高等学校电子与通信工程类专业“十二五”规划教材

EDA技术与VHDL设计

黄沛昱　刘科征　谭钦红　雷　芳　编著

西安电子科技大学出版社
http://www.xduph.com

014009302

TN702.2
05

高等学校电子与通信工程类专业“十二五”规划教材

EDA 技术与 VHDL 设计

黄沛昱　刘科征　谭钦红　雷芳　编著

北京航空航天大学
藏书
图书馆

西安电子科技大学出版社

TN702.2
05

北航　C1695453

内 容 简 介

本书紧跟 EDA 技术发展趋势，以工程实践应用为出发点，以流行 EDA 集成工具 Quartus II 和仿真工具 Modelsim 为设计平台，以实例的形式深入浅出地讲解了 EDA 技术、VHDL 语言以及数字电子系统的设计。

全书共 9 章，分为“EDA 技术概述”、“基础电路设计”、“系统电路设计”三个层次。“EDA 技术概述”层次作为 EDA 技术的入门，首先讲解了与 EDA 技术相关的软硬件知识(第 1 章)，然后比较全面地介绍了可编程逻辑器件，包括硬件结构、特点、编程和配置电路、主流型号等(第 2 章)；“基础电路设计”层次以 VHDL 语言和经典电路模块为主，首先以实例引入 VHDL 语言的基本语法结构(第 3 章)，然后系统地讲解了 VHDL 语言的要素(第 4 章)、VHDL 语言的各种基本语句(第 5 章)以及状态机的设计方法(第 6 章)；“系统电路设计”层次以代码的重用、层次型设计和数字电子系统的设计为主，介绍了程序包和子程序的使用(第 7 章)、仿真测试平台(第 8 章)以及数字电子系统的设计及实例(第 9 章)。

本书可供高等院校电子工程、通信工程、自动化、计算机等相关专业本科生或研究生使用，也可作为相关工程技术人员的参考用书。

图书在版编目(CIP)数据

EDA 技术与 VHDL 设计/黄沛昱等编著.
—西安：西安电子科技大学出版社，2013.8
高等学校电子与通信工程类专业“十二五”规划教材
ISBN 978-7-5606-3132-5

Ⅰ. ① E… Ⅱ. ① 黄… Ⅲ. ① 电子电路—电路设计—计算机辅助设计—高等学校—教材
② VHDL 语言—程序设计—高等学校—教材 Ⅳ. ① TN702 ② TP312

中国版本图书馆 CIP 数据核字(2013)第 167040 号

策　　划　邵汉平
责任编辑　南　景　邵汉平
出版发行　西安电子科技大学出版社(西安市太白南路 2 号)
电　　话　(029)88242885　88201467　邮　　编　710071
网　　址　www.xduph.com　电子邮箱　xdupfxb001@163.com
经　　销　新华书店
印刷单位　陕西华沐印刷科技有限责任公司
版　　次　2013 年 8 月第 1 版　2013 年 8 月第 1 次印刷
开　　本　787 毫米×1092 毫米　1/16　印张 21
字　　数　497 千字
印　　数　1～3000 册
定　　价　36.00 元
ISBN 978 – 7 – 5606 – 3132 – 5/TN
XDUP 3424001-1

* * * 如有印装问题可调换 * * *

前　言

自20世纪90年代以来，EDA技术(即电子设计自动化技术)的发展非常迅速，在电子、通信、工业自动化、智能仪表、图像处理及计算机等领域得到了广泛应用，已成为电子工程师必须掌握的重要技能之一。随着EDA技术的不断发展，社会对EDA技术人才的要求不断提高，EDA技术的教学也在不断改革。

本书紧跟EDA技术的最新发展趋势，尽可能引入流行的FPGA/CPLD器件，采用最新版本的EDA软件工具(Quartus Ⅱ 9.1和Quartus Ⅱ 11.1版本)以及新颖的实例，以工程实践应用为出发点，旨在培养符合社会需求的具有实践动手能力和创新能力的EDA技术人才。本书主要特点如下：

(1) 知识点完整、系统。本书基于EDA技术和VHDL语言构建了一个有机、完整的知识体系，教学内容按照“EDA技术概述”、“基础电路设计”、“系统电路设计”三个层次划分，各章节内容安排合理、循序渐进。

(2) 以实例引入语法，打破传统计算机类语言学习的模式，使读者能够快速入门，有效地增强其学习兴趣。本书第3章首先通过简单的组合逻辑电路和时序逻辑电路的实例让读者感性认识VHDL语言，使其能够仅通过这一章的学习就掌握基本的VHDL描述方法，极大地降低了学习的难度。其他章节同样采用实例的形式进行知识点的讲解，读者在学习理论知识后能够立即实现验证，避免了单纯讲解语法带来的枯燥无味，有利于提高其学习的积极性，提升学习的效率。

(3) 精选教学内容和实例。实例的选取既突出经典性，又注重新颖性。本书既采用多种不同的方式设计数据选择器、计数器、移位寄存器等经典模块，又引入了与通信等专业相关的设计项目。

全书共9章，其中第1、2章属于“EDA技术概述”层次，第3～6章属于“基础电路设计”层次，第7～9章属于“系统电路设计”层次。

“EDA技术概述”层次对EDA技术涉及的相关软硬件知识作了介绍。第1章简要介绍了EDA技术的发展、设计流程、设计方法、硬件描述语言、常用工具，目的是使读者了解技术的由来、发展的前景以及学习的重点。第2章讲解了目前常用的可编程逻辑器件的分类、结构特点、下载电缆、编程与配置电路、下载文件类型以及主流产品型号等，帮助读者了解实验过程或实际系统中与硬件相关的内容。编者多年的教学实践表明，很多同学在学习完该门课程后仍然对下载电缆类型、下载模式、下载文件类型等不能区分和理解。编者认为该部分内容是使用可编程逻辑器件的基础，有必要花费一定时间进行讲述。

“基础电路设计”层次针对VHDL语言的基本语法和经典电路模块作了介绍。第3章以几个简单的实例引出VHDL语言的基本结构以及一些基础的语法知识，有利于读者快速入门。第4章系统地讲解了VHDL语言的要素，包括文字规则、数据对象、数据类型、操作符等。第5章将VHDL基本语句分为并行语句和顺序语句两类，分别进行介绍，并给出

了多个设计实例。学习 VHDL 语句时，特别要注意语言的硬件特性，即硬件行为的并行性决定了 VHDL 语言的并行性。第 6 章介绍了有限状态机的设计，它是一种重要的设计方法。

“系统电路设计”层次以代码的重用、层次型设计以及电子系统的设计为主进行介绍。第 7 章讲述了程序包和子程序的使用。子程序通常在程序包中定义，放置于库中，以供其他设计实体直接调用。它引入了新的设计层次，体现了代码分享和重用的意义。第 8 章介绍了 VHDL 语言的另一重要功能，即仿真功能，采用实例的形式介绍了几种常用的仿真测试平台以及断言语句的使用。第 9 章介绍了数字电子系统的设计流程，并通过三个实例进一步讲解具体的设计方法。

本书内容编排完整、系统，保证了 EDA 技术和 VHDL 语言知识体系的完整性。各章节之间既存在联系，又具有相对的独立性。本科学校教师可根据课时设置情况、专业特点等选择其中的某些章节讲授，在课时不足的情况下，可只讲述前 6 章，完成“EDA 技术概述”和“基础电路设计”层次，确保学生能够掌握 VHDL 语言、EDA 软件工具及一般电路的设计。如课时充足或针对研究生授课，则可以增加“系统电路设计”层次的讲述，构建更加系统、完整的教学内容和教学层次。当然，为了配合有限的学时数，将教学内容剪裁只是权宜之计。我们应鼓励学生利用课余时间多学习、多实践，自学完成本书其余内容，以更好地提升自己的实践动手能力，拓宽知识面，加深理解。

本书由黄沛昱统稿并担任主编，刘科征、谭钦红、雷芳参与了编写。

由于 EDA 技术是一门发展迅速的新技术，因此对 EDA 技术的研究还有待深入，相应的教学内容、方法和手段也有待进一步改进、完善。限于作者水平，书中难免存在疏漏、不妥之处，敬请读者批评指正。作者邮箱：huangpyu@cqupt.edu.cn。

编　者

2013 年 4 月

于重庆邮电大学

此为试读，需要完整PDF请访问：www.ertongbook.com

目　录

第 1 章　EDA 技术概述

EDA 技术是现代电子设计技术之一，已广泛应用于电子、通信、工业自动化、智能仪表、图像处理以及计算机等领域。本章首先介绍 EDA 技术的广义定义和狭义定义。就狭义定义而言，EDA 技术主要包括硬件载体(大规模可编程逻辑器件)、设计表达形式(硬件描述语言)、开发环境(EDA 工具)等。接下来对 EDA 工程设计流程、设计方法、硬件描述语言、EDA 工具和 IP 核进行介绍，目的是使读者对 EDA 技术有一个全面、基础的了解。

1.1　EDA 技术及其发展

1.1.1　EDA 技术的概念

电子技术是 19 世纪末发展起来的新兴技术，电子技术的发展与电子器件的发展息息相关。从 1904 年弗莱明发明第一只真空二极管，1906 年德福雷斯特发明真空三极管，到 1950 年 PN 结型晶体管的出现，开辟了电子器件的新纪元，引起了一场电子技术的革命。随着电子产品的日趋复杂，单个电子器件中需要的晶体管越来越多，对于上百万个晶体管，如何确保其可靠性并缩小体积、减轻重量等成为电子产品发展中迫切需要进行的突破。这一突破的结果是集成电路的出现。1958 年，杰克·基尔比制成了第一块基于硅的集成电路板。集成电路在一小块半导体晶片上，将电路所需的成千上万的晶体管、二极管、电阻、电容及布线互连在一起。集成电路的出现使得电子器件向微小型化、高可靠性方面迈进了一大步。随着集成度的不断提高，大规模集成电路(LSI，Large Scale Integrated circuits)、超大规模集成电路(VLSI，Very Large Scale Integrated circuits)、特大规模集成电路(ULSI，Ultra Large Scale Integrated circuits)，以及巨大规模集成电路(GSI，Giga Scale Integration circuits)相继出现，集成度平均每两年提高近 3 倍。进入 21 世纪，电子技术发展的根基就是微电子技术的进步，它表现在大规模集成电路加工技术(即半导体工艺技术)的发展上。目前，表征半导体工艺水平的线宽已经达到 22 nm。微电子技术和现代电子设计技术是相互促进、相互推动又相互制约的两个技术环节。微电子技术的进步意味着传统电子设计技术的不适应，要求现代先进的电子理论、电子技术、仿真技术、设计工艺等现代电子设计技术必须满足微电子技术的进步需求。电子设计自动化(EDA，Electronic Design Automation)技术就是在电子技术快速发展的过程中产生的现代电子设计技术。

由于 EDA 技术发展迅速、内容丰富、涵盖范围广，目前对其并无统一的定义。在此，作者认为 EDA 技术的定义可分为广义和狭义两种。广义的 EDA 技术，是指以计算机为工作平台，融合了电子技术、计算机技术、信息处理技术等各种先进技术，可进行电子产品

自动设计的技术。从该定义出发，电子电路设计、PCB(Printed Circuit Board，印制电路板)设计、IC(Integrated Circuit，集成电路)设计等均属于 EDA 技术范畴。狭义的 EDA 技术，仅指以大规模可编程逻辑器件为硬件载体，以硬件描述语言(HDL，Hardware Description Language)为系统逻辑描述的表达形式，以相关 EDA 软件工具为开发环境，自动完成逻辑编译、逻辑化简、逻辑综合及优化、布局布线、仿真测试等多项功能，以及对特定目标芯片的适配编译、逻辑映射、编程下载等工作，直至最终实现特定的电子系统功能。本书讨论的所有对象仅指狭义 EDA 技术。总的来说，狭义 EDA 技术的定义包含以下几个主要内容：

(1) 大规模可编程逻辑器件(PLD，Programmable Logic Devices)，即一种可由用户定义其具体实现逻辑功能的集成器件。目前的主流产品有现场可编程门阵列(FPGA，Field Programmable Gate Array)和复杂可编程逻辑器件(CPLD，Complex Programmable Logic Devices)两类，具体将在第 2 章中详细介绍。

(2) 硬件描述语言，即实现系统逻辑功能的具体表述形式，它用软件编程的方式来描述电子系统的逻辑功能、电路结构和连接形式。目前常用的硬件描述语言有 VHDL 和 Verilog。本书从第 3 章开始将详细讲述 VHDL 语言的语法结构以及应用。

(3) 相关 EDA 软件开发工具。EDA 工具既有与 EDA 整个设计流程中某一个技术环节相对应的专用 EDA 工具(如著名的逻辑综合器 Synplify 和仿真器 Modelsim)，也有 PLD 生产厂商为方便用户所提供的集成开发环境(如 Altera 公司的 Quartus II 和 Xilinx 公司的 ISE-Web PACK Series)。本书的配套实验教材《EDA 技术与 VHDL 设计实验指导》中将介绍 Quartus II 和 Modelsim 的使用方法。当然，要实现一个完整的电子系统，还需要有相关外围电路，这里不做进一步的说明。

1.1.2　EDA 技术的发展

EDA 技术的发展可分为计算机辅助设计(CAD，Computer-Aided Design)、计算机辅助工程设计(CAE，Computer-Aided Engineering design)以及电子设计自动化(EDA)三个阶段。

20 世纪 70 年代，是 EDA 技术发展的初期，设计者开始使用计算机辅助进行 IC 版图的编辑、PCB 布局布线等这些在产品设计过程中重复性很高的繁杂劳动，最具有代表性的产品是美国 ACCEL 公司开发的 Tango 布线软件。但由于当时软件工具受到计算机工作平台的制约，其支持的设计工作有限且性能也比较差。

20 世纪 80 年代，伴随着计算机和集成电路的发展，EDA 技术的发展进入到计算机辅助工程设计(CAE)阶段。这一阶段的 EDA 工具，除了具有图形绘制功能外，还增加了逻辑模拟、定时分析、故障仿真、自动布局布线等功能，主要目的是解决电路设计完成前的功能检测等问题。利用这些新增的功能，设计者能够在产品制作完成前就预知产品的功能与性能，能生成制造产品的相关文件，使设计阶段对产品性能的分析前进了一大步。这一时期的 EDA 工具已经能够代替设计者的部分工作。

20 世纪 90 年代，随着可编程逻辑器件的发展，设计者可以选择不同规模的 PLD 器件，通过对器件功能的设计，实现电子系统功能。这个阶段发展起来的 EDA 工具能够完成设计者从事的许多高层次的设计工作，如将用户需求转换为设计技术规范，有效地处理可用的

设计资源与理想的设计目标之间的矛盾，按具体的硬件、软件和算法分解设计等。另一方面，硬件描述语言(HDL)的出现是这个阶段最重要的成果之一，它使得 EDA 的设计进入到抽象描述的设计层次，设计者可以在不熟悉具体电路结构的情况下，完成电子系统的设计。

各 EDA 设计公司都致力于推出兼容各种硬件实现方案、支持标准硬件描述语言以及含有各种工艺标准元件库的 EDA 工具，有效地将 EDA 技术推向了成熟。由于电子技术和 EDA 工具的发展，设计者可以使用 EDA 工具在较短的时间内通过一些简单标准化的设计过程，利用厂商提供的设计库来完成系统的设计与验证。

1.2　EDA 设计流程

1.2.1　FPGA/CPLD 工程设计流程

大规模可编程逻辑器件(PLD)是 EDA 设计的硬件载体。PLD 种类繁多，目前的主流器件是 CPLD 和 FPGA。PLD 器件的出现，其影响丝毫不亚于 70 年代单片机的发明和使用。PLD 能够完成任何数字器件的功能，在速度、芯片容量和数字逻辑方面均优于单片机。FPGA/CPLD 工程设计流程如图 1-1 所示。

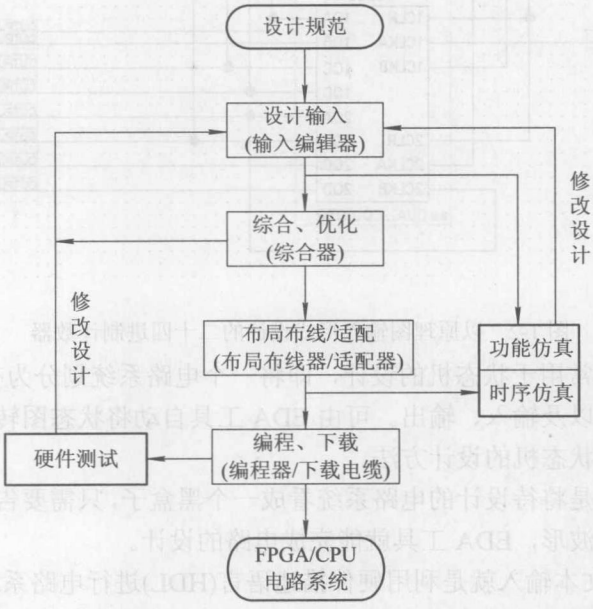

图 1-1　FPGA/CPLD 工程设计流程

1. 设计规范

设计者首先需要对产品的应用场合、功能、要求等进行考虑和分析，确定一些技术指标，如速度、面积、功耗等。

2. 设计输入

设计输入即用一定的逻辑表达方式将电路系统的设计表达出来。常用的表达方式有图

此为试读，需要完整PDF请访问：www.ertongbook.com

形输入和文本输入，对应的 EDA 工具为图形编辑器和文本编辑器。

(1) 图形输入。图形输入形式通常包括原理图输入、状态图输入和波形图输入。

原理图输入形式是最常用的图形输入形式，类似于传统电子设计方法中电路原理图的绘制。原理图由逻辑器件和连线构成，其中逻辑器件既可以是 EDA 软件库中预定义的功能模块(如与门、或门、非门、触发器、74 系列器件、加法器、乘法器等)，也可以是自定义的功能模块。图 1-2 是在 Quartus II 软件中绘制的电路原理图，其功能是采用 74390 和与门完成二十四进制计数器。原理图输入形式的优点在于简单、直观，不需要学习 HDL，容易被初学者接受。但它也具有十分明显的缺点：① 设计规模一旦增大，设计的易读性将迅速下降，面对复杂的电路连线，要搞清电路功能非常困难；② 如果出现错误，查找错误和修改错误都十分困难；③ 更改电路功能和结构比较困难；④ 功能模块的不兼容导致设计的可移植性较差。

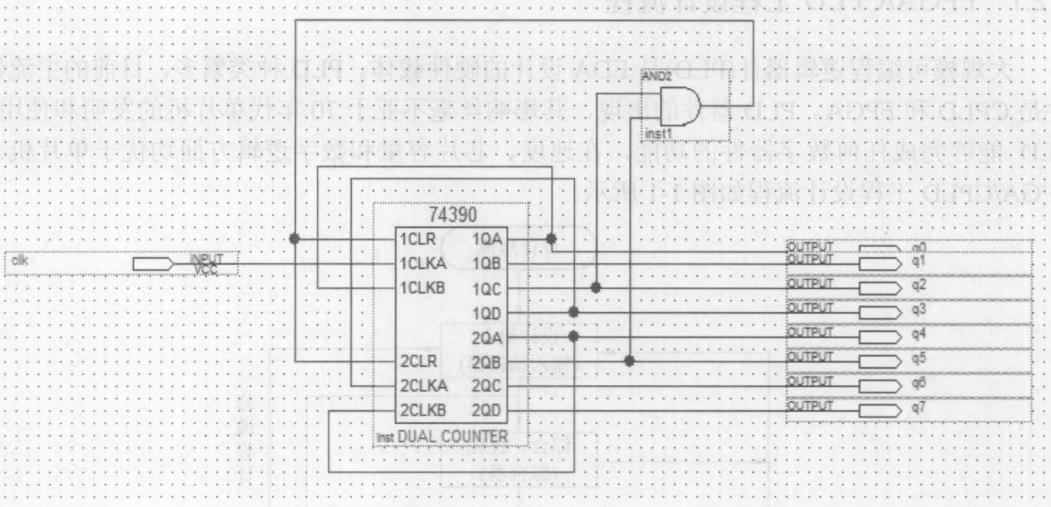

图 1-2　以原理图输入形式设计的二十四进制计数器

状态图输入形式常用于状态机的设计，即将一个电路系统划分为有限个状态，确定不同状态间的转移条件以及输入、输出。可由 EDA 工具自动将状态图转化为 HDL 代码。本书第 6 章将专门介绍状态机的设计方法。

波形图输入形式是将待设计的电路系统看成一个黑盒子，只需要告诉 EDA 工具电路系统输入、输出的时序波形，EDA 工具就能完成电路的设计。

(2) 文本输入。文本输入就是利用硬件描述语言(HDL)进行电路系统的设计，常用的硬件描述语言主要有 VHDL 和 Verilog HDL 两种。这种方式和传统的计算机编程输入方式类似。应用 HDL 的文本输入形式克服了原理图输入形式的弊端。

在设计中，可以将原理图和 HDL 设计结合起来，实现高效、稳定、符合要求的设计。

3. 综合、优化

综合是将利用 HDL、原理图等实现的软件设计转化为基本逻辑门、触发器、存储器等基本逻辑单元的连接关系，即门级电路甚至更底层的电路结构描述文件的过程。如何将软件设计转化为硬件电路，就需要利用 EDA 工具中的综合器进行“翻译”。综合器类似于软

件程序的编译器，但较编译器有更高级的功能。编译器也能够将高级语言翻译成基于某种特定 CPU 的机器代码，但这种代码仅限于这种 CPU 而不能移植，并且不能代表硬件结构。另一方面，编译器的工作只是机械、单纯地将高级语言“一一对应”地翻译为机器代码；综合器则能够根据预先设置的各类约束条件(如时间约束、面积约束等，以及设计库和工艺库)，能动地选择最优的方式将软件设计翻译为底层电路结构。这就是说，对于相同的设计表述，综合器可以综合出不同的电路结构，有的面积小，但速度慢；有的速度快，但面积大。选择电路的实现方案正是综合器的任务，综合器能够尽最大努力选择一种满足各项约束条件且成本最低的实现方案，而且综合后产生的电路结构(被称为网表文件)不依赖于任何硬件环境，能够被移植到任何通用的硬件环境中。网表文件有多种格式，如 EDIF、VHDL、VQM、Verilog 等。

总的来说，整个综合过程就是将设计者在 EDA 工具中输入的 HDL 文本设计、原理图设计或状态图设计等，依据给定的硬件结构组件和约束条件进行编译、优化、转化和综合，最终获得门级电路甚至更底层的电路结构描述的网表文件。综合器既可以使用第三方 EDA 公司提供的专用综合器(如 Synplicity 公司提供的 Synplify 综合器)，也可以使用 FPGA/CPLD 供应商提供的综合器(如 Altera 公司集成 EDA 软件工具 Quartus II 中自带的 Analysis &Synthesis 模块)。

4. 布局布线/适配

通过综合后产生的电路结构网表文件，还需要与指定的目标器件进行逻辑映射，即将工程的逻辑和时序要求与目标器件的可用资源相匹配。布局布线/适配用来将每个逻辑功能分配给最合适的逻辑单元位置，进行布线和时序，并选择相应的互连路径和引脚分配，产生最终的下载文件。下载文件有多种格式，如 .sof、.pof、.hex、.jam 等，具体将在第 2 章中讲述。因为需要与具体目标器件的硬件结构细节相对应，布局布线器/适配器一般由 FPGA/CPLD 供应商提供。

5. 仿真

在硬件验证前，最好使用 EDA 工具对设计进行模拟验证，即仿真。通过仿真，可以检查设计文件是否和预期结果一致，可以在设计的早期就排除错误，缩短设计周期和成本。仿真通过仿真器完成，既可以采用第三方 EDA 公司提供的专用仿真工具(如 Mentor Graphic 公司的 ModelSim 仿真器)，也可以采用 FPGA/CPLD 供应商提供的 EDA 工具直接完成。

仿真分功能仿真和时序仿真两种。功能仿真仅对逻辑功能进行模拟测试，以验证逻辑功能是否正确，是否满足原设计要求。仿真过程不涉及任何具体器件的硬件特性，不考虑延时。不需要经过综合和布局布线/适配阶段就能够进行功能仿真。时序仿真是接近真实器件运行特性的仿真，包含了硬件延时信息。时序仿真的文件来自于综合、布局布线/适配后产生的文件。

6. 编程、下载

布局布线/适配阶段产生的最终下载文件，可以通过编程器或者下载电缆下载到 FPGA/CPLD 中，以便进行硬件测试和调试。由于 FPGA 和 CPLD 结构上的差别，导致二者在使用上有一些区别。一般来说，对 FPGA 进行最终文件的下载称为配置(Configure)；对 CPLD 进行下载操作称为编程(Programmable)。具体关于配置和编程的概念，以及下载电

缆的类型、适用范围等都将在第 2 章中讲述。

7. 硬件测试

把最终下载文件下载到 FPGA/CPLD 后，就可以对设计进行硬件测试，以便最终验证设计在目标器件上实际工作的情况，有助于在完成最终电路系统前排除错误、改进设计。

1.2.2 ASIC 工程设计方法及流程

在 EDA 技术领域，除了 CPLD 和 FPGA 这两个常用的硬件载体外，还有一个使用频繁的概念，那就是 ASIC(Application Specific Integrated Circuits，专用集成电路)。专用集成电路是相对于通用集成电路而言的，是指应特定用户要求和特定电子系统的需要而设计、制造的集成电路。就设计方法而言，设计 ASIC 的方法可分为全定制法、半定制法和可编程逻辑器件法。

1. 全定制法

全定制法是利用集成电路最基本的设计方法，基于晶体管级的、对所有元器件都进行精工细作的设计，且采用手工设计版图的制造方法。全定制设计需要考虑工艺条件，根据电路的复杂程度决定器件的工艺类型、布线层数、材料参数、工艺方法、极限参数、成品率等，一般由专业微电子集成电路设计人员完成。全定制法的优点是：面积利用率高、性能较好、功耗较低；有利于提高芯片的集成度和工作速度，以及降低功耗。但由于其设计周期长、设计成本昂贵，且功能模块和单元库越来越成熟，全定制法逐渐被半定制法所替代。

2. 半定制法

半定制法又分为门阵列法和标准单元法。

门阵列包括规则的、未连接的行和列的晶体管结构，器件的连接完全是由设计所决定的。一旦完成设计，布线软件就能算出哪些晶体管要进行连接。采用门阵列法，软件从低层次功能模块的连接开始直至完成整个器件连接的设计。门阵列法适用于开发周期短、低成本的小批量数字电路设计。

标准单元法采用预先设计好的称为标准单元的逻辑单元，如 D 触发器、加法器、计数器等。所有标准单元均采用定制方法预先设计，设计者只需要确定标准单元的布局以及连线。但当工艺更新后，标准单元库也要随之更新，这是一项十分繁重的工作。为了解决人工设计单元库费时费力的问题，目前市场上销售的 IC CAD 系统几乎都含有标准单元自动设计工具。此外，设计重用(Design Reuse)技术也可用于解决单元库的更新问题。

3. 可编程逻辑器件法

可编程逻辑器件法是利用可编程逻辑器件设计专用集成电路的方法。用户可以借助 EDA 软件和开发系统在实验室内自行设计、测试、验证。可编程逻辑器件法能够缩短设计周期，提高设计效率，但采用此种方法设计的 ASIC 器件在性能、速度以及单位成本上相对于全定制法和半定制法设计的 ASIC 器件不具竞争性。可编程逻辑器件法特别适用于从事电子系统设计的工程师利用 EDA 工具进行 ASIC 设计。

目前，为降低单位成本，在利用可编程逻辑器件实现设计后，可以采用特殊的方法将设计转化为 ASIC 电路，如 Altera 公司的 FPGA 器件 Stratix 系列在设计成功后可以通过

HardCopy 技术转化成对应的门阵列 ASIC 产品。

ASIC 设计的流程如图 1-3 所示。第一步是进行项目分析，包括市场需求分析、可行性研究、论证与决策、形成任务书等。第二步是系统设计，该阶段需要确定采用的设计方式，如全定制、半定制或可编程逻辑器件转化，还需要确定系统的功能、性能等；同时进行模块的划分，即将系统分割成各个功能子模块，给出各模块之间的信号连接关系。第三步是模块设计，即确定各个模块的电路实现，可采用硬件描述语言或原理图等形式进行具体逻辑的描述，设计完成后可进行功能验证，检查功能是否与预期一致。第四步是验证，即采用综合器对设计进行综合，以获得具体的电路网表文件，再次对网表文件进行仿真验证。第五步是版图设计，即将逻辑设计中的每一个逻辑元件、电阻、电容等以及它们之间的连线转化成集成电路制造所需要的版图信息，可采用手工或自动的方式进行版图规划、布局布线。由于不同的布局布线会产生不同的电路连线延时，所以布局布线可能会反复多次，以选择最佳方案。第六步是版图验证，包括引入电路连线延时后进行的仿真，以及设计规则检查、电气规则检查等。第七步是制版、流片，即生成光刻掩膜板，加工厂家进行试验性生产。第八步是芯片测试，以确保是否满足设计要求，并评估成品率。最后一步是投入量产，完成 ASIC 设计。

图 1-3　ASIC 设计的一般流程

1.3　EDA 设计方法

传统数字电路系统的设计方法是自底向上(Bottom-Up)，设计者首先需要决定所使用的目标器件的类别和规格，如 74 系列器件、某种 RAM 或 ROM、某类单片机或某些专用功能芯片等；然后构成多个功能模块，如数据采集控制模块、信号处理模块、数据交换模块和接口模块等，直至最后构成完整的系统。自底向上的设计方法必须始终考虑实现系统的目标器件的功能、技术参数等细节，如果在设计过程中的任一阶段，最底层的目标器件由于不满足技术参数、市场缺货或是设计要求降低成本等各种因素而导致不得不更换的情况出现，都会使得设计工作前功尽弃。自底向上的设计方法不适用于大规模数字系统的设计，因其效率较低，且设计周期较长。

随着 EDA 技术的快速发展，自顶向下(Top-Down)的设计方法得到了有效利用，成为

CPLD、FPGA 以及 ASIC 设计的主要手段。自顶向下的设计方法的本质是层次建模，分模块设计。设计者首先规划整个系统的功能和性能，然后对系统功能进行划分，将系统分割为功能较简单、规模较小的若干个功能子块，并确立它们之间的相互关系；然后进一步对各个子模块进行分解，直到达到无法进一步分解的底层功能模块。在进行模块划分时，需要注意模块功能的完整性和可重复利用性。

采用自顶向下的设计方法，由于整个设计是从顶层系统开始的，可以从一开始就掌握系统的性能状况。功能划分时，较高层次的设计描述比较抽象，与具体的硬件实现无关，当然也不用考虑硬件实现中的技术细节，可以对其进行功能仿真，在设计的早期阶段就验证设计方案的正确性。一旦高层次的逻辑功能满足要求，就可以在较低层次针对具体的目标器件进行具体描述。另一方面，系统被分解为若干功能子块后，可以对每个独立的模块指派不同的设计人员，他们可以在不同的地点工作，最后再将不同的模块集成为最终的系统。自顶向下的设计方法缩短了设计周期；设计规模越大，优势越明显。总结说来，EDA 设计方法与传统设计方法的区别如表 1-1 所示。

表 1-1　传统设计方法与 EDA 设计方法的比较

比较内容	传统设计方法	EDA 设计方法
设计思路	自底向上	自顶向下
硬件载体	通用元器件	可编程逻辑器件
仿真、调试阶段	系统硬件设计后期	系统设计早期开始，层层验证
设计文件	电路原理图	多种设计形式，包括原理图、HDL、状态图等
实现方式	手工实现	软件自动实现

1.4　硬件描述语言

1.4.1　硬件描述语言的出现和意义

长期以来，我们比较熟悉的是诸如 C、C++一类的计算机程序设计语言，这类语言在本质上都是顺序执行的。同样，在硬件设计领域，设计人员也希望使用一种标准的语言来进行硬件的设计。在这种情况下，硬件描述语言应运而生。硬件描述语言(HDL)的出现是 EDA 技术发展中的一个重要成果，它是一种用形式化的方法描述数字电路和系统的语言。通过这种语言，设计者能够描述自己的设计思想，表达复杂的电路系统。硬件描述语言发展至今已有 30 多年的历史，已成功运用于电路设计的各个阶段，包括建模、仿真、综合等。目前使用的 HDL 主要有 VHDL、Verilog HDL、AHDL、ABEL、SystemC、Superlog 等，其中 VHDL 和 Verilog HDL 是最常用的两种 HDL，均是 IEEE (the Institute of Electrical and Electronics Engineers)标准 HDL。

使用 HDL 进行设计，设计者可以在抽象的层次上对电路进行描述，而不必关心特定器件的选取、制造工艺等，逻辑综合工具能够将设计自动转化为任意一种制造工艺版图。此外，通过使用 HDL，设计者还可以在设计周期的早期就对电路的功能进行验证，大大降低

了在设计后期的门级网表或物理版图上出现错误的可能性，避免了设计工作的反复，显著地缩短了设计周期。最后，通过简洁明确的代码来描述复杂的逻辑控制，特别是使用 IEEE 标准所规范的 HDL，使得设计便于修改，具有较好的移植能力。

目前，随着数字电路复杂性的不断增加以及 EDA 工具功能的强大，HDL 已成为硬件设计师必须掌握的语言。

1.4.2　VHDL 和 Verilog HDL

VHDL 的英文全称是 Very High Speed Integrated Circuit Hardware Description Language，即超高速集成电路硬件描述语言。VHDL 诞生于 1982 年，最初是由美国国防部开发供美军提高设计的可靠性和缩减开发周期的一种设计语言，在 1987 年底被 IEEE 确定为标准硬件描述语言，作为“IEEE 标准 1076-1987”(简称 87 版本)。1993 年，IEEE 对 VHDL 进行了修订，公布了新版本的 VHDL，即“IEEE 标准 1076-1993”(简称 93 版本)。93 版本较 87 版本从更高的抽象层次和系统描述能力上扩展了 VHDL 的内容，增加了一些新的命令和属性。目前公布的最新 VHDL 标准版本是 1076-2002 版。自从 VHDL 成为业界标准之后，在设计领域得到了广泛的认可，得到众多 EDA 公司的支持。

Verilog HDL 是在 C 语言的基础上发展起来的，1983 年由 GDA(Gateway Design Automation)公司提出。1989 年，GDA 公司被世界上最大的 EDA 公司——Candence 公司收购，Verilog HDL 成为 Cadence 公司的私有财产。该公司大力发展 Verilog HDL，于 1990 年成立了 OVI(Open Verilog International)组织，负责促进 Verilog HDL 的发展。基于 Verilog HDL 的优越性，IEEE 于 1995 年制定了 Verilog HDL 的 IEEE 标准，即 Verilog HDL 1364-1995 标准；2001 年对其加以修订，又发布了 Verilog HDL 1364-2001 标准，后来还加入了 Verilog HDL-A 标准，使 Verilog HDL 有了模拟设计描述的能力。

不管是 VHDL 还是 Verilog HDL，都允许在同一个电路模型内进行不同层次的描述，具有强大的系统硬件描述能力。但二者各有其优缺点。VHDL 语法较严谨，能够通过 EDA 工具的自动语法检查排除很多设计中的错误；此外，VHDL 的行为级描述能力也强于 Verilog HDL，它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。但 VHDL 对数据类型匹配严格，代码较 Verilog HDL 冗长，且 VHDL 不具有晶体管开关级的描述能力和模拟设计的描述能力。Verilog HDL 语法较自由，初学者容易上手，但自由的语法也容易造成更多的错误。编程语言接口 PLI 是 Verilog HDL 最重要的特性之一，它使得设计者可以通过自己编写的 C 程序代码来访问 Verilog HDL 内部的数据结构。设计者还可以使用 PLI 按照自己的需要来配置 Verilog HDL 仿真器。

当然，除了上述两种最常用的 HDL 外，还有 ABEL(Advanced Boolean Equation Language)、AHDL(Altera Hardware Description Language)等。ABEL 是一种早期的硬件描述语言，由美国 Data I/O 公司推出，是设计 PAL/GAL 器件的主要工具。AHDL 是 Altera 公司推出的硬件描述语言，主要针对其公司生产的 CPLD/FPGA 器件。

1.4.3　硬件描述语言的发展

随着系统级 FPGA 以及系统级芯片的出现，软硬件协调设计和系统设计变得越来越重

此为试读，需要完整PDF请访问：www.ertongbook.com