

国外电子与通信教材系列

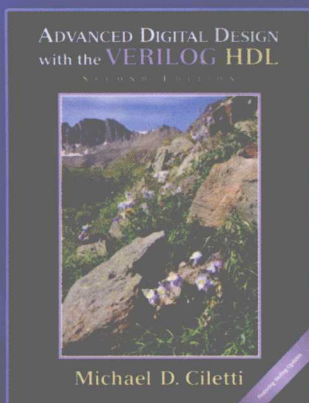
英文版

PEARSON

Verilog HDL

高级数字设计

(第二版)



Advanced Digital Design
with the Verilog HDL
Second Edition

[美] Michael D. Ciletti 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

国外电子与通信教材系列

Verilog HDL 高级数字设计

(第二版)

(英文版)

Advanced Digital Design with the Verilog HDL
Second Edition

[美] Michael D. Ciletti 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书依据数字集成电路系统工程开发的要求与特点,利用 Verilog HDL 对数字系统进行建模、设计与验证,对 ASIC/FPGA 系统芯片工程设计开发的关键技术与流程进行了深入讲解,内容包括:集成电路芯片系统的建模、电路结构权衡、流水线技术、多核微处理器、功能验证、时序分析、测试平台、故障模拟、可测性设计、逻辑综合、后综合验证等集成电路系统的前后端工程设计与实现中的关键技术及设计案例。书中以大量设计实例叙述了集成电路系统工程开发须遵循的原则、基本方法、实用技术、设计经验与技巧。

本书既可作为电子与通信、电子科学与技术、自动控制、计算机等专业领域的高年级本科生和研究生的教材或参考资料,也可用于电子系统设计及数字集成电路设计工程师的专业技术培训。

Original edition, entitled *Advanced Digital Design with the Verilog HDL, Second Edition, 9780136019282* by Michael D. Ciletti, published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2011 Pearson Education, Inc. All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, Copyright © 2010.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in the mainland of China exclusively (except Taiwan, Hong Kong SAR and Macau SAR).

本书英文影印版专有版权由 Pearson Education (培生教育出版集团) 授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

此版本仅限在中国大陆出版发行。

本书贴有 Pearson Education (培生教育出版集团) 激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2010-0895

图书在版编目(CIP)数据

Verilog HDL 高级数字设计:第2版:英文/(美)西勒提(Ciletti, M. D.)著.

北京:电子工业出版社,2010.4

(国外电子与通信教材系列)

ISBN 978-7-121-10477-0

I. V… II. 西… III. 硬件描述语言—Verilog HDL—程序设计—教材—英文 IV. TP312

中国版本图书馆CIP数据核字(2010)第037541号

策划编辑:马 岚

责任编辑:马 岚

印 刷:北京市天竺颖华印刷厂

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

开 本:787×980 1/16 印张:62.25 字数:1813千字

印 次:2010年4月第1次印刷

定 价:98.00元

凡所购买电子工业出版社的图书有缺损问题,请向购买书店调换;若书店售缺,请与本社发行部联系。联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

序

2001年7月间,电子工业出版社的领导同志邀请各高校十几位通信领域方面的老师,商量引进国外教材问题。与会同志对出版社提出的计划十分赞同,大家认为,这对我国通信事业、特别是对高等院校通信学科的教学工作会很有好处。

教材建设是高校教学建设的主要内容之一。编写、出版一本好的教材,意味着开设了一门好的课程,甚至可能预示着一个崭新学科的诞生。20世纪40年代MIT林肯实验室出版的一套28本雷达丛书,对近代电子学科、特别是对雷达技术的推动作用,就是一个很好的例子。

我国领导部门对教材建设一直非常重视。20世纪80年代,在原教委教材编审委员会的领导下,汇集了高等院校几百位富有教学经验的专家,编写、出版了一大批教材;很多院校还根据学校的特点和需要,陆续编写了大量的讲义和参考书。这些教材对高校的教学工作发挥了极好的作用。近年来,随着教学改革不断深入和科学技术的飞速进步,有的教材内容已比较陈旧、落后,难以适应教学的要求,特别是在电子学和通信技术发展神速、可以讲是日新月异的今天,如何适应这种情况,更是一个必须认真考虑的问题。解决这个问题,除了依靠高校的老师 and 专家撰写新的符合要求的教科书外,引进和出版一些国外优秀电子与通信教材,尤其是有选择地引进一批英文原版教材,是会有好处的。

一年多来,电子工业出版社为此做了很多工作。他们成立了一个“国外电子与通信教材系列”项目组,选派了富有经验的业务骨干负责有关工作,收集了230余种通信教材和参考书的详细资料,调来了100余种原版教材样书,依靠由20余位专家组成的出版委员会,从中精选了40多种,内容丰富,覆盖了电路理论与应用、信号与系统、数字信号处理、微电子、通信系统、电磁场与微波等方面,既可作为通信专业本科生和研究生的教学用书,也可作为有关专业人员的参考材料。此外,这批教材,有的翻译为中文,还有部分教材直接影印出版,以供教师用英语直接授课。希望这些教材的引进和出版对高校通信教学和教材改革能起一定作用。

在这里,我还要感谢参加工作的各位教授、专家、老师与参加翻译、编辑和出版的同志们。各位专家认真负责、严谨细致、不辞辛劳、不怕琐碎和精益求精的态度,充分体现了中国教育工作者和出版工作者的良好美德。

随着我国经济建设的发展和科学技术的不断进步,对高校教学工作会不断提出新的要求和希望。我想,无论如何,要做好引进国外教材的工作,一定要联系我国的实际。教材和学术专著不同,既要注意科学性、学术性,也要重视可读性,要深入浅出,便于读者自学;引进的教材要适应高校教学改革的需要,针对目前一些教材内容较为陈旧的问题,有目的地引进一些先进的和正在发展中的交叉学科的参考书;要与国内出版的教材相配套,安排好出版英文原版教材和翻译教材的比例。我们努力使这套教材能尽量满足上述要求,希望它们能放在学生们的课桌上,发挥一定的作用。

最后,预祝“国外电子与通信教材系列”项目取得成功,为我国电子与通信教学和通信产业的发展培土施肥。也恳切希望读者能对这些书籍的不足之处、特别是翻译中存在的问题,提出意见和建议,以便再版时更正。



中国工程院院士、清华大学教授

“国外电子与通信教材系列”出版委员会主任

出版说明

进入21世纪以来,我国信息产业在生产和科研方面都大大加快了发展速度,并已成为国民经济发展的支柱产业之一。但是,与世界上其他信息产业发达的国家相比,我国在技术开发、教育培训等方面都还存在着较大的差距。特别是在加入WTO后的今天,我国信息产业面临着国外竞争对手的严峻挑战。

作为我国信息产业的专业科技出版社,我们始终关注着全球电子信息技术的发展方向,始终把引进国外优秀电子与通信信息技术教材和专业书籍放在我们工作的重要位置上。在2000年至2001年间,我社先后从世界著名出版公司引进出版了40余种教材,形成了一套“国外计算机科学教材系列”,在全国高校以及科研部门中受到了欢迎和好评,得到了计算机领域的广大教师与科研工作者的充分肯定。

引进和出版一些国外优秀电子与通信教材,尤其是有选择地引进一批英文原版教材,将有助于我国信息产业培养具有国际竞争能力的技术人才,也将有助于我国国内在电子与通信教学工作中掌握和跟踪国际发展水平。根据国内信息产业的现状、教育部《关于“十五”期间普通高等教育教材建设与改革的意见》的指示精神以及高等院校老师们反映的各种意见,我们决定引进“国外电子与通信教材系列”,并随后开展了大量准备工作。此次引进的国外电子与通信教材均来自国际著名出版商,其中影印教材约占一半。教材内容涉及的学科方向包括电路理论与应用、信号与系统、数字信号处理、微电子、通信系统、电磁场与微波等,其中既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择 and 自由组合使用。我们还将与国外出版商一起,陆续推出一些教材的教学支持资料,为授课教师提供帮助。

此外,“国外电子与通信教材系列”的引进和出版工作得到了教育部高等教育司的大力支持和帮助,其中的部分引进教材已通过“教育部高等学校电子信息科学与工程类专业教学指导委员会”的审核,并得到教育部高等教育司的批准,纳入了“教育部高等教育司推荐——国外优秀信息科学与技术系列教学用书”。

为做好该系列教材的翻译工作,我们聘请了清华大学、北京大学、北京邮电大学、南京邮电大学、东南大学、西安交通大学、天津大学、西安电子科技大学、电子科技大学、中山大学、哈尔滨工业大学、西南交通大学等著名高校的教授和骨干教师参与教材的翻译和审校工作。许多教授在国内电子与通信专业领域享有较高的声望,具有丰富的教学经验,他们的渊博学识从根本上保证了教材的翻译质量和专业学术方面的严格与准确。我们在此对他们的辛勤工作与贡献表示衷心的感谢。此外,对于编辑的选择,我们达到了专业对口;对于从英文原书中发现的错误,我们通过作者联络、从网上下载勘误表等方式,逐一进行了修订;同时,我们对审校、排版、印制质量进行了严格把关。

今后,我们将进一步加强同各高校教师的密切关系,努力引进更多的国外优秀教材和教学参考书,为我国电子与通信教材达到世界先进水平而努力。由于我们对国内外电子与通信教育的发展仍存在一些认识上的不足,在选题、翻译、出版等方面的工作中还有许多需要改进的地方,恳请广大师生和读者提出批评及建议。

电子工业出版社

教材出版委员会

主任	吴佑寿	中国工程院院士、清华大学教授
副主任	林金桐	北京邮电大学校长、教授、博士生导师
	杨千里	总参通信部副部长，中国电子学会会士、副理事长 中国通信学会常务理事、博士生导师
委员	林孝康	清华大学教授、博士生导师、电子工程系副主任、通信与微波研究所所长 教育部电子信息科学与工程类专业教学指导分委员会委员
	徐安士	北京大学教授、博士生导师、电子学系主任
	樊昌信	西安电子科技大学教授、博士生导师 中国通信学会理事、IEEE 会士
	程时昕	东南大学教授、博士生导师
	郁道银	天津大学副校长、教授、博士生导师 教育部电子信息科学与工程类专业教学指导分委员会委员
	阮秋琦	北京交通大学教授、博士生导师 计算机与信息技术学院院长、信息科学研究所所长 国务院学位委员会学科评议组成员
	张晓林	北京航空航天大学教授、博士生导师、电子信息工程学院院长 教育部电子信息科学与电气信息类基础课程教学指导分委员会副主任委员 中国电子学会常务理事
	郑宝玉	南京邮电大学副校长、教授、博士生导师 教育部电子信息与电气学科教学指导委员会委员
	朱世华	西安交通大学副校长、教授、博士生导师 教育部电子信息科学与工程类专业教学指导分委员会副主任委员
	彭启琮	电子科技大学教授、博士生导师、通信与信息工程学院院长 教育部电子信息科学与电气信息类基础课程教学指导分委员会委员
	毛军发	上海交通大学教授、博士生导师、电子信息与电气工程学院副院长 教育部电子信息与电气学科教学指导委员会委员
	赵尔沅	北京邮电大学教授、《中国邮电高校学报（英文版）》编委会主任
	钟允若	原邮电科学研究院副院长、总工程师
	刘彩	中国通信学会副理事长兼秘书长，教授级高级工程师 信息产业部通信科技委副主任
	杜振民	电子工业出版社原副社长
	王志功	东南大学教授、博士生导师、射频与光电集成电路研究所所长 教育部高等学校电子电气基础课程教学指导分委员会主任委员
	张中兆	哈尔滨工业大学教授、博士生导师、电子与信息技术研究院长
	范平志	西南交通大学教授、博士生导师、信息科学与技术学院院长

前 言

精炼、明晰化与验证

用硬件描述语言(HDL)建立行为级模型是现代专用集成电路设计的关键技术。如今,大多数设计者使用基于硬件描述语言的设计方法,创建基于语言的高层、抽象的电路描述,以验证其功能和时序。在本书第一版的使用过程中,讲授设计方法学所用的语言(IEEE 1464-1995)已经历了两次修改,分别是IEEE 1364-2001及2005年的修订版,即Verilog-2001和Verilog-2005,以提高其有效性和效率。

这一版的编写动机和第一版基本是相同的。对那些准备在产品研发团队做出成绩的学生们来说,必须了解如何在设计流程的关键阶段使用硬件描述语言。因此,需要有一门在内容上超越先修课程“数字设计”中学习过的基本原则和方法的课程,本书就是为该课程而著的。

现在,市面上讨论硬件描述语言的书籍的数量已远远超过本书第一版出版时的数量。但是,这些书大部分都定位于解释语法,而不是如何运用语言进行设计,不太适合于课堂教学。本书的重点是硬件描述语言的设计方法学,因此语言本身只是一个配角。这一版中强化了如何通过实例证明,将一个数字系统描述并划分为数据通路、状态(反馈)信号和控制器(有限状态机)系统结构的重要性。我们认为,这种描述可使设计和验证复杂数字系统的方法更加清楚、直接、明了。本书给出了大量的仿真结果和注释,以帮助学生掌握时序机的操作过程,并深入理解由控制器产生的信号间的时序互动关系,数据通路的操作,以及从数据通路回馈给控制器的信号。其目的都是为了设计开发出可综合,无锁存且无竞争的设计。

Verilog 2001和2005的语言增强功能已用于重新描述和简化书中模型的代码。我们强调工业界通用的规范和风格,但并不鼓励不考虑模型能否被综合的学术模型风格。对于部分习题的答案及书中未包含的其他一些练习题的答案,授课教师可通过配套网站

<http://www.pearsonhighered.com/ciletti>

获得。本书第二版已把第一版中处理同步FIFO的部分改为同步和异步FIFO,并给出了精心设计的例子,以解释使用异步FIFO来同步跨越时钟域的数据传输问题。

书中的设计实例已多次优化和改进。从设计方法学的角度,对一个嵌入式控制器,用C语言建模和用Verilog建模,这两种设计方法学之间存在着竞争和互补的关系。基于C的方法执行陈述性语句,而Verilog HDL模拟了某个机器的多个并发的行为动作。后一种设计方法对硬件进行编译,而前一种是编译预先存储在硬件单元中的语句。对于某个特定应用,Verilog模

型编译的硬件在主机接口处生成了等效的I/O信号。对于嵌入式代码而言,其区别是不会产生等效的硬件。本书的目标就是讲授硬件建模/编译的范例,并预测综合实现后的结果。C语言编程是预测程序产生的数据,而状态机/处理器的应用却显而易见。作为对比,用Verilog描述的模型预测该硬件将产生应用所需求的I/O信号,因此需要开发者根据寄存器操作时序控制进行思考和设计。Verilog的模型鼓励学习者理解一个数字电路和系统的本质。

本书要求学生已学过逻辑设计的入门课程,本书的目标是:(1)简要复习组合时序逻辑的基本原理,(2)介绍HDL在设计中的应用,(3)强调的是快速设计通过ASIC和/或FPGA实现的电路设计描述风格,(4)提供具有一定难度的设计实例。章末习题的目的是鼓励学生精炼、明晰化并验证他们自己的设计。从本质上讲,许多习题均为开放式的设计,要求验证以达到所要求的设计规范。

广泛使用的Verilog硬件描述语言(IEEE 1364标准),作为一个公共框架为本书的设计实例的讨论提供了支持。第一版重点关注数字电路的设计、验证和综合,而不是Verilog语言本身的语法,本版仍然保持这种风格。

选修数字设计中级课程的大多数学生至少应该熟悉一种编程语言,并且在阅读本书时能够将其作为可以借鉴的背景知识。本书仅讨论Verilog的核心设计方法及其广泛使用的特性。为了强调在面向综合的设计环境中使用该语言,我们还特意将许多语法的细节、特点和解释放在附录中及出版社的网站上,以便于读者参考。附录中也提供了Verilog的所有形式化语法。

大部分数字设计的入门课程都介绍过通过状态转移图表示的有限状态机及算法状态图(ASM)。同样,本书中也大量使用了ASM图,演示了其在设计时序状态机的行为模型中的功用。对利用ASMD图(即通过标注显示出被控数据通道的寄存器操作的ASM图)系统地设计有限状态机来控制数字状态机中复杂数据通道的重要问题,进行了深入论述。并将精简指令集计算机中央处理器(RISC CPU)和其他重要硬件单元的设计作为实例给出。我们的支持网站上包含了RISC计算机的源代码和可用于应用程序开发的汇编程序。这个汇编程序也可作为研究鲁棒性更好的指令集和其他派生架构的基础。

本书完整地引入了Verilog语言,但仅在支持设计实例的需要时才进行详细说明。正文中使用了大量的实例,讲解使用Verilog硬件描述语言进行VLSI电路设计时的重要和关键设计步骤。设计实例的源代码都经过了验证,并且所有实例的源代码和测试平台都可以从出版社的网站下载。

读者对象

本书适用于学习高级数字系统设计课程的学生,以及那些想通过实例学习Verilog的现代集成电路设计专业工程师。本书适合电子工程、计算机工程和计算机科学等专业的高年级本科生和低年级研究生,也适合学习过逻辑设计入门课程的专业工程师使用。本书假定读者具有布尔代数及其在逻辑电路设计中应用的背景知识,并熟悉同步时序有限状态机。在此基础

上, 本书讨论了一些应用于计算机系统、数字信号处理、图像处理、跨时钟域的数据传输、内置自测试 (BIST) 和一些其他应用的重要电路的设计实例。这些实例涵盖了建模、架构的设计折中、流水线技术、多处理器执行、功能验证、定时分析、测试生成、故障模拟、可测性分析、逻辑综合和综合后验证的关键设计问题。

本版的新颖之处

- 探索了 Verilog 2001 和 2005 的主要特点
- 阐述并推广基于 Verilog 2001 和 2005, 且可综合的寄存器传输级 (RTL) 描述和算法建模的设计风格
- 深入讨论基于 Verilog 2001 和 2005 的数字处理系统 (如图像处理器、数字滤波器和环形缓冲器) 算法和架构
- 给出了基于 Verilog 2001 和 2005 语言的综合设计实例 (如 RISC 计算机和各种数据通道控制器)
- 提供了大量有评注和解释的仿真结果的图形化描述
- 给出了 150 多个经过完全验证的基于 Verilog 2001 和 2005 的设计实例
- 含有利用 Verilog 2001 和 2005 编写的具备 JTAG 和 BIST 可测功能的实用设计案例
- 附录中给出了 Verilog 2001 和 2005 HDL 的语法形式
- 讨论了异步和同步 FIFO 设计

本书特色

- 简要回顾了组合时序逻辑设计的基本原则
- 重点讨论现代数字设计方法
- 说明了行为级建模中 ASM 和 ASMD 图的作用
- 明确指出了可综合和不可综合循环的区别
- 通过实例对时序分析、故障模拟、测试和可测性设计进行切合实际的讨论
- 每章后均设计了一些涉及面广且难度高的习题
- 给出了本书中所有实例模型的索引
- 包含一套与本书内容配套的可适合实验室实验验证的 FPGA 设计实例, 如算术逻辑单元 (ALU), 可编程电子锁, 有 FIFO 的键盘扫描器, 可纠错的串行通信接口, 基于 SRAM 的控制器, 先入先出 (FIFO) 存储器及 RISC CPU
- 本书支持网站内容包含:
 1. 例子中涉及的所有模型的源文件
 2. 用于仿真实例的测试平台源文件
 3. Instructor's Classroom Kit 中包含了可用于课堂讲授的按主题划分的幻灯片文件夹

4. 部分习题的解答
5. 额外的习题
6. 提供帮助学生使用某些工具软件（如仿真器）尽快得到结果的速成教案
7. 常见问题解答（FAQ）

课程讲授次序

本书首先对组合逻辑设计做了简要介绍和回顾，接着描述了一个 ASIC 或 FPGA 的设计流程。按照书中内容的顺序，第 1 章至第 6 章利用综合的方法来研究设计了一些题目和内容。但是，阅读第 7 章至第 10 章时，则不必按照书中的顺序。课后作业具有挑战性，而且基于 FPGA 的实验练习适于同步实验或学期末的课题。第 10 章列出了一些算术运算的架构，覆盖了较多的应用范围。第 11 章介绍了后综合设计验证、时序分析、故障模拟和可测试性设计。根据课程教学的深度和重点，本章涵盖的内容和范围也可省略。

说明

我们没有坚持常规使用大写和小写字体，或使用代码清单专用字体。本书的选择一直是基于最大化的整体视觉效果及所列代码的可读性。我们认为，设计实例中的代码得到正确表达才是至关重要的。模块框图已被简化，以减少视觉混乱。所以，我们通常只显示信号的实际外部名称，而省略其形式的内在对应名称。由于 D 触发器在现代 EDA 工具的综合中起着主导作用，因此书中几乎唯一性地使用了 D 触发器。

各章概述

第 1 章简要论述了硬件描述语言在基于库单元的 ASIC 和 FPGA 设计流程中的作用。第 2 章和第 3 章则根据传统的教学方式（例如卡诺图算法），回顾了数字设计先修课程中涉及的主要知识。这些资料可以奠定读者的数字设计的背景知识，便于之后利用实例介绍许多基于硬件描述语言的数字设计方法。第 4 章和第 5 章介绍了组合电路与时序电路的 Verilog 语言建模方法，重点强调了行为级建模中的代码编写风格。第 6 章着重基于库单元的 ASIC 综合，介绍了组合逻辑与时序逻辑的综合。这一章追求两个主要目标：(1) 提出可综合的代码描述风格；(2) 建立能够让读者预测综合结果的基础知识和能力。尤其在对时序状态机综合时，通常会把时序状态机分成数据通道和控制通道两部分来编写。第 7 章介绍了一些例子，这些例子描述了怎样设计一个数据通道的控制器，包括带有从数据通道反馈给控制器的状态信号的状态机设计。而精简指令集（RISC）CPU 设计和通用异步收发器（UART，用于系统间传输数据的电路）的设计作为这个例子的应用平台。第 8 章讲述了可编程逻辑器件（PLD）、复杂 PLD、只读存储器（ROM）和静态随机存储器（SRAM）的知识，并将综合目标扩展为 FPGA 的综合。第 9 章主要涉及计

算机结构、数字滤波器和其他信号处理器中有关的计算单元和算法的建模和综合。第10章研究并描述了数字状态机中计算单元的算法和结构。在第11章中,使用 Verilog 语言,结合故障仿真器和时序分析器,重新审查了之前设计的状态机选择方案,并考虑性能、时序问题及可测性问题,来优化和完善这个主要取决于设计者的设计流程和任务。本章建模的测试访问端口(TAP)控制器由 IEEE 1149.1 标准定义(即俗称的 JTAG 标准),并提出了其应用实例。另外还给出了一个内建自测试(BIST)的详细实例。

致谢

本书作者非常感谢曾为本书做出贡献并提出宝贵意见的同事和学生们的支持。本书是我的研发经验和在科罗拉多大学教学经历的综合成果,也包括我在惠普、福特微电子公司和 Prisma 公司的工作经历,在荷兰的 Delft 技术大学的经验,以及在欧洲和亚洲的短期课程的教学经验。虽然其中有的公司如今已成回忆,但是我仍然深深感谢这些公司和科罗拉多大学对我进行 VLSI 电路设计研究工作的支持。本书手稿的第一版审稿人也提出了鼓励、关键内容的调整与许多有益的建议。我非常感谢 Jim Tracey 博士和 Rodger Ziemer 博士,他们支持并鼓励了我在 VLSI 电路设计方面的努力和成就,我也十分感谢福特微电子公司的 Deepak Goel 先生,他向我介绍了后来成为最先进的 VLSI 设计平台的福特微电子的 Daisy 工作站。感谢 Simucad 公司的 Bill Fuchs 先生,他帮助我获取了工业级的 Verilog 仿真器。感谢惠普公司的 Tom Saponas 和 Dave Ritchey 先生,他们给我机会领导完成一个动态时序分析器的反向设计工程,两名学生 David Uranek 和 Jerry Barnett 参与并获成功。十分感谢我在 Prisma 公司暑期工作的主管 Dave Still 先生,给我提供了设计环境与精神鼓励,使我完成了高性能多核系统中建模的难题。感谢 Sutherland HDL 的 Stu Sutherland 帮助我理解并更深入研究了数字系统建模中的竞争条件问题,这些观点使我坚持使用非阻塞赋值来进行边缘敏感的行为级建模和使用阻塞赋值进行电平敏感的行为级建模的描述风格,让我更好地帮助学生理解同步数字系统的操作和设计。谢谢我的朋友兼同行、瑞士联邦理工学院的 Hubert Kaeslin 博士,与他进行的有意义的讨论让我能更深入钻研数字处理器的算法和结构。感谢 Kirk Sprague 和 Scott Kukel 帮助研发了一个可用于 UART 的汉明编码器。同时感谢 Cris Hagan,他的论文提供了本书第9章的数字信号处理器中的抽取器和其他功能单元建模。非常感谢 Rex Anderson 先生帮助校正了几章的内容,并对第一版进行了修改。谢谢我的学生 Terry Hansen 和 Lisa Horton,他们提供了咖啡自动贩卖机例子的灵感,并开发了支持 RISC CPU 的汇编代码。同时我还要感谢科罗拉多大学的 Greg Tumbush 教授和 Temple 大学的 Chen-Huan Chiang 教授为本书第二版提供了重要建议,也谢谢许多学生的课堂讨论,他们的发言为第二版提供了帮助。谢谢 Scott Disanno 和 Irwin Zucker 领导了第二版的出版,谢谢 Haseen Khan 精心策划本书的结构。我向所有给予本书支持的朋友表示衷心的感谢!

Preface

Simplify, Clarify, and Verify

Behavioral modeling with a hardware description language (HDL) is the key to modern design of application-specific integrated circuits (ASICs). Today, most designers use an HDL-based design method to create a high-level, language-based, abstract description of a circuit, and verify its functionality and timing. The language used to teach design methodology in the first edition of this text, IEEE 1464-1995, has undergone two revisions to improve the effectiveness and efficiency of the language: IEEE 1364-2001 followed by a revision in 2005, known as Verilog-2001 and Verilog-2005, respectively.

The motivation behind this edition is basically the same as that which guided the first edition: students preparing to contribute to a productive design team must know how to use a HDL at key stages of the design flow. Thus, there is a need for a course going beyond the basic principles and methods learned in a first course in digital design. This book is written for such a course.

The quantity of books discussing HDLs far exceeds that which was available at the time of the first edition, and most of these are still oriented toward explanations of language syntax, rather than toward design, and are not well-suited for classroom use. Our focus is on design methodology enabled by an HDL. Thus, the language itself has a subordinate role. In this edition, we have made a strong effort to demonstrate by examples the importance of partitioning a digital machine to expose its datapath, status (feedback) signals, and controller (finite state machine). This effort leads, we think, to a much clearer and straightforward approach to designing and verifying complex digital machines. We present an abundance of simulation results, with annotation to help students (1) understand the operation of a sequential machine and (2) appreciate the time-sequential interaction between the signals produced by the controller, the operations in the datapath, and the signals reported back to the controller from the datapath, all with the aim of developing synthesizable, latch-free, race-free designs.

The language enhancements of Verilog 2001, 2005 have been used to reexamine and simplify the code of our models. We emphasize industry practices, and do not unwittingly encourage academic styles of modeling without regard for whether a model can be synthesized. Solutions to selected problems, together with additional solved exercises that are not included in the text, are available for instructors at the companion Web site: <http://www.pearsonhighered.com/ciletti>. The first edition treated synchronous FIFOs; this edition treats synchronous and asynchronous FIFOs, and presents

an elaborate example of using an asynchronous FIFO to synchronize transfer of data across clock domains.

Design practice is always in a flux. One tension is between the approach of writing a model in C for an embedded controller versus writing a model in Verilog. The C-based approach executes statements; Verilog models execution of multiple concurrent behaviors of a machine. The latter approach compiles hardware; the former compiles statements for a preexisting hardware unit. The compiled hardware of the Verilog model produces equivalent I/O signals at the interface to the host machine for a particular application. The distinction here is that embedded code does not equal embedded hardware. This text aims to teach the hardware modeling/compilation paradigm and to anticipate the results of synthesis. Programming in C anticipates the data produced by a program, and the machine/processor is itself transparent. In contrast, modeling in Verilog anticipates the hardware that will produce the I/O signals demanded by the application, and requires the developer to think in terms of time-sequential control of register operations. Verilog modeling encourages this understanding about the nature of a digital machine.

Our goal in this book is to build on a student's background from a first course in logic design by (1) briefly reviewing basic principles of combinational and sequential logic, (2) introducing the use of HDLs in design, (3) emphasizing descriptive styles that will allow the reader to quickly design working circuits suitable for ASICs and/or field-programmable gate array (FPGA) implementation, and (4) providing design examples having a range of difficulty. The end-of-chapter problems encourage students to simplify, clarify, and verify their designs. The nature of many of the problems is that of open-ended design, requiring verification that the design meets a prescribed specification.

The widely used Verilog HDL (IEEE Standard 1364) serves as a common framework supporting the design activities treated in this book. The first edition focused on developing, verifying, and synthesizing designs of digital circuits, and *not* on the Verilog language. This edition maintains that focus.

Most students taking a second course in digital design will be familiar with at least one programming language and will be able to draw on that background in reading this text. We cover only the core and most widely used features of Verilog. In order to emphasize *using* the language in a synthesis-oriented design environment, we have purposely placed many details, features, and explanations of syntax in the appendices and at the publisher's Web site for reference on an "as needed" basis, but an appendix provides the complete formal syntax of the language.

Most entry-level courses in digital design introduce finite state machines using state transition diagrams, and algorithmic state machine (ASM) charts. We do likewise, but we make heavier use of ASM charts and demonstrate their utility in developing behavioral models of sequential machines. The important problem of systematically designing a finite state machine to control a datapath in a complex digital machine is treated in-depth with ASMD charts, i.e. ASM charts annotated to display the register operations of the controlled datapath. The design of a RISC CPU and other important hardware units are given as examples. Our companion Web site includes the RISC machine's source code and an assembler that can be used to develop programs for applications. The machine also serves as a starting point for developing a more robust instruction set and architectural variants.

The Verilog language is introduced in an integrated, but selective manner, only as needed to support design examples. The text has a large set of examples illustrating how to address the key steps in a VLSI circuit design methodology using the Verilog HDL. The source code of the examples has been verified to be correct. Source code for all of the examples and their testbenches are available at the publisher's Web site.

The Intended Audience

This book is for students in an advanced course in digital design, and for professional engineers interested in learning Verilog by example, in the context of its use in the design flow of modern integrated circuits. The level of presentation is appropriate for seniors and first-year graduate students in electrical engineering, computer engineering, and computer science, as well as for professional engineers who have had an introductory course in logic design. The book presumes that the reader has a basic background in Boolean algebra and its use in logic circuit design, and a familiarity with synchronous finite state machines. Building on this foundation, the book addresses the design of several important circuits used in computer systems, digital signal processing, image processing, data transfer across clock domains, built-in self-test (BIST) and, other applications. The examples cover the key design problems of modeling, architectural trade-offs, pipelining, multiprocessor implementations, functional verification, timing analysis, test generation, fault simulation, design for testability, logic synthesis, and post-synthesis verification.

What's New in this Edition

- Exploits key features of Verilog 2001, 2005
- Illustrates and promotes a synthesis-ready style of register transfer level (RTL) and algorithmic modeling with Verilog 2001, 2005
- Provides an in-depth treatment of algorithms and architectures for digital machines (e.g. an image processor, digital filters, and circular buffers) with Verilog 2001, 2005
- Includes comprehensive design examples (e.g. a RISC machine and various datapath controllers) with Verilog 2001, 2005
- Includes numerous annotated and explained graphical illustrations of simulation results
- Contains over 150 fully verified examples with Verilog 2001, 2005
- Contains a worked example with JTAG and BIST for testing with Verilog 2001, 2005
- Contains an Appendix with full formal syntax of the Verilog 2001, 2005 HDL
- Treats asynchronous and synchronous FIFOs

Special Features of the Book

- Provides a brief review of basic principles in combinational and sequential logic
- Focuses on modern digital design methodology
- Demonstrates the utility of ASM and ASMD charts for behavioral modeling
- Clearly distinguishes between synthesizable and nonsynthesizable loops
- Provides practical treatment of timing analysis, fault simulation, testing, and design for testability, with examples
- Provides several problems with a wide range of difficulty after each chapter
- Combines a solution manual with an on-line repository of additional worked exercises
- Lists an index of all models developed in the examples
- Includes a set of FPGA-based, lab-ready exercises linked to the book (e.g. arithmetic and logic unit (ALU), programmable lock, a keypad scanner with a FIFO, a serial communications link with error correction, an SRAM controller, and first in, first out (FIFO) memory, RISC CPU, and FIFO)
- Supported by an ongoing Companion Web site containing:
 1. Source files of all models developed in the examples
 2. Source files of testbenches for simulating all of the examples
 3. An Instructor's Classroom Kit containing transparency files for a complete course based on the subject matter is available for instructors only

-
4. Solutions to selected problems is available for instructors only
 5. Additional worked problems
 6. Jump-start tutorials helping students get immediate results with selected software tools (e.g. simulator)
 7. Answers to frequently asked questions (FAQs)

Sequences for Course Presentation

The material in the text begins with a brief review of combinational and sequential logic design, but then progresses in the order dictated by the design flow for an ASIC or an FPGA. Chapters 1 to 6 treat design topics through synthesis, and should be covered in order, but Chapters 7 to 10 can be covered in any order. The homework exercises are challenging, and the lab-ready FPGA-based exercises are suitable for a companion lab or for end-of-semester projects. Chapter 10 presents several architectures for arithmetic operations, affording a diversity of coverage. Chapter 11 treats post-synthesis design validation, timing analysis, fault simulation, and design for testability. The coverage of these topics can be omitted, depending on the level and focus of the course.

Some Caveats

We do not adhere to common practice for using upper and lower case text, or for using courier font in code listings. Our choices have been based on maximizing the overall visual appeal and readability of the listed code. The visual result offsets, we think, the extra care required to ensure that the code is composed correctly in our examples. Block diagrams have been simplified to reduce the visual clutter, so we typically show only the actual, external names of signals, and omit their formal, internal counterparts. D-type flip-flops are used almost exclusively because they play a dominant role in synthesis with modern EDA tools.

Chapter Descriptions

Chapter 1 briefly discusses the role of HDLs in design flows for cell-based ASICs and FPGAs. Chapters 2 and 3 review mainstream topics that would be covered in a first course in digital design, using classical methods, i.e. Karnaugh maps. This material will refresh the reader's background, and the examples will be used later to introduce HDL-based methods of design. Chapters 4 and 5 introduce modeling of combinational and sequential logic with the Verilog HDL, and place emphasis on coding styles that are used in behavioral modeling. Chapter 6 addresses cell-based synthesis of ASICs, and introduces synthesis of combinational and sequential logic. Here we pursue two main objectives: (1) present synthesis-friendly coding styles and (2) form a foundation that will enable the reader to anticipate the results of synthesis, especially when synthesizing sequential machines. Many sequential machines are partitioned into a datapath and a controller. Chapter 7 covers examples that illustrate how to design a controller for a datapath, including machines having feedback of status signals from the datapath to the controller. The designs of a simple RISC CPU and a UART¹ serve as platforms for the subject matter.

¹Universal Asynchronous Receiver and Transmitter (UART), a circuit used in data transmission between systems.

Chapter 8 covers PLDs, complex PLDs (CPLDs), ROMs, and SRAMs, and then expands the synthesis target to include FPGAs. Chapter 9 treats the modeling and synthesis of computational units and algorithms found in computer architectures, digital filters, and other signal processors. Chapter 10 develops and refines algorithms and architectures for the arithmetic units of digital machines. In Chapter 11, we use the Verilog HDL in conjunction with fault simulators and timing analyzers to revisit a selection of previously designed machines and consider performance/timing issues and testability, to complete the treatment of design flow tasks that rely heavily on designer intervention. This chapter models the test access port (TAP) controller defined by IEEE 1149.1 standard (commonly known as the JTAG standard), and presents an example of its use. Another elaborate example covers BIST.

Acknowledgments

The author is grateful for the support of colleagues and students who expanded his vision of Verilog and contributed to this text. It represents the combined experience of my developing and teaching courses at the University of Colorado, taking sabbaticals in industry with Hewlett-Packard and Ford Microelectronics Inc., Prisma Inc., spending a sabbatical at the Technical University of Delft (Netherlands), and developing and presenting short courses in Europe and Asia. Some of the companies now exist only in memory, but I am deeply grateful to many individuals in industry and at the University of Colorado for their helping to shape my path in the realm of VLSI circuit design. The reviewers of the original manuscript for the first edition provided encouragement, critical judgment, and many helpful suggestions. Dr. Jim Tracey and Dr. Rodger Ziemer supported and encouraged my initial efforts to focus my work on the design of VLSI circuits. Deepak Goel (Ford Microelectronics) introduced me to VLSI design on the then state-of-the-art Daisy workstations at Ford Microelectronics. Bill Fuchs (Simucad, Inc.) supported my efforts to acquire an industrial-strength Verilog simulator. Tom Saponas and Dave Ritchey (Hewlett-Packard) placed me at the helm of a project to reverse-engineer a dynamic timing analyzer. Two students, David Uranek and Jerry Barnett, assured success. Dave Still (Prisma Corp.), my manager during a summer job, provided the environment and encouragement for me to tackle a significant problem in modeling of a high-performance, multicore system; Stu Sutherland (Sutherland HDL) helped the author gain a deeper appreciation for the issue of race conditions that can creep into the models of a digital system. These insights led to my adhering to the disciplined style of using nonblocking assignments for modeling edge-sensitive behavior and blocked assignments for modeling level-sensitive behavior, and to my efforts to help students understand the operation and design of synchronous digital machines. Rich discussions with Dr. Hubert Kaeslin, friend and colleague (Swiss Federal Institute of Technology – Zurich), led to my delving more deeply into algorithms and architectures for digital processors. Kirk Sprague and Scott Kukel were helpful in developing a Hamming encoder to work with a UART. Cris Hagan's thesis led to the models presented in Chapter 9 for decimators and other functional units found in digital signal processors. Rex Anderson proofread several chapters and scrubbed down the work of the first edition. Students Terry Hansen and Lisa Horton provided the inspiration for the coffee vending machine example, and developed the assembler that supports the RISC CPU. Thanks also to the many students whose classroom dialogue was helpful in the second edition. Profs. Greg Tumbush (University of Colorado at Colorado Springs) and Chen-Huan Chiang (Temple University) provided critical and helpful suggestions for this edition. Thanks also to the many students whose classroom dialogue was helpful in the second edition. Scott Disanno and Irwin Zucker shepherded this edition through the publication process, and Haseen Khan orchestrated the composition of the book from my manuscript. My deep thanks to all of you.

Dedication

This book is dedicated to the memory of Sr. Laurencia Rihn, RSM, and Fr. Jerry Wilson, CSC. My life has been shaped by their faith, encouragement, and love. To my wife, Jerilynn, and our children, Monica, Lucy, Rebecca, Christine, and Michael, their spouses, Mike McCormick, David Steigerwald, Peter Van Dusen, and Michelle Puhr Ciletti, and our grandchildren, the “cousin dozen”: Michael Angus, Katherine, Brigid, David, Jackson, Samantha, Peter, Matthew, Ella, Anthony, Abigail, and Joseph—thank you for the journey and love we’ve shared.