

国外电子与通信教材系列

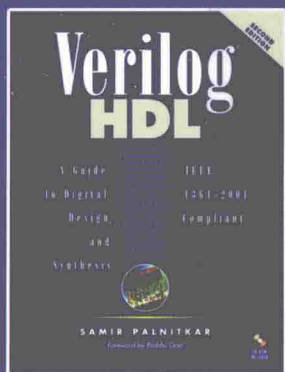
符合 Verilog HDL IEEE 1364-2001 标准

PEARSON

Verilog HDL 数字设计与综合 (第二版) (本科教学版)

Verilog HDL: A Guide to Digital Design and Synthesis

Second Edition



[美] Samir Palnitkar 著

夏宇闻 胡燕祥 刁岚松 等译

夏宇闻 审校



中国工信出版集团



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 数字设计与综合

(第二版) (本科教学版)

Verilog HDL
A Guide to Digital Design and Synthesis
Second Edition

[美] Samir Palnitkar 著

夏宇闻 胡燕祥 刁岚松 等译

夏宇闻 审校

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书从用户的角度全面阐述了Verilog HDL语言的重要细节和基本设计方法,并详细介绍了Verilog 2001版的主要改进部分。本书重点关注如何应用Verilog语言进行数字电路和系统的设计和验证,而不仅仅讲解语法。全书从基本概念讲起,并逐渐过渡到编程语言接口以及逻辑综合等高级主题。书中的内容全部符合Verilog HDL IEEE 1364-2001标准。

本书适合电子、计算机、自动控制等专业的学习数字电路设计的大学本科高年级学生阅读,也适合数字系统设计工程师和已具有多年Verilog设计工作经验的资深工程师参考。

Authorized translation from the English language edition, entitled VERILOG HDL: A GUIDE TO DIGITAL DESIGN AND SYNTHESIS, SECOND EDITION, 9780130449115 by Samir Palnitkar, published by Pearson Education, Inc., Publishing as Prentice Hall PTR, Copyright © 2003 by Sun Microsystems, 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.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY Copyright © 2015.

本书中文简体字版专有出版权由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。

本书封面贴有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:01-2003-1047

图书在版编目(CIP)数据

Verilog HDL数字设计与综合:第2版:本科教学版/(美)帕尔尼卡(Palnitkar, S.)著;夏宇闻等译.

北京:电子工业出版社,2015.8

国外电子与通信教材系列

书名原文:Verilog HDL: A Guide to Digital Design and Synthesis, Second Edition

ISBN 978-7-121-26124-4

I. ①V… II. ①帕… ②夏… III. ①硬件描述语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2015)第110144号

策划编辑:马 岚

责任编辑:马 岚

印 刷:北京季蜂印刷有限公司

装 订:北京季蜂印刷有限公司

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:787×1092 1/16 印张:19.75 字数:506千字

版 次:2015年8月第1版(原著第2版)

印 次:2015年8月第1次印刷

定 价:49.00元

凡所购买电子工业出版社图书有缺损问题,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zlbs@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010)88258888。

作者简介

Samir Palnitkar 是美国 Jambo Systems 公司总裁。Jambo Systems 公司是一流的专用集成电路 (ASIC) 设计和验证服务公司, 专门从事高级微处理器、网络和通信芯片的设计服务。Palnitkar 先生曾创办一系列小型的高科技公司。他是 Integrated Intellectual Property 公司的创办人。该公司是一家专用集成电路设计公司, 已被 Lattice Semiconductor 公司收购。后来, 他创建了电子商务软件公司 Obongo, 已被 AOL Time Warner 公司收购。

Palnitkar 先生毕业于位于印度坎普尔市的印度理工学院电气工程系, 获得学士学位, 后来在美国西雅图的华盛顿大学电气工程系获得硕士学位, 接着在美国加州圣何塞州立大学获得 MBA 学位。

Palnitkar 先生目前是数字系统设计领域 Verilog HDL 建模、逻辑综合和基于 EDA 的设计方法学等方面的公认权威。他在设计和验证方面有丰富的工作经验, 成功地完成过多种微处理器、专用集成电路和系统的设计。他是第一个使用 Verilog 语言为共享内存、高速缓冲存储器组合 (cache coherent) 和多处理器体系结构搭建框架的人物。他领导研发了多处理器体系结构 (一般称为 UltraSPARC 端口体系结构)。Sun Microsystems 公司在下一代台式机的设计中采用了他研发的这种体系结构。除了 UltraSPARC CPU 以外, 他还为许多一流的公司完成过许多不同类型的设计和验证项目。

Palnitkar 先生与一些研发仿真产品的公司有合作关系, 是首批试用基于周期仿真的技术的领军人物。他有使用多种 EDA 工具的经验, 诸如 Verilog-NC, Synopsys VCS, Specman, Vera, System Verilog, Synopsys, SystemC, Verplex 和 Design Data Management Systems 等。

Palnitkar 先生是三项美国专利的发明人。第一项是分析有限状态机的新方法; 第二项是基于周期的仿真技术; 第三项是独特的电子商务技术 (待批准)。他还发表了几篇技术论文。在业余时间, Palnitkar 先生喜欢板球运动、阅读书籍和环球旅行。

序

早在 1984 年, Gateway Design Automation 公司就谨慎地开始了 Verilog 硬件描述语言的研发。这种语言得到了集成电路芯片和数字系统设计工程师的广泛认可和普遍采用, 因此已经成为了一项工业标准。Verilog 最初是一种靠仿真环境支持的专利语言, 是第一种能够支持混合层次 (mixed-level) 设计表达方式的语言。这些层次包括数字电路的各种级别的抽象, 即从开关级、门级、RTL 级一直到更高级别的抽象。仿真环境提供了功能强大的统一的方法, 不但能用于数字系统的设计, 还能进行数字系统的测试, 即对正在进行的数字系统设计进行验证。

Verilog 之所以能在市场上得到认可并占据主导地位, 有三个关键的因素。第一个关键因素是, 在 Verilog 语言中引入了编程语言接口 (PLI)。利用 PLI, Verilog 用户可以扩展具有自己的特色的仿真环境。如果用户明白了如何开发 PLI, 并成功地采用 Verilog 扩展了自己的仿真环境, 那么这些用户就能成为真正的 Verilog 赢家。第二个关键因素是, Gateway 公司一直密切注意 ASIC 制造厂商的需求。从 1987 年到 1989 年期间, 公司曾努力与 Motorola, National 和 UTMC 等 ASIC 制造厂商在 Verilog 应用和开发方面加强合作, 这些工作使得 Verilog 在这一领域逐渐占据了主导地位。Gateway 公司认识到, 绝大多数的数字逻辑仿真工作是由 ASIC 芯片的设计者完成的, 这一认识增加了 Verilog 取得成功的机会。随着 ASIC 制造厂商提倡使用 Verilog, Verilog 仿真器逐渐被 ASIC 制造厂商认可, 作为接收设计制造订单时的签字认可测试工具^①。工业界对 Verilog 的认可, 更进一步使得它在数字逻辑设计领域占据统治地位。最后一个关键因素是, 1987 年 Synopsys 公司引入了以 Verilog 为基础的综合技术, 从而有力地支持了 Verilog 取得成功。Gateway 公司为了让 Verilog 在综合技术方面取得优势, 把其专有的 Verilog 使用权授予了 Synopsys 公司, 仿真和综合技术的结合使得 Verilog 成为硬件设计工程师首选的硬件描述语言。

VHDL (VHSIC Hardware Description Language, 甚高速集成电路硬件描述语言) 的出现, 得到了许多其他 EDA 厂商的强力追捧, 使得 VHDL 很快成为 IEEE 标准, 并因此使 Verilog 在许多公开的场合受到过“排挤”。1995 年, Verilog 也被批准成为 IEEE 1364 标准。并且, 自从 1995 年以来, 根据 Verilog 用户提出的需求, Verilog 做了许多增补。这些增补都已经归入最新推出的 Verilog 标准, IEEE 1364-2001。今天, Verilog 已经成为数字设计的首选语言, 它是综合、验证和布局布线技术的基础。

本书是 Verilog 语言用户的最好的指南, 不但用丰富的实例解释了该语言的许多结构, 还深入到许多应用细节, 如编程语言接口 (PLI) 的开发使用和综合技术等。本书章节安排合理, 循序渐进, 从实际设计工作者的需求的角度来编写, 而不只是讲解一些语法现象。

本书第二版在两个方面具有特色。第一, 包括了 IEEE 1364-2001 标准中所有新增补的内容, 让读者有机会了解有关 Verilog 的最新信息, 提高自己的设计水平。第二, 新增加了一章, 专门介绍高级验证技术, 这些高级验证技术现在已经成为 Verilog 设计方法学整体结构的一部分。对设计

^① 这里指的是业界认可的测试工具, 用于对设计方的投片委托进行严格的验证, 并由制造方签字认可该设计是正确无误的。如果制造出的芯片不合格, 一切损失将由制造方承担。——译者注

和验证由几百万个门构成的数字系统的 Verilog 用户来说，了解和掌握这些技术是极其关键的。

给用户讲解 Verilog 及其各种设计和验证方法学是一件相当困难并具有挑战意义的工作。手头有这本书，一定能使 Verilog 语言的初学者大大加快学习过程，也能使有经验的 Verilog 老用户很方便地获取 Verilog 新知识。这是一本每个 Verilog 用户手头必备的好书。

Prabhu Goel
Gateway Design Automation 公司前任总裁

前 言

刚开始学习 Verilog HDL 时，我到处寻找能帮助我很快进入 Verilog HDL 用户角色的书。我想参考一些基础数字电路设计范例来学习必要的 Verilog HDL 结构，这样就能自己用 Verilog 语言来设计一些小规模的数字电路并运行仿真。后来，我逐渐积累了一些编写基本 Verilog 模块的经验，并想进一步学习用 Verilog HDL 来设计更复杂一些的电路。当时，我到处打听，希望能搜寻到一本好书，希望能通过阅读这本书而全面了解高级 Verilog 数字系统设计概念，并掌握实际数字电路设计方法学。但是，我一直没有找到合适的书。我是通过经常接触到的 Verilog 产品手册，慢慢地积累了数字电路设计和实际集成电路验证的经验。如果当时手头有一本好的 Verilog 参考书，可以随时翻阅，我的学习进度一定可以大大加快。这个想法促使我编写了本书的第一版。

自从本书第一版发行以来，已经过了 6 年多。在此期间，我完成了各种类型的许多 ASIC 设计和微处理器设计项目，我的设计和验证经验也随之逐渐加深和丰富。同时我也注意到，反映技术发展水平的高级验证方法学和各种设计工具也更加成熟。Verilog HDL 的新标准 IEEE 1364-2001 也已经得到批准。编写本书第二版的目的是增补有关 IEEE 1364-2001 标准的新内容，把最先进的验证方法介绍给 Verilog 用户，并希望新版能为读者提供更加丰富的学习内容。

本书重点放在内容的宽广度而不是深度上，旨在传授给读者基于 Verilog 的广泛的实用工作知识，使读者能够对基于 Verilog HDL 的设计技术有全面的了解。本书把深入涵盖每个 Verilog 主题的任务留给 Verilog HDL 语言参考手册和各种基于 Verilog 产品的参考手册。

本书虽然应该归属于 Verilog HDL 语法类书籍，但总的来说更应该归属于数字设计类书籍。Verilog HDL 只是一种设计数字电路和系统的工具，认识到这一点是很重要的。Verilog HDL 只是实现我们的最终目的（即数字集成电路芯片）的手段，因此本书强调设计实践的全面知识，而不仅仅只介绍 Verilog HDL 语言方面的知识。基于硬件描述语言的数字电路和系统的设计方法已经逐渐成为工程师们必须掌握的方法，没人能承受无视硬件描述语言所造成的工作损失。

读者对象

本书面向的主要对象是 Verilog 的初学者和中级水平的读者。然而，对于 Verilog 高级用户来说，本书涵盖的知识面很广，为这些高级读者提供了一本极好的参考书，能帮助他们理解各种手册和基于 Verilog 产品的培训教材。

关于 Verilog 硬件描述语言的要点，在本书中安排合理、循序渐进。本书从最基本的概念出发，例如，从介绍基于硬件描述语言的设计方法学着手，然后逐渐过渡到高级主题，例如编程语言接口和逻辑综合等。因此本书对专业水平不同的读者都有用，具体解释如下：

- **学习数字逻辑设计的大学生**

本书的第一部分用做 Verilog HDL 逻辑设计课程教学是很合适的。学生们受到书申层次建模的概念、基本 Verilog 结构、建模等技术的熏陶，很快就可以学会编写小型模块和运行仿真的知识。

- **业界的 Verilog 新用户**

许多公司正在转向用 Verilog 进行设计。本书的第一部分对于想把自己的技术转向基于 HDL

设计的设计师们来说，是完美的入门课程。

- **已经具有 Verilog 基础知识并希望理解高级概念的用户**

本书的第二部分讨论了许多高级概念，诸如用户自定义原语（UDP）、时序仿真、编程语言接口（PLI）和逻辑综合等，这些知识对于正从编写小型 Verilog 模型过渡到较大设计的研究生来说是必需的。

- **Verilog 专家**

本书涵盖了所有有关 Verilog 的主题，从基本的建模结构一直到高级主题，诸如编程语言接口（PLI）、逻辑综合和高级的验证技术等。对于 Verilog 专家来说，本书是一本唾手可得的参考书，与 IEEE Standard Verilog Hardware Description Language 参考手册相辅相成。

本书包含的材料倾向于专用集成电路（ASIC）的设计方法学，但是本书中所解释的概念完全适用于 FPGA，PAL，总线，线路板和系统的设计。本书为了简化讨论，采用中规模集成电路（MSI）作为例子，但这些概念完全适用于超大规模集成电路（VLSI）的设计。

组织结构

本书由如下三部分组成。

第一部分 Verilog 基础知识

这一部分涵盖了 Verilog 初学者编写小型 Verilog 模型和运行仿真所必须掌握的全部资料。注意，本书将门级建模的介绍放在在行为级建模之前。之所以这样做，是因为我认为对初学者来说理解门级电路与 Verilog 行为描述之间的一一对应关系是比较容易的。一旦理解了门级建模，初学者就能很容易地理解更高层次的抽象，诸如数据流建模和行为建模，不会把 Verilog HDL 只当成普通的编程语言，而忽略了它是数字设计语言的事实。这样，初学者在一开始就可以建立起 Verilog 是数字电路设计语言的概念。初学者如果一开始就学习行为描述，往往倾向于像编写 C 程序那样来编写 Verilog，有时候看不到自己正在试图用 Verilog 语言来表示硬件电路。第一部分共有 9 章。

第二部分 Verilog 高级主题

这一部分包含许多高级概念，对于想从编写小型 Verilog 模型过渡到较大设计的 Verilog 用户来说，这些概念是必须知道的。本部分覆盖的高级内容包括时序仿真、开关级建模、用户自定义原语（UDP）、编程语言接口（PLI）、逻辑综合和高级验证技术等。第二部分共有 6 章。

第三部分 附录

这一部分包含的内容可以作为参考资料使用，包括强度建模、编程语言接口（PLI）子程序清单、形式化语法定义、Verilog 有关问题解答和大型 Verilog 设计举例等。第三部分共有 6 个附录。

本书采用的约定

本书用黑体字表示属于 Verilog HDL 的一部分的关键字、系统任务和编译指令，例如 **and**，**nand**，**\$display** 和 **define**。另外，还有两个约定需要说明：

- 本书中凡是用到 Verilog 和 Verilog HDL，都指的是 Hardware Description Language，即 Verilog 硬件描述语言；凡是用到 Verilog 仿真器或仿真器产品的商标，如 Verilog-XL 或 VCS，都指的是基于 Verilog 的仿真器。
- 本书中经常提到的设计人员主要是指数字逻辑设计人员，然而更经常的情况是指 Verilog HDL 的使用者或验证工程师。

致 谢

本书的第一版是在许多人的帮助下完成的。为了本书的出版，他们付出了心血。下面列出了那些曾为我编写本书做出过主要贡献的人员：

John Sanguinetti, Stuart Sutherland, Clifford Cummings, Robert Emberley, Ashutosh Mauskar, Jack McKeown, Dr. Arun Somani, Dr. Michael Ciletti, Larry Ke, Sunil Sabat, Cheng-I Huang, Maqsoodul Mannan, Ashok Mehta, Dick Herlein, Rita Glover, Ming-Hwa Wang, Subramanian Ganesan, Sandeep Aggarwal, Albert Lau, Samir Sanghani, Kiran Buch, Anshuman Saha, Bill Fuchs, Babu Chilukuri, Ramana Kalapatapu, Karin Ellison 和 Rachel Borden。

作为本书第二版的致谢辞的开场白，我要再次感谢这些人！

对于本书的第二版，我要特别感谢下面这些人员，他们帮助我完成了审阅的过程，并提供了有价值的反馈意见：

Anders Nordstrom	ASIC Consultant
Stefen Boyd	Boyd Technology
Clifford Cummings	Sunburst Design
Harry Foster	Verplex System
Yatin Trivedi	Magma Design Automation
Rajeev Madhavan	Magma Design Automation
John Sanguinetti	Forte Design System
Dr. Arun Somani	Iowa State University
Michael McNamara	Verisity Design
Berend Ozceri	Cisco System
Shrenik Mehta	Sun Microsystem
Mike Meredith	Forte Design System

我还要感谢下面这些人：

Simucad 公司的 Richard Jones 和 John Williamson，感谢他们提供了免费的 Verilog 仿真器 SILOS 2001。本书附带了包含该仿真器软件的光盘^①。

Prentice Hall 出版公司的 Greg Doench 和 Sun Microsystem 公司的 Myrna Rivera，感谢他们在本书的出版过程中所给予的帮助。

本书第二版中的有些资料得益于业界同仁的谈话、电子邮件和建议。我已经在书中提及了资料来源，但如果我万一忽略了某人，请一定要接受我的歉意。

Samir Palnikar
写于加州硅谷

① 英文原书附带的光盘未包括在中译本中，而是直接以网站下载方式提供。光盘内容包括免费的 Verilog 仿真器 SILOS 2001 以及本书中所用例子的源代码。读者可以登录华信教育资源网 (<http://www.hxedu.com.cn>) 注册下载有关内容。——编者注

目 录

第一部分 Verilog 基础知识

第 1 章 Verilog HDL 数字设计综述	2
1.1 数字电路 CAD 技术的发展历史	2
1.2 硬件描述语言的出现	2
1.3 典型设计流程	3
1.4 硬件描述语言的意义	4
1.5 VERILOG HDL 的优点	5
1.6 硬件描述语言的发展趋势	5
第 2 章 层次建模的概念	7
2.1 设计方法学	7
2.2 四位脉动进位计数器	8
2.3 模块	9
2.4 模块实例	10
2.5 逻辑仿真的构成	12
2.6 举例	12
2.7 小结	15
2.8 习题	16
第 3 章 基本概念	17
3.1 词法约定	17
3.2 数据类型	20
3.3 系统任务和编译指令	25
3.4 小结	29
3.5 习题	30
第 4 章 模块和端口	31
4.1 模块	31
4.2 端口	33
4.3 层次命名	38
4.4 小结	39
4.5 习题	39

第 5 章 门级建模	40
5.1 门的类型	40
5.2 门延迟	50
5.3 小结	54
5.4 习题	55
第 6 章 数据流建模	56
6.1 连续赋值语句	56
6.2 延迟	58
6.3 表达式、操作符和操作数	59
6.4 操作符类型	60
6.5 举例	67
6.6 小结	74
6.7 习题	74
第 7 章 行为级建模	76
7.1 结构化过程语句	76
7.2 过程赋值语句	79
7.3 时序控制	83
7.4 条件语句	88
7.5 多路分支语句	89
7.6 循环语句	91
7.7 顺序块和并行块	94
7.8 生成块	98
7.9 举例	103
7.10 小结	108
7.11 习题	109
第 8 章 任务和函数	112
8.1 任务和函数的区别	112
8.2 任务	113
8.3 函数	117
8.4 小结	121
8.5 习题	122
第 9 章 实用建模技术	123
9.1 过程连续赋值	123
9.2 改写(覆盖)参数	125
9.3 条件编译和执行	127
9.4 时间尺度	130

9.5 常用的系统任务	131
9.6 小结	137
9.7 习题	138

第二部分 Verilog 高级主题

第 10 章 时序和延迟	142
10.1 延迟模型的类型	142
10.2 路径延迟建模	145
10.3 时序检查	151
10.4 延迟反标注	153
10.5 小结	154
10.6 习题	154
第 11 章 开关级建模	156
11.1 开关级建模元件	156
11.2 举例	160
11.3 小结	164
11.4 习题	165
第 12 章 用户自定义原语	166
12.1 UDP 的基础知识	166
12.2 表示组合逻辑的 UDP	168
12.3 表示时序逻辑的 UDP	173
12.4 UDP 表中的缩写符号	176
12.5 UDP 设计指南	177
12.6 小结	178
12.7 习题	178
第 13 章 编程语言接口	180
13.1 PLI 的使用	182
13.2 PLI 任务的连接和调用	182
13.3 内部数据表示	184
13.4 PLI 库子程序	185
13.5 小结	195
13.6 习题	196
第 14 章 使用 Verilog HDL 进行逻辑综合	197
14.1 什么是逻辑综合	197
14.2 逻辑综合对数字设计行业的影响	199

14.3	VERILOG HDL 综合	200
14.4	逻辑综合流程	204
14.5	门级网表的验证	210
14.6	逻辑综合建模技巧	212
14.7	时序电路综合举例	217
14.8	小结	224
14.9	习题	224
第 15 章 高级验证技术		226
15.1	传统的验证流程	226
15.2	断言检查	234
15.3	形式化验证	235
15.4	小结	237

第三部分 附 录

附录 A	强度建模和高级线网类型定义	240
附录 B	PLI 子程序清单	243
附录 C	关键字、系统任务和编译指令	259
附录 D	形式化语法定义	261
附录 E	Verilog 有关问题解答	290
附录 F	Verilog 举例	293
参考文献		303
译者后记		304

第一部分 Verilog 基础知识

第 1 章

Verilog HDL 数字设计综述

第 2 章

层次建模的概念

第 3 章

基本概念

第 4 章

模块和端口

第 5 章

门级建模

第 6 章

数据流建模

第 7 章

行为级建模

第 8 章

任务和函数

第 9 章

实用建模技术

第 1 章 Verilog HDL 数字设计综述

1.1 数字电路 CAD 技术的发展历史

在过去的 20 多年中，数字电路设计技术的发展非常迅速。设计人员最早使用真空管和晶体管来设计数字电路。后来他们把逻辑门安置在单个芯片上，于是发明了集成电路。第一代集成电路（Integrated Circuit, IC）的门数非常少，称为小规模集成电路（Small Scale Integrated, SSI）。随着制造工艺技术的发展，设计者可以在单个芯片上布置数百个逻辑门，我们称之为中规模集成电路（Medium Scale Integrated, MSI）。随着大规模集成电路（Large Scale Integrated, LSI）的出现，数千个逻辑门能够集成在一起。设计过程由此开始变得非常复杂，因此设计者希望某些设计阶段能够自动完成。正是这种需要促进了电子设计自动化（Electronic Design Automation, EDA）的出现和发展^①。设计者开始使用电路和逻辑仿真技术对使用的基本组件的功能进行验证，这些基本组件的规模一般相当于几百个晶体管。不过这时的测试仍然在面包板上完成，设计人员在设计图纸或计算机图形终端上用手工完成电路的版图设计。

超大规模集成电路（Very Large Scale Integrated, VLSI）的出现，使得设计人员可以将超过 10 万个晶体管集成在一块芯片上。在这种情况下，已经不可能在面包板上对设计的功能进行验证了。计算机辅助技术对于超大规模集成电路的设计和验证变得非常重要，同时，使用计算机进行电路版图的布局和布线也开始流行，设计者在图形终端上用手工完成数字电路的门级设计。从小的功能模块开始设计，逐步使用小的功能模块来搭建高层功能模块，直到完成顶层设计。在最后制成芯片之前，设计者还会使用逻辑仿真工具对设计的功能进行验证。

随着设计规模的不断增大，其功能越来越复杂，逻辑仿真在整个设计过程中的作用越来越重要，使得设计者可以尽早地排除设计结构中存在的问题。

1.2 硬件描述语言的出现

很久以来，人们使用诸如 FORTRAN, Pascal, C 等语言来进行计算机程序设计，这些程序本质上是顺序执行的。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设计。在这种情况下，许多种硬件描述语言（Hardware Description Languages, HDL）应运而生。设计者可以使用它对硬件中的并发执行过程建模。在出现的各种硬件描述语言中，Verilog HDL 和 VHDL 使用得最为广泛。Verilog HDL 于 1983 年源自 Gateway Design Automation 公司。稍后，由美国国防部的高级研究计划署牵头（制定合同）开发了 VHDL。设计人员很快认可了 Verilog HDL 和 VHDL 这两种语言，使用它们对大型数字电路进行仿真。

① 本书第一版中使用了 CAD 工具这个术语。从技术角度看，CAD（Computer-Aided-Design，计算机辅助设计）工具这个术语指的是设计后端使用的工具，这些工具可以完成布局、布线和芯片的版图绘制等工作。而 CAE（Computer-Aided-Engineering，计算机辅助工程）工具这个术语指的是设计前端使用的工具，如 HDL 仿真、逻辑综合和时序分析。过去设计人员常把 CAD 和 CAE 这两个术语混用。目前，EDA 的范围包括了 CAD 和 CAE 两个部分。为了简单起见，在本书中我们把所有的设计工具都称为 EDA 工具。

虽然当时用 HDL 进行逻辑验证已经很普及,但是设计人员仍然需要用手将基于 HDL 的设计转换为由相互连接的逻辑门表示的电路简图。在 20 世纪 80 年代后期,逻辑综合工具的发展对数字电路的设计方法学产生了巨大的影响。设计者可以使用 HDL 在寄存器传输级(Register Transfer Level, RTL)对电路进行描述。在这种设计方法中,设计者只需要说明数据(信息)是如何在寄存器之间移动以及如何被处理的,而构成电路的逻辑门及其相互之间的连接数据(资料)由逻辑综合工具自动地从 RTL 描述中提取出来。

逻辑综合工具的出现和发展使得 HDL 在数字电路设计中占据了重要的地位。设计者不再需要通过手工用逻辑门来搭建电路。他们可以使用硬件描述语言来描述电路的功能和数据的流向,然后由逻辑综合工具自动综合出由逻辑门及其相互连接构成的电路结构细节,实现 HDL 所描述和指定的特定功能。

同样, HDL 在系统级设计中也得到了应用。HDL 用来仿真电路板、互连总线、FPGA (Field Programmable Gate Arrays) 以及 PAL (Programmable Array Logic) 等。通常的方法是使用 HDL 单独设计每个芯片,然后通过仿真来验证整个系统的功能。

目前, Verilog HDL 已经是公认的 IEEE 标准。Verilog HDL 的第一个标准 (IEEE 1364-1995) 是在 1995 年批准的。最近公布的 IEEE 1364-2001 标准与原标准相比有了显著的改进。

1.3 典型设计流程

图 1.1 表示的是超大规模集成电路设计的典型流程。图中不带阴影的方框表示设计描述的层次,带阴影的方框表示设计的过程。

在任何设计流程中,必须首先编写设计电路的技术指标和功能要求细节,从抽象的角度对电路的功能、接口和总体结构进行描述,在这一阶段无须考虑电路的具体实现方式。接下来设计者使用行为级描述来分析电路的功能、性能、标准兼容性以及其他高层次的问题。行为级描述一般也使用 HDL 来编写^①。

由行为级描述向 RTL 级描述的转换是由设计者手工完成的。在这个过程中,设计者需要对实现电路功能的数据流进行详细描述。在以后的各个设计步骤中,设计者都可以借助工具软件。

综合工具的作用是将 RTL 级描述转换成门级网表。门级网表从逻辑门及其相互连接关系的角度来描述电路的结构。综合工具需要保证综合出来的门级网表满足时序、面积以及功耗的要求。自动布局、布线工具读入综合得到的网表并生成电路的版图。电路的版图经过验证就可以制成芯片。

在整个设计流程中,设计的重点主要在于手工对 RTL 描述的优化。在 RTL 描述完成之后,设计者就可以在 EDA 工具的辅助下完成后续的设计过程。从 RTL 级描述着手可以将设计周期从几年缩短为几个月,也有可能较短的时间内对设计进行多次调整和改进。

最近出现的行为级综合工具可以将电路或算法的行为级描述转换为 RTL 级描述。随着行为级综合工具的不断完善,数字电路设计将越来越类似于软件程序设计。设计者只需在非常抽象的层次使用 HDL 对电路的算法进行描述,然后在 EDA 工具的帮助下完成从行为级描述到最终芯片结构的转换。

^① 新出现的 EDA 工具可以对电路的行为描述进行仿真。这些新工具将 HDL 和面向目标的编程语言(如 C++等)的强大功能结合在一起。有了这些工具,则不需要使用 Verilog HDL 来编写电路的行为模型。

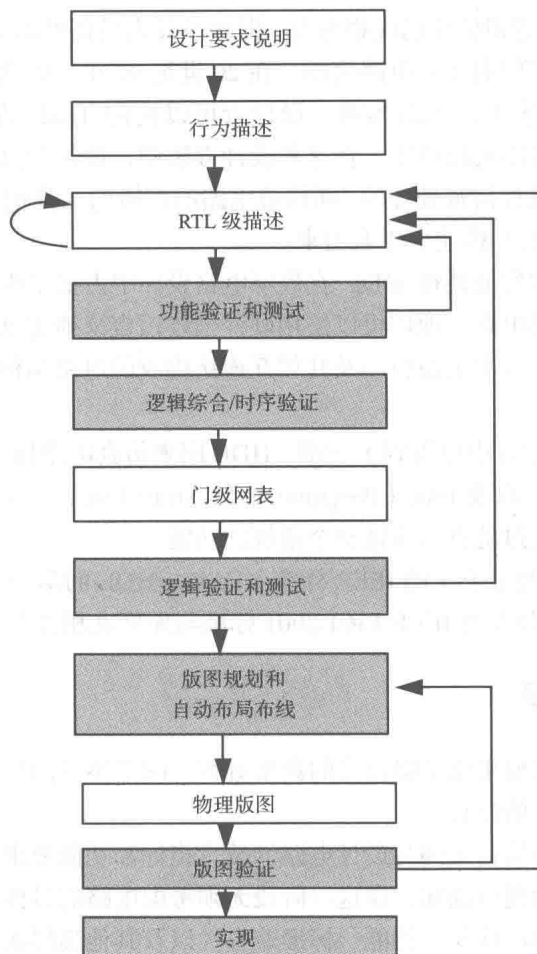


图 1.1 典型的设计流程

虽然 EDA 工具能够使设计过程自动化并显著缩短设计周期,但设计者仍然是整个设计过程的核心。如果使用不当,EDA 工具也会产生很差的结果。因此设计者还必须对这种设计方法的缺陷有所了解,才能借助于 EDA 工具得出优化的设计。

1.4 硬件描述语言的意义

与传统的基于电路原理图的设计方法相比,使用硬件描述语言(HDL)进行设计具有许多优点:

- 通过使用 HDL,设计者可以在非常抽象的层次上对电路进行描述。设计者可以在 RTL 级对电路进行描述而不必选择特定的制造工艺,逻辑综合工具能够将设计自动转换为任何一种制造工艺版图。如果出现新的制造工艺,设计者不必对电路进行重新设计,只需将 RTL 级描述输入逻辑综合工具,即可生成针对新工艺的门级网表。逻辑综合工具将根据新的工艺对电路的时序和面积进行优化。
- 通过使用 HDL,设计者可以在设计周期的早期对电路的功能进行验证。设计者可以很容易地对 RTL 描述进行优化和修改,满足电路功能的要求。由于能够在设计初期发现和排除绝大多数设计错误,因此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,避免了设计过程的反复,显著地缩短了设计周期。