

<<Verilog数字系统设计教程>>

图书基本信息

书名：<<Verilog数字系统设计教程>>

13位ISBN编号：9787811243093

10位ISBN编号：7811243091

出版时间：2008-6

出版时间：北京航空航天大学出版社

作者：夏宇闻

页数：477

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>

<<Verilog数字系统设计教程>>

前言

数字信号处理（DSP）系统的研究人员一直在努力寻找各种优化的算法来解决相关的信号处理问题。

当他们产生了比较理想的算法思路后，就在计算机上用C语言或其他语言程序来验证该算法，并不断修改以期完善，然后与别的算法作性能比较。

在现代通信和计算机系统中，对于DSP算法评价最重要的指标是看它能否满足工程上的需要。

而许多工程上的需要都有实时响应的要求，也就是所设计的数字信号处理（DSP）系统必须在限定的时间内，如在几个毫秒（ms）甚至于几个微秒（ μs ）内，对所输入的大量数据完成相当复杂的运算，并输出处理结果。

这时如果仅仅使用通用的微处理器，即使是专用于信号处理的微处理器，往往也无法满足实时响应的要求。

因此，不得不设计专用的高速硬线逻辑来完成这样的运算。

设计这样的有苛刻实时要求的、复杂的高速硬线运算逻辑是一件很有挑战性的工作，即使有了好的算法而没有好的设计工具和方法也很难完成。

近30年来，我国在复杂数字电路设计技术领域与国外的差距越来越大。

作为一名在大学讲授专用数字电路与系统设计课程的老师深深感到责任的重大。

我个人认为，我国在这一技术领域的落后与大学的课程设置和教学条件有关。

因为我们没有及时把国外最先进的设计方法和技术介绍给学生，也没有给他们创造实践的机会。

1995年我受学校的委托，筹建世行贷款的电路设计自动化（EDA）实验室。

通过13年的摸索、实践，逐步掌握了利用VerilogHDL设计复杂数字电路的仿真和综合技术。

在此期间为航天部等有关单位设计了卫星信道加密用的复杂数字电路，提供给他们经前后仿真验证的VerilogHDL源代码，得到很高的评价。

在其后的几年中又为该单位设计了卫星下行信道RS（255，223）编码/解码电路和卫星上行信道BCH（64，56）编码/解码电路，这几个项目已先后通过有关单位的验收。

1999年到2000年期间，又成功地设计了用于小波（Wavelet）图像压缩/解压缩的小波卷积器和改进的零修剪树算法（即SPIHT算法）的RTL级VerilogHDL模型。

不但成功地对模型进行了仿真和综合，而且制成的可重新配置硬线逻辑（采用ALTERAFLEX10K系列CPLD/10/30/50各一片）的PCI线路板，能完成约2000条C语句程序才能完成的图像/解压缩算法。

运算结果与软件完成的完全一致，而且速度比用微型计算机快得多。

2003年由我协助指导的JPEG2000算法硬线逻辑设计，在清华同行的努力下完成了FPGA验证后并成功地投片，该芯片目前已应用于实时监控系統。

近年来我带领的研究生分别为日本某公司、香港科技大学电子系、革新科技公司和神州龙芯集成电路设计公司完成多项设计，其中包括SATA接口、AMBA总线接口、LED控制器和USB控制器等在内的多项IP设计，取得了良好的社会效益和声誉。

2006年秋，正式受聘于神州龙芯集成电路设计公司担任技术顾问，为中国的龙芯二号等IC事业提供必要的技术支持。

本书是在1998年北京航空航天大学出版社出版的《复杂数字电路与系统的VerilogHDL设计技术》和2003年《Verilog数字系统设计教程》基础上修订的，是一本既有理论又有实践的设计大全。

由于教学、科研、技术资料翻译和实验室的各项工作很忙，只能利用零碎时间，一点一滴地把积累的教学经验和新收集到的材料补充输入到计算机中，抽空加以整理。

我们使用Verilog设计复杂数字逻辑电路虽然已经有13年的时间，但仍旧在不断地学习提高之中，书中难免存在疏忽、错误之处，敬请细心的读者不吝指教。

我之所以在原版基础上把这本书再版，是想把原教材中一些不足的地方作一些必要的补充和修改，在大学生和研究生中加快Verilog设计技术的推广，尽快培养一批掌握先进设计技术的跨世纪的人才。

期望本书能在这一过程中起到抛砖引玉的作用。

回想起来，这本书实质上是我们实验室全体老师和同学们多年的劳动成果，其中在EDA实验室工

<<Verilog数字系统设计教程>>

作过的历届研究生张琰、山岗、王静璇、田玉文、冯文楠、杨柳、傅红军、龚剑、王书龙、胡瑛、杨雷、邢伟、管丽、刘曦、王进磊、王煜华、苏宇、张云帆、杨鑫、徐伟俊、邢小地、霍强、宋成伟、邢志成、李鹏、李琪、陈岩、赵宗民等都帮我做了许多工作，如部分素材的翻译、整理、录入和一些VerilogHDL模块的设计修改和验证。

本次再版是在本书第8次印刷之后受北航出版社之托进行的，趁此机会让我衷心地感谢在编写本书过程中所有给过我帮助和鼓励的老师和同学们。

教学中使用的多媒体课件已在本书第3次印刷后交给出版社，有需要者可向北航出版社发行部索取，可以免费提供给有关教师指导教学和备课演示之用。

<<Verilog数字系统设计教程>>

内容概要

本书讲述了自20世纪90年代开始在美国和其他先进的工业化国家逐步推广的利用硬件描述语言（Verilog HDL）建模、仿真和综合的设计复杂数字逻辑电路与系统的方法和技术。书中内容从算法和计算的基本概念出发，讲述如何由硬线逻辑电路来实现复杂数字逻辑系统的方法。全书共分4部分。

第一部分共8章，即Verilog数字设计基础篇，可作为本科生的入门教材。

第二部分共10章，即设计和验证篇，可作为本科高年级学生或研究生学习数字系统设计的参考书。

第三部分为实践篇，共提供12个上机练习和实验范例。

第四部分是语法篇，即Verilog 硬件描述语言参考手册；IEEE Verilog13642001标准简介，以反映Verilog语法的最新变化，可供读者学习、查询之用。

本书的教学方式以每2学时讲授一章为宜，每次课后需要花10 h复习思考。

完成10章学习后，就可以开始做上机练习，由简单到复杂，由典型到一般，循序渐进地学习Verilog HDL基础知识。

按照书上的步骤，可以使大学电子类及计算机工程类本科及研究生，以及相关领域的设计工程人员在半年内掌握Verilog HDL设计技术。

本书可作为电子工程类、自动控制类、计算机类的大学本科高年级及研究生教学用书，亦可供其他工程人员自学与参考。

<<Verilog数字系统设计教程>>

作者简介

夏宇闻，北京航空航天大学EDA实验室教授。

<<Verilog数字系统设计教程>>

书籍目录

第一部分 Verilog数字设计基础	第1章 Verilog的基本知识	1.1 硬件描述语言HDL	1.2 Verilog HDL的历史
	1.2.1 什么是Verilog HDL	1.2.2 Verilog HDL的产生及发展	1.3 Verilog HDL和VHDL的比较
	1.4 Verilog的应用情况和适用的设计	1.5 采用Verilog HDL设计复杂数字电路的优点	1.5.1 传统设计方法——电路原理图输入法
	1.5.2 Verilog HDL设计法与传统的电路原理图输入法的比较	1.5.3 Verilog的标准化与软核的重用	1.5.4 软核、固核和硬核的概念及其重用
	1.6 采用硬件描述语言(Verilog HDL)的设计流程简介	1.6.1 自顶向下(Top_Down)设计的基本概念	1.6.2 层次管理的基本概念
	1.6.3 具体模块的设计编译和仿真的过程	1.6.4 具体工艺器件的优化、映像和布局布线	小结
	思考题	第2章 Verilog语法的基本概念	概述
	2.1 Verilog模块的基本概念	2.2 Verilog用于模块的测试	小结
	思考题	第3章 模块的结构、数据类型、变量和基本运算符	概述
	3.1 模块的结构	3.1.1 模块的端口定义	3.1.2 模块内容
	3.1.3 理解要点	3.1.4 要点总结	3.2 数据类型及其常量和变量
	3.2.1 常量	3.2.2 变量	3.3 运算符及表达式
	3.3.1 基本的算术运算符	3.3.2 位运算符	小结
	思考题	第4章 运算符、赋值语句和结构说明语句	概述
	4.1 逻辑运算符	4.2 关系运算符	4.3 等式运算符
	4.4 移位运算符	4.5 位拼接运算符	4.6 缩减运算符
	4.7 优先级别	4.8 关键词	4.9 赋值语句和块语句
	4.9.1 赋值语句	4.9.2 块语句	小结
	思考题	第5章 条件语句、循环语句、块语句与生成语句	概述
	5.1 条件语句 (if_else语句)	5.2 case语句	5.3 条件语句的语法
	5.4 多路分支语句	5.5 循环语句	5.5.1 forever语句
	5.5.2 repeat语句	5.5.3 while语句	5.5.4 for语句
	5.6 顺序块和并行块	5.6.1 块语句的类型	5.6.2 块语句的特点
	5.7 生成块	5.7.1 循环生成语句	5.7.2 条件生成语句
	5.7.3 case生成语句	5.8 举例	5.8.1 四选一多路选择器
	5.8.2 四位计数器	小结	思考题
	第6章 结构语句、系统任务、函数语句和显示系统任务	概述	6.1 结构说明语句
	6.1.1 initial语句	6.1.2 always语句	6.2 task和function说明语句
	6.2.1 task和function说明语句的不同点	6.2.2 task说明语句	6.2.3 function说明语句
	6.2.4 函数的使用举例	6.2.5 自动(递归)函数	6.2.6 常量函数
	6.2.7 带符号函数	6.3 关于使用任务和函数的小结	6.4 常用的系统任务
	6.4.1 \$ display和\$ write任务	6.4.2 文件输出	6.4.3 显示层次
	6.4.4 选通显示	6.4.5 值变转储文件	6.5 其他系统函数和任务
	小结	思考题	第7章 调试用系统任务和常用编译预处理语句
	概述	7.1 系统任务 \$ monitor	7.2 时间度量系统函数 \$ time
	7.3 系统任务 \$ finish	7.4 系统任务 \$ stop	7.5 系统任务 \$ readmemb和 \$ readmemh
	7.6 系统任务 \$ random	7.7 编译预处理	7.7.1 宏定义?define
	7.7.2 文件包含"处理?include	7.7.3 时间尺度?timescale	7.7.4 条件编译命令?ifdef、?else、?endif
	7.7.5 条件执行	小结	思考题
	第8章 语法概念总复习练习	概述	小结
	第二部分 设计和验证部分	第9章 Verilog HDL模型的不同抽象级别	概述
	9.1 门级结构描述	9.1.1 与非门、或门和反向器及其说明语法	9.1.2 用门级结构描述D触发器
	9.1.3 由已经设计成的模块构成更高层的模块	9.2 Verilog HDL的行为描述建模	9.2.1 仅用于产生仿真测试信号的Verilog HDL行为描述建模
	9.2.2 Verilog HDL建模在TopDown设计中的作用和行为建模的可综合性问题	9.3 用户定义的原语	小结
	思考题	第10章 如何编写和验证简单的纯组合逻辑模块	概述
	10.1 加法器	10.2 乘法器	10.3 比较器
	10.4 多路器	10.5 总线和总线操作	10.6 流水线
	小结	思考题	第11章 复杂数字系统的构成
	概述	11.1 运算部件和数据流动的控制逻辑	11.1.1 数字逻辑电路的种类
	11.1.2 数字逻辑电路的构成	11.2 数据在寄存器中的暂时保存	11.3 数据流动的控制
	11.4 在Verilog HDL设计中启用同步时序逻辑	11.5 数据接口的同步方法	小结
	思考题	第12章 同步状态机的原理、结构和设计	概述
	12.1 状态机的结构	12.2 Mealy状态机和Moore状态机的不同点	12.3 如何用Verilog来描述可综合的状态机
	12.3.1 用可综合Verilog模块设计状态机的典型办法	12.3.2 用可综合的Verilog模块设计、用独热码表示状态的状态机	12.3.3 用可综合的Verilog模块设计、由输出指定的码表示状态的状态机
	12.3.4 用可综合的Verilog模块设计复杂的多输出状态机时常用的方法	小结	思考题
	第13章 设计可综合的状态机的指导原则	概述	13.1 用Verilog HDL语言设计可综合的状态机的指导原则
	13.2 典型的Verilog HDL实例	13.3 综合的一般原则	13.4 语言指导原则
	13.5 可综合风格的Verilog HDL		

<<Verilog数字系统设计教程>>

模块实例 13.5.1 组合逻辑电路设计实例 13.5.2 时序逻辑电路设计实例 13.6 状态机的置位与复位
 13.6.1 状态机的异步置位与复位 13.6.2 状态机的同步置位与复位 小结 思考题 第14章 深入理解
 阻塞和非阻塞赋值的不同 概述 14.1 阻塞和非阻塞赋值的异同 14.1.1 阻塞赋值 14.1.2 非阻塞
 赋值 14.2 Verilog模块编程要点 14.3 Verilog的层次化事件队列 14.4 自触发always块 14.5 移
 位寄存器模型 14.6 阻塞赋值及一些简单的例子 14.7 时序反馈移位寄存器建模 14.8 组合逻
 辑建模时应使用阻塞赋值 14.9 时序和组合的混合逻辑——使用非阻塞赋值 14.10 其他阻塞和非
 阻塞混合使用的原则 14.11 对同一变量进行多次赋值 14.12 常见的对于非阻塞赋值的误解
 小结 思考题 第15章 较复杂时序逻辑电路设计实践 概述 小结 思考题 第16章 复杂时
 序逻辑电路设计实践 概述 16.1 二线制I2C CMOS串行EEPROM的简单介绍 16.2 I2C总线特征介
 绍 16.3 二线制I2C CMOS串行EEPROM的读写操作 16.4 EEPROM的Verilog HDL程序 总结 思
 考题 第17章 简化的RISC_CPU设计 概述 17.1 课题的来由和设计环境介绍 17.2 什么是CPU
 17.3 RISC_CPU结构 17.3.1 时钟发生器 17.3.2 指令寄存器 17.3.3 累加器 17.3.4 算术运算器
 17.3.5 数据控制器 17.3.6 地址多路器 17.3.7 程序计数器 17.3.8 状态控制器 17.3.9 外围模块
 17.4 RISC_CPU操作和时序 17.4.1 系统的复位和启动操作 17.4.2 总线读操作 17.4.3 总线写操作
 17.5 RISC_CPU寻址方式和指令系统 17.6 RISC_CPU模块的调试 17.6.1 RISC_CPU模块的前仿真
 17.6.2 RISC_CPU模块的综合 17.6.3 RISC_CPU模块的优化和布局布线 小结 思考题 第18章
 虚拟器件/接口、IP和基于平台的设计方法及其在大型数字系统设计中的作用 概述 18.1 软核和
 硬核、宏单元、虚拟器件、设计和验证IP以及基于平台的设计方法 18.2 设计和验证IP供应商
 18.3 虚拟模块的设计 18.4 虚拟接口模块的实例 小结 思考题第三部分 设计示范与实验练
 习 概述 练习一 简单的组合逻辑设计 练习二 简单分频时序逻辑电路的设计 练习三 利用条件语
 句实现计数分频时序电路 练习四 阻塞赋值与非阻塞赋值的区别 练习五 用always块实现较复杂的组
 合逻辑电路 练习六 在Verilog HDL中使用函数 练习七 在Verilog HDL中使用任务(task) 练习八
 利用有限状态机进行时序逻辑的设计 练习九 利用状态机实现比较复杂的接口设计 练习十 通过模
 块实例调用实现大型系统的设计 练习十一 简单卷积器的设计 附录一 A/D转换器的Verilog HDL模
 型机所需要的技术参数 附录二 2K*8位 异步 CMOS 静态RAM HM65162模型 练习十二 利用SRAM
 设计一个FIFO第四部分 语法篇 语法篇1 关于Verilog HDL的说明 一、关于IEEE 1364标准 二
 、 Verilog简介 三、语法总结 四、编写Verilog HDL源代码的标准 五、设计流程 语法
 篇2 Verilog硬件描述语言参考手册 一、 Verilog HDL语句与常用标志符(按字母顺序排列) 二
 、系统任务和函数(System task and function) 三、常用系统任务和函数的详细使用说明 四、
 Command Line Options 命令行的可选项 五、IEEE Verilog 13642001标准简介参考文献

<<Verilog数字系统设计教程>>

章节摘录

第一部分 Verilog数字设计基础 数字通信和自动化控制等领域的高速发展和世界范围内的高技术竞争对数字系统提出了越来越高的要求,特别是需要设计具有实时信号处理能力的专用集成电路,要求把包括多个CPU内核在内的整个电子系统综合到一个芯片(SOC)上。设计并验证这样复杂的电路及系统已不再是简单的个人劳动,而需要综合许多专家的经验 and 知识才能够完成。

近10年来电路制造工艺技术进步非常迅速,目前国际上60 nm的制造工艺,已达到工业化生产的规模,而电路设计能力远远落后于制造技术的进步。

在数字逻辑设计领域,迫切需要一种共同的工业标准来统一对数字逻辑电路及系统的描述,这样就能把系统设计工作分解为逻辑设计(前端)、电路实现(后端)和验证三个互相独立而又相关的部分。

由于逻辑设计的相对独立性就可以把专家们设计的各种常用数字逻辑电路和组件(如FFT算法、DCT算法部件,DDRAM读写控制器等)建成宏单元(megcell)或软(固/硬)核,也称作Soft(firm/hard)Core,即IP(知识产权内核的英文缩写)库供设计者引用,设计者可以直接利用它们的行为模型设计并验证其他电路,以减少重复劳动,提高工作效率。

电路的实现则可借助于综合工具和IP的重复利用,以及布局布线工具(与具体工艺技术有关)自动地完成。

Verilog HDL和VHDL这两种工业标准的产生顺应了历史的潮流,因而得到了迅速的发展。

美国、日本等国由于高级设计工程师人力资源成本远高于中国,所以,近年来把许多设计工作转移到中国大陆,以降低设计成本。

作为新世纪的中国大学生和年轻的电子工程师应该尽早掌握这种新的设计方法,使我国在复杂数字电路及系统的设计竞争中逐步缩小与美国等先进的工业发达国家的差距。

第1章 Verilog的基本知识 1.1 硬件描述语言HDL 硬件描述语言(HDL, hardware description language)是一种用形式化方法来描述数字电路和系统的语言。数字电路系统的设计者利用这种语言可以从上层到下层(从抽象到具体)逐层描述自己的设计思想,用一系列分层次的模块来表示极其复杂的数字系统。

然后利用电子设计自动化(以下简称为EDA)工具逐层进行仿真验证,再把其中需要变为具体物理电路的模块组合经由自动综合工具转换到门级电路网表。

接下去再用专用集成电路(ASIC)或现场可编程门阵列(FPGA)自动布局布线工具把网表转换为具体电路布线结构的实现。

在制成物理器件之前,还可以用Verilog的门级模型(原语元件或UDP)来代替具体基本元件。

因其逻辑功能和延时特性与真实的物理元件完全一致,所以在仿真工具的支持下能验证复杂数字系统物理结构的正确性,使投片的成功率达到100%。

目前,这种称为高层次设计(high-level-design)的方法已被广泛采用。

据统计,目前在美国硅谷约有90%以上的ASIC和FPGA已采用Verilog硬件描述语言方法进行设计。

硬件描述语言的发展至今已有近30年的历史,并成功地应用于设计的各个阶段:建模、仿真、验证和综合等。

到20世纪80年代,已出现了上百种硬件描述语言,并对设计自动化曾起到了极大的促进和推动作用。

但是,这些语言一般各自面向特定的设计领域与层次,而且众多的语言使用户无所适从。

因此急需一种面向设计的多领域、多层次、并得到普遍认同的标准硬件描述语言。

进入20世纪80年代后期,硬件描述语言向着标准化的方向发展。

最终,VHDL和Verilog HDL语言适应了这种趋势的要求,先后成为IEEE标准。

把硬件描述语言用于自动综合还只有10多年的历史。

最近10多年来,用综合工具把可综合风格的HDL模块自动转换为具体电路发展非常迅速,大大地提高了复杂数字系统的设计生产率。

在美国和日本等先进电子工业国,Verilog语言已成为设计数字系统的基础。

本书第一部分将通过具体例子,由浅入深地帮助同学们学习: (1) Verilog的基本语法; (2)

<<Verilog数字系统设计教程>>

)简单的可综合Verilog模块与逻辑电路的对应关系； (3)简单的Verilog测试模块和它的意义。

书中第二部分将通过较复杂的设计实例,帮助同学们掌握: (1)如何编写复杂的多层次的可综合风格的Verilog HDL模块; (2)如何用可综合的Verilog模块构成一个可靠的复杂IP软核和固核模块; (3)如何借助于Verilog语言,并利用已有的虚拟行为模块对所设计的系统模块(由可综合的自主和商业IP模块组成)进行全面可靠的测试和验证(包括软/硬件协同测试的基本概念)。

1.2 Verilog HDL的历史 1.2.1什么是Verilog HDL Verilog HDL是硬件描述语言的一种,用于数字电子系统设计。

该语言允许设计者进行各种级别的逻辑设计,进行数字逻辑系统的仿真验证、时序分析、逻辑综合。它是目前应用最广泛的一种硬件描述语言。

据有关文献报道,目前在美国使用Verilog HDL进行设计的工程师大约有10多万人,全美国有200多所大学教授用Verilog HDL硬件描述语言的设计方法。

在我国台湾地区几乎所有著名大学的电子和计算机工程系都讲授Verilog有关的课程。

1.2.2 Verilog HDL的产生及发展 Verilog HDL是在1983年由GDA (GateWay Design Automation)公司的Phil Moorby首创的。

Phil Moorby后来成为Verilog—XL的主要设计者和Cadence公司 (Cadence Design System) 的第一个合伙人。

在1984至1985年, Moorby设计出了第一个名为Verilog—XL的仿真器;1986年,他对Verilog HDL的发展又作出了另一个巨大贡献,即提出了用于快速门级仿真的XL算法。

随着Verilog—XL算法的成功, Verilog HDL语言得到迅速发展。

1989年, Cadence公司收购了GDA公司, Verilog HDL语言成为Cadence公司的私有财产。

1990年, Cadence公司决定公开Verilog HDL语言,于是成立了OVI (Open Verilog International) 组织来负责促进Verilog HDL语言的发展。

基于Verilog HDL的优越性, IEEE于1995年制定了Verilog HDL的IEEE标准,即Verilog HDL1364—1995;2001年发布了Verilog HDL1364—2001标准;2005年SystemVerilog IEEE 1800—2005标准的公布,更使得Verilog语言在综合、仿真验证和模块的重用等性能方面都有大幅度的提高。

图1.1展示了Verilog的发展历史和未来。

1.3 Verilog HDL和VHDL的比较 Verilog HDL和VHDL都是用于逻辑设计的硬件描述语言,并且都已成为IEEE标准。

VHDL是在1987年成为IEEE标准, Verilog HDL则在1995年才正式成为IEEE标准。

之所以VHDL比Verilog HDL早成为IEEE标准,这是因为VHDL是由美国军方组织开发的,而Verilog HDL则是从一个普通的民间公司的私有财产转化而来,基于Verilog HDL的优越性,才成为IEEE标准,因而有更强的生命力。

VHDL其英文全名为VHSIC Hardware Description Language,而VHSIC则是Very High Speed Integrated Circuit的缩写词,意为甚高速集成电路,故VHDL其准确的中文译名为甚高速集成电路的硬件描述语言。

Verilog HDL和VHDL作为描述硬件电路设计的语言,其共同的特点在于:能形式化地抽象表示电路的行为和结构;支持逻辑设计中层次与范围的描述;可借用高级语言的精巧结构来简化电路行为的描述;具有电路仿真与验证机制以保证设计的正确性;支持电路描述由高层到低层的综合转换;硬件描述与实现工艺无关(有关工艺参数可通过语言提供的属性包括进去);便于文档管理;易于理解和设计重用。

但是Verilog HDL和VHDL又各有其自己的特点。

由于Verilog HDL早在1983年就已推出,至今已有20多年的应用历史,因而Verilog HDL拥有更广泛的设计群体,成熟的资源也远比VHDL丰富。

与VHDL相比Verilog HDL的最大优点是:它是一种非常容易掌握的硬件描述语言,只要有C语言的编程基础,通过20学时的学习,再加上一段实际操作,一般同学可在2~3个月内掌握这种设计方法的基本技术。

而掌握VHDL设计技术就比较困难。

<<Verilog数字系统设计教程>>

这是因为VHDL不很直观，需要有Ada编程基础，一般认为至少需要半年以上的专业培训，才能掌握VHDL的基本设计技术。

2005年，SystemVerilog IEEE1800—2005标准公布以后，集成电路设计界普遍认为Verilog HDL将在10年内全面取代VHDL成为ASIC设计行业包揽设计、测试和验证功能的唯一语言。

图1.2所示的是Verilog HDL和VHDL建模能力的比较图，供读者参考。

-2005标准，不但使Verilog的可综合性能和系统仿真性能方面有大幅度的提高，而且在IP的重用方面（包括设计和验证模块的重用）也有重大的突破。

因此，Verilog HDL不但作为学习HDL设计方法的入门和基础是比较合适的，而且对于ASIC设计专业人员而言，也是必须掌握的基本技术。

学习掌握Verilog HDL建模、仿真、综合、重用和验证技术不仅可以使同学们对数字电路设计技术有更进一步的了解，而且可以为以后学习高级的行为综合、物理综合、IP设计和复杂系统设计和验证打下坚实的基础。

1.4 Verilog的应用情况和适用的设计 近10多年以来，EDA界一直对在数字逻辑设计中究竟采用哪一硬件描述语言争论不休。

近二三年来，美国、日本和我国台湾地区电子设计界的情况已经清楚地表明，在高层次数字系统设计领域，Verilog已经取得压倒性的优势；

<<Verilog数字系统设计教程>>

编辑推荐

《普通高等教育"十一五"国家级规划教材?北京高等教育精品教材?Verilog数字系统设计教程(第2版)》可作为电子工程类、自动控制类、计算机类的大学本科高年级及研究生教学用书,亦可供其他工程人员自学与参考。

《Verilog数字系统设计教程》是在1998年北京航空航天大学出版社出版的《复杂数字电路与系统的Verilog HDL设计技术》和2003年《Verilog数字系统设计教程》基础上修订的,是一本既有理论又有实践的设计大全。

<<Verilog数字系统设计教程>>

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>