

<<CPLD/FPGA设计与应用高级教程>>

图书基本信息

书名：<<CPLD/FPGA设计与应用高级教程>>

13位ISBN编号：9787512402461

10位ISBN编号：7512402465

出版时间：2011-1

出版时间：北京航空航天大学出版社

作者：郭利文，邓月明 编著

页数：308

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>

## 内容概要

本书结合目前主流的cpld / fpga产品以及最流行的设计理念，系统、详细地介绍cpld / fpga的硬件结构、硬件描述语言与验证语言的基础应用以及高级应用；详细介绍如何使用verilog hdl语言进行有限状态机设计和testbench设计，以及如何使用modelsim进行功能仿真和时序仿真；简要介绍验证方法学的基本概念以及验证语言的比较，并就cpld / fpga的系统应用进行了详细探讨，包括dsp设计、嵌入式处理器设计、hardcopy设计、嵌入式逻辑分析仪的使用以及cpld / fpga的板级设计。

本书既可作为电子信息、通信工程以及相关工科专业的本科高年级学生和研究生教材，也可作为全国大学生电子设计竞赛的培训教材，以及从事电子电路系统设计与cpld / fpga / asic设计的工程技术人员的参考用书。

## 书籍目录

第1章 概述 1.1 数字电路基础及发展演变 1.2 cpld / fpga的介绍 1.3 设计语言及其方法的介绍 1.4 硬件语言与软件语言的区别 1.5 设计与验证流程 1.6 cpld / fpga的前景与展望 1.7 本章小结 1.8 思考与练习 第2章 cpld / fpga硬件结构 2.1 pld的分类 2.2 乘积项结构的基本原理 2.3 查找表结构的基本原理 2.4 传统cpld的基本结构 2.5 传统fpga的基本结构 2.6 最新cpld的基本结构 2.7 最新fpga的基本结构 2.8 cpld与fpga的选择 2.9 cpld / fpga的配置 2.10 本章小结 2.11 思考与练习 第3章 verilog hdl语法基础 3.1 verilog hdl的特点 3.2 verilog hdl的描述方式 3.3 模块和端口 3.4 注释 3.5 常量、变量与逻辑值 3.6 操作符 3.7 操作数 3.8 参数指令 3.9 编译指令 3.10 系统任务和系统函数 3.11 实例1：串并转换程序设计 3.12 本章小结 3.13 思考与练习 第4章 verilog的描述与参数化设计 4.1 数据流描述 4.2 行为级描述 4.3 结构化描述 4.4 高级编程语句 4.5 参数化设计 4.6 混合描述 4.7 实例2：i2c slave控制器的设计 4.8 本章小结 4.9 思考与练习 第5章 有限状态机设计 5.1 有限状态机的基本概念 5.2 状态机描述的基本语法 5.3 状态编码 5.4 状态初始化 5.5 full case 与parallel case 5.6 状态机的描述 5.7 实例3：pci slave接口设计 5.8 本章小结 5.9 思考与练习 第6章 约束与延时分析 6.1 约束的目的 6.2 引脚约束及电气标准设定 6.3 时序约束的基本概念 6.4 时序约束的本质 6.5 静态延时分析 6.6 统计静态延时分析 6.7 动态延时分析 6.8 实例4：建立时间和保持时间违例分析 6.9 时序违例及解决方式 6.10 实例5：四角测试中的时序分析 6.11 实例6：lpc slave接口设计 6.12 本章小结 6.13 思考与练习 第7章 rtl设计原则及技巧 7.1 rtl设计的主要原则 7.2 rtl设计的主要技巧 7.3 组合逻辑设计 7.4 时序逻辑设计 7.5 代码风格 7.6 实例8：信号消抖时的亚稳态及解决方案 7.7 本章小结 7.8 思考与练习 第8章 仿真与testbench设计 8.1 仿真概述 8.2 仿真器的选择 8.3 modelsim简介与仿真 8.4 testbench设计 8.5 testbench结构化 8.6 实例9：基于modelsim的i2c slavetestbench设计 8.7 实例10：基于modelsim的lpc slave接口仿真设计 8.8 实例11：基于modelsim的信号消抖程序仿真设计 8.9 本章小结 8.10 思考与练习 第9章 cpld / fpga的验证方法学 9.1 验证与仿真 9.2 验证与测试 9.3 验证的期望 9.4 验证的语言 9.5 断言 9.6 验证的分类 9.7 代码覆盖 9.8 验证工具 9.9 验证计划 9.10 dft 9.11 版本控制 9.12 实例12：基于fsm的sva断言验证设计 9.13 本章小结 9.14 思考与练习 第10章 cpld / fpga的高级应用 10.1 基于dsp的fpga设计 10.2 基于嵌入式处理器的fpga设计 10.3 典型的sopc运用：nios ii简介及应用 10.4 基于hardcopy技术的fpga设计 10.5 嵌入式逻辑分析仪 10.6 本章小结 10.7 思考与练习 第11章 cpld / fpga系统设计 11.1 常用电平标准及其接口设计 11.2 信号完整性概述 11.3 高速设计与serdes 11.4 电源完整性概述 11.5 功耗与热设计 11.6 pcb设计与cpld / fpga系统设计 11.7 实例16：基于 $\mu$ c / os-ii的fpga系统设计 11.8 本章小结 11.9 思考与练习 参考文献

## 章节摘录

与等效性检查和模型检查不同的一个类别是自动推理，它使用逻辑推理来证明具体实现和相关的规范是否一致，就好像一个数学形式的论证和推演一样，因此自动推理是建立在定理证明理论之上，验证者需要先提取系统的模型并表示为逻辑的命题、谓词、引理和定理等，并且确定待验证的性质。验证者通过不断的引导、不断地对现有条件包括已证明的定理应用规则来产生新的定理，直到推出所需要的定理为止。

自动推理相对于模型检查和等效性检查而言，最大的优点莫过于它既可以应用于无穷状态转换系统，又可以应用于有限的状态系统。

人为因素在自动推理中占有很重要的位置——依照目前的技术，自动推理还不能由计算机来自动完成，需要验证者具有相当丰富的证明经验才能得以实现。

等效性检查、模型检查以及自动推理互相补充，在验证过程中同时应用。

如果形式验证与仿真和断言结合，那么又可以分为静态形式验证和动态形式验证。

仿真器可以覆盖很大的范围，但是需要动态地产生测试激励或者验证环境。

另外有些仿真并不能完全覆盖设计中的每一项，特别是深藏在设计中的大量的交互逻辑行为。

静态形式验证中的最大的特点就是采用一个合适的工具读取设计的功能描述，然后尽可能地对逻辑进行分析以确保异常的状况不会出现。

它不需要仿真器就可以检查所有的状态空间，并且非常精准。

但是它只能检测设计中的一小部分，随着空间的呈指数型增大就有可能出现“状态空间爆炸”，因此在这种情况下使用静态形态验证就不再适合。

为了解决静态形态验证的局限性，动态形式验证把仿真和静态形式验证相结合来实现形式验证。

动态形式验证会直接设置一个边界条件——在设计中很难遇到或者很难达到的功能性条件。

在边界条件里面使用静态形态验证工具尽最大可能来评估边界条件，而不采用仿真器。

一旦达到一个边界条件，或者说静态形态验证已经完成了对边界条件的评估后，控制权就交给仿真器继续原来的仿真。

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>