

## <<EDA技术实用教程>>

### 图书基本信息

书名：<<EDA技术实用教程>>

13位ISBN编号：9787030179739

10位ISBN编号：7030179730

出版时间：2006-9

出版时间：科学出版社

作者：潘松//黄继业

页数：408

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;EDA技术实用教程&gt;&gt;

## 前言

随着基于PLD的EDA技术的发展和应用领域的扩大与深入，EDA技术在电子信息、通信、自动控制及计算机应用等领域的重要性日益突出。

随着技术市场与人才市场对EDA的需求不断提高，产品的市场效率和技术要求也必然会反映到教学和科研领域中来。

以全国大学生电子设计竞赛为例，直到上届赛事中，需要使用EDA技术的赛题超过全部赛题的三分之一，其中有的赛题达到了如果没有EDA技术，将无从下手的程度。

事实上，电子设计竞赛赛题的内容既是市场产品要求和技术进步的一种反映，也是对高校相关教学实验内容改革的要求和促进。

对美国一些高校电子与计算机实验室建设情况的调研结果表明，许多著名院校基于PLD的EDA技术在本科教学中有两个明显的特点：一是各专业中EDA教学实验课程的普及率极高；二是在实验中。

EDA试验成为主流，大部分传统的实验如数字电路、计算机组成、接口、通信、处理器等实验内容，都融入了EDA实验，并更多地注重创新性实验。

这显然是科技发展和市场需求的结果。

为了适应EDA技术的发展和EDA技术教学实验的要求，更加突出实验中EDA技术的实用性，以及面向工程实际的特点和电子设计的自主创新性，本书第三版在实验类型和内容上都增加，每一实验的层次性更加清楚，并注重学生设计能力和自主创新能力的培养，以及与工程实际相结合的动手能力的培养。

另一方面，在实验内容的选择上更突出EDA技术的特点和优势，所以将原来的“实验”改为“实验与设计”。

在章节的安排上也做了一定的调整。

在实验项目上增加和扩充了一些与全国大学生电子设计竞赛典型赛题相关的内容，对于比较典型的设计项目给出了更为详尽的说明。

## <<EDA技术实用教程>>

### 内容概要

《EDA技术实用教程》根据课堂教学和实验操作的要求，以提高实际工程设计能力为目的，深入浅出地对EDA技术、VHDL硬件描述语言、FPGA开发应用及相关知识做了系统和完整的介绍，使读者通过《EDA技术实用教程》的学习并完成推荐的实验，能初步了解和掌握EDA的基本内容及实用技术。

全书包括四部分内容。

第一部分对EDA的基本知识、常用EDA工具的使用方法和目标器件的结构原理做了介绍；第二部分以向导的形式和实例为主的方法介绍了三种不同的设计输入方法；第三部分对VHDL的设计优化做了介绍；第四部分详述了基于EDA技术的典型设计项目。

各章都安排了习题和针对性较强的实验与设计。

书中列举的大部分VHDL设计实例和实验示例实现的EDA工具平台是Quartus II 6.0，硬件平台是Cyclone II FPGA，并在EDA实验系统上通过了硬件测试。

《EDA技术实用教程》可作为高等院校电子工程、通信、工业自动化、计算机应用技术、电子对抗、仪器仪表、数字信号或图像处理等学科的本科生或研究生的电子设计、EDA技术课程和VHDL硬件描述语言的教材及实验指导书，也可作为相关专业技术人员的自学参考书。

## &lt;&lt;EDA技术实用教程&gt;&gt;

## 书籍目录

第1章 概述1.1 EDA技术及其发展1.2 EDA技术实现目标1.3 硬件描述语言VHDL1.4 VHDL综合1.5 基于VHDL的自顶向下设计方法1.6 EDA技术的优势1.7 EDA的发展趋势习题第2章 EDA设计流程及其工具2.1 设计流程2.1.1 设计输入(原理图 / HDL文本编辑)2.1.2 综合2.1.3 适配2.1.4 时序仿真与功能仿真2.1.5 编程下载2.1.6 硬件测试2.2 ASIC及其设计流程2.2.1 ASIC设计方法2.2.2 一般ASIC设计的流程2.3 常用EDA工具2.3.1 设计输入编辑器2.3.2 HDL\_ , 综合器2.3.3 仿真器2.3.4 适配器2.3.5 下载器2.4 Quartus 简介2.5 IP核简介习题第3章 FPGA / CPLD结构与应用3.1 概述3.1.1 可编程逻辑器件的发展历程3.1.2 可编程逻辑器件的分类3.2 简单PLD原理3.2.1 电路符号表示3.2.2 PROM3.2.3 PLA3.2.4 PAL3.2.5 GAL3.3 CPLD结构与工作原理3.4 FPGA结构与工作原理3.4.1 查找表逻辑结构3.4.2 Cyclone / Cyclone 系列器件的结构与原理3.5 硬件测试技术3.5.1 内部逻辑测试3.5.2 JTAG边界扫描测试3.5.3 嵌入式逻辑分析仪3.6 FPGA / CPLD产品概述3.6.1 Lattice公司的CPLD器件系列3.6.2 Xilinx公司的FPGA和CPLD器件系列3.6.3 Altera公司的FPGA和CPLD器件系列3.6.4 Actel公司的FPGA器件3.6.5 Altera公司的FPGA配置方式与配置器件3.7 编程与配置3.7.1 JTAG方式的在系统编程3.7.2 使用PC并行口配置FPGA3.7.3 FPGA专用配置器件3.7.4 使用单片机配置FPGA3.7.5 使用CPLD配置FPGA习题第4章 VHDL设计初步4.1 多路选择器的VHDL描述4.1.1 2选1多路选择器的VHDL描述4.1.2 相关语句结构和语法说明4.2 寄存器描述及其VHDL语言现象4.2.1 D触发器的VHDL描述4.2.2 VHDL描述的语言现象说明4.2.3 实现时序电路的VHDL不同表述4.2.4 异步时序电路设计4.3 1位二进制全加器的VHDL描述4.3.1 半加器描述4.3.2 CASE语句4.3.3 全加器描述和例化语句4.4 计数器设计4.4.1 4位二进制加法计数器设计4.4.2 整数类型4.4.3 计数器设计的其他表达方式4.5 一般加法计数器设计4.5.1 相关语法说明4.5.2 程序分析4.5.3 含并行置位的移位寄存器设计习题第5章 QuartusII应用向导5.1 基本设计流程5.1.1 建立工作库文件夹和编辑设计文件5.1.2 创建工程5.1.3 编译前设置5.1.4 全程编译5.1.5 时序仿真5.1.6 应用RTL电路图观察器5.2 引脚设置和下载5.2.1 引脚锁定5.2.2 配置文件下载5.2.3 AS模式编程配置器件5.2.4 JTAG间接模式编程配置器件5.2.5 USBBlaster编程配置器件使用方法5.3 嵌入式逻辑分析仪使用方法5.4 原理图输入设计方法5.4.1 设计流程5.4.2 应用宏模块的原理图设计习题实验与设计5.1 组合电路的设计5.2 时序电路的设计5.3 设计含异步清零和同步时钟使能的加法计数器5.4 用原理图输入法设计8位全加器5.5 用原理图输入法设计较复杂数字系统第6章 VHDL设计进阶6.1 数据对象6.1.1 常数6.1.2 变量6.1.3 信号6.1.4 进程中的信号与变量赋值6.2 双向和三态电路信号赋值例解6.2.1 三态门设计6.2.2 双向端口设计6.2.3 三态总线电路设计6.3 IF语句概述6.4 进程语句归纳6.4.1 进程语句格式6.4.2 进程结构组成6.4.3 进程要点6.5 并行语句例解6.6 仿真延时6.6.1 固有延时6.6.2 传输延时6.6.3 仿真习题实验与设计6.1 7段数码显示译码器设计6.2 8位数码扫描显示电路设计6.3 数控分频器的设计6.4 32位并进, 并出移位寄存器设计第7章 宏功能模块与IP应用7.1 宏功能模块概述7.1.1 知识产权核的应用7.1.2 使用MegaWizardPlug-InManager7.1.3 在QuartusII中对宏功能模块进行例化7.2 宏模块应用实例7.2.1 工作原理7.2.2 定制初始化数据文件7.2.3 定制LPM\_ROM元件7.2.4 完成顶层设计7.3 在系统存储器数据读写编辑器应用7.4 编辑SignalTapII的触发信号7.5 其他存储器模块的定制与应用7.5.1 RAM定制7.5.2 FIFO定制7.6 流水线乘法累加器的混合输入设计.....第8章 状态机设计第9章 VHDL结构与要素第10章 VHDL基本语句第11章 优化和时序分析第12章 系统仿真第13章 电子系统设计实践附录 EDA实验系统简介主要参考文献

## 章节摘录

对设计者而言有两种情况，一是在高抽象层次进行系统设计并利用综合工具将设计转化为低层次的表示，二是直接在低抽象层次上设计系统，这类似于一个程序员用高级语言编程并用编译器将程序编译成机器代码和直接用机器代码进行编程的情况。

在前一种情况下，设计者可以将精力主要集中于系统级问题上，而不必关心低级结构设计的细节问题。因此将减少设计和编程所花费的时间和精力，并且减少错误的发生。

另一方面，尽管从表面上看，VHDL，等硬件描述语言综合器和软件程序编译器都不过是一种“翻译器”，它们都能将高层次的设计表达转化为低层次的表达，但它们却具有许多本质的区别（如图1.2所示）。

编译器将软件程序翻译成基于某种特定CPU的机器代码，这种代码仅限于这种CPU而不能移植，并且机器代码不代表硬件结构，更不能改变CPU的硬件结构，只能被动地为其特定的硬件电路结构所利用。

如果脱离了已有的硬件环境（CPU），机器代码将失去意义。

此外，编译器作为一种软件运行，除了某种单一目标器件，即CPU的硬件结构外，不需要任何与硬件相关的器件库和工艺库参与编译。

因而，编译器的工作单纯得多，编译过程基本属于一种一一对应式的，机械转换式的“翻译”行为。

综合器则不同，同样是类似的软件代码（如VHDL程序），综合器转化的目标是底层的电路结构网表文件，这种满足原设计程序功能描述的电路结构不依赖于任何特定硬件环境，因此可以独立地存在，并能轻易地被移植到任何通用硬件环境中，如ASIC、FPGA等。

换言之，电路网表代表了特定的硬件结构，因此具备了随时改变硬件结构的依据。

综合的结果具有相对独立性。

另一方面，综合器在将硬件描述语言表达的电路功能转化成具体的电路结构网表过程中，具有明显的能动性和创造性，它不是机械的一一对应式的“翻译”，而是根据设计库、工艺库以及预先设置的各种约束条件，选择最优的方式完成电路结构的设计。

这就是说，对于相同的VHDL表述，综合器可以用不同的电路结构实现相同的功能。

## <<EDA技术实用教程>>

### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>