在“硬核软化”的需要。这是因为：硬核必须通过“交换”或出售才能实现价值最大化。而在硬核“交换”过程中，为了保护硬核的知识产权，IP 提供者往往将硬核以黑匣子形式提供给用户。同时将硬核软化，即采用硬件描述语言对硬核的功能进行行为级描述。通过使用该行为级描述，在保护黑匣子具体内容的同时，用户仍可进行 IP 的仿真。

从本节的介绍，不难看出 FPGA/CPLD 在 EDA 设计中具有重要作用。概括起来有以下几点：

1．VHDL 程序通过 FPGA/CPLD 可以得到 EDA 设计的最终产品，在产品数量较小时快速占领市场。

2．VHDL 程序通过 FPGA/CPLD 验证可以形成固核，具有一定的商品价值。

3．VHDL 程序的 FPGA/CPLD 验证实验是一种 ASIC 设计的硬件仿真手段。

习　　题

1．Top-Down 设计方法主要包括哪几个阶段？分别包含什么内容？

2．什么是 IP 核？学习 VHDL 与掌握 IP 核技术的关系是什么？

3．软核有什么价值？

4．固核有什么价值？为什么要进行固核硬化？

5．为什么说硬核的价值最高？为什么要进行硬核软化？

6．HDL 指的是什么？它有什么特点？常见的 HDL 有哪些？

7．VHDL 指的是什么？它有什么优点？它最突出的优点是什么？

8．学习 VHDL 的目的是什么？VHDL 有哪些主要应用？

9．FPGA/CPLD 的主要优缺点是什么？ASIC 的主要优缺点是什么？

10．FPGA/CPLD 在 EDA 设计中有何特殊作用？

第二章　VHDL 程序的结构及其描述方式

本章读导

通过本章的学习，应当对 VHDL 程序的宏观结构有一个明确的概念性认识，理解实体、构造体、配置、程序包、库的基本功能，并能够尝试编写简单的 VHDL 程序。同时，理解 VHDL 的三种描述方式：行为级描述、RTL 级描述、结构级描述。

2.1　VHDL 程序的结构

如前所述，硬件描述语言是高层次、自动化设计的起点和基础。要掌握 EDA 技术必须从学习一种硬件描述语言开始。从整体上理解 VHDL 程序架构，对于迅速掌握 VHDL 是十分有利的。

为了使读者对 VHDL 程序有全面的了解，我们对 VHDL 程序进行了解剖，将一个完整的 VHDL 程序分为实体（Entity）、构造体（Architecture）、配置（Configuration）、程序包（Package）、库（Library）五个组成部分，如图 2.1 所示。其中实体、构造体和配置是需要设计工程师编写的程序段；而程序包和库则是在编程中要调用的程序段。从设计共享的角度来看，包集合、库以及配置均是设计共享的手段。我们可以将所编写的程序单元用包集合、库或配置的形式存起来，成为共享或复用的部分。

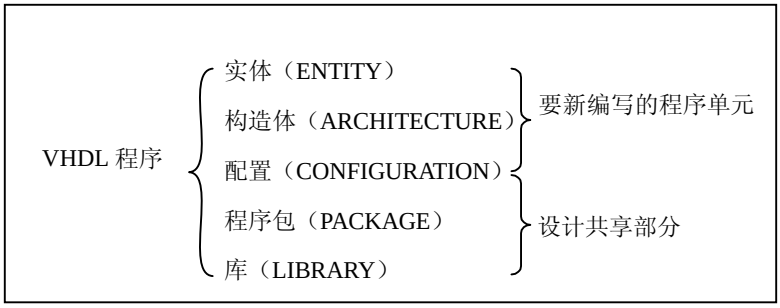


图 2.1　VHDL 程序组成部分及其功能

若我们把当前正在设计的电路模型称之为设计单元，已有的可调用基本逻辑门和电路模型称之为元件，那么这五个组成部分的主要作用如下：

- 实体：描述设计单元的外围接口信号和内部参数。
- 构造体：描述设计单元的内部结构和逻辑行为。
- 配置：为设计单元从多个构造体中选择合适的构造体或从库中选取合适的元件，以便于进行设计单元的仿真或综合。
- 程序包：存放各设计模块都能共享的数据类型、常数和子程序等。
- 库：存放已经编译了的元件和程序包，以便在设计单元中使用。库可由系统工程师自行设计或由 ASIC 芯片制造商提供。