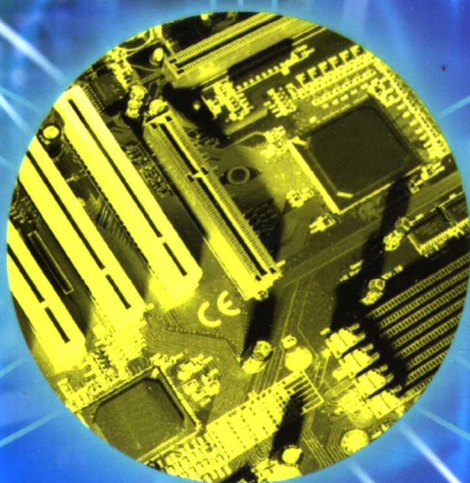


系统芯片 (SoC) 验证方法与技术

System-on-a-Chip Verification Methodology
and Techniques



Prakash Rashinkar

[美] Peter Paterson 著

Leena Singh

孙海平 丁健 译



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

系统芯片 (SoC) 验证方法与技术

System-on-a-Chip Verification Methodology and Techniques

Prakash Rashinkar

[美] Peter Paterson 著
Leena Singh

孙海平 丁 健 译

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书从最高层次的系统级验证直至最终的物理实现和签付,详细介绍了各种设计抽象级别和各阶段所涉及到的各种验证方法及工具。主要内容包括各种不同类型的仿真、软件/硬件协同验证、数字/模拟混合验证、网表静态验证、物理验证、测试平台迁移、形式模型与等价性检查、代码静态检查与代码覆盖状况分析、定向随机测试等验证技术。

本书以蓝牙系统芯片为例,各章中有结合实际的代码和脚本可供读者参考,以帮助读者进一步深入理解。

本书内容全面、翔实,可作为从事系统芯片设计的工程人员、研究者和高等院校相关专业师生的参考材料,对于从事传统集成电路设计和验证的人员有较高的指导和借鉴价值。

Copyright © 2001 Kluwer Academic Publishers.

本书中文简体专有翻译出版权由美国 Kluwer Academic Publishers 授予电子工业出版社。该专有出版权受法律保护。

版权贸易合同登记号 图字: 01-2004-2717

图书在版编目(CIP)数据

系统芯片(SoC)验证方法与技术/(美)拉申卡(Rashinkar,P.), (美)帕特森(Paterson,P.), (美)信赫(Singh,L.)

著;孙海平,丁健译. —北京:电子工业出版社, 2005.1

书名原文: System-on-a-Chip Verification Methodology and Techniques

ISBN 7-121-00589-1

I. 系… II. ①拉…②帕…③信…④孙…⑤丁… III. 集成电路—芯片—设计 IV. TN402

中国版本图书馆 CIP 数据核字(2005)第 120854 号

责任编辑:高买花

印 刷:北京兴华印刷厂

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

经 销:各地新华书店

开 本:787×1092 1/16 印张:17.5 字数:448 千字

印 次:2005 年 1 月第 1 次印刷

印 数:4 000 册 定价:29.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。
联系电话:(010) 68279077。质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

译者序

硅集成电路自 1958 年美国德州仪器公司 (TI) 发明至今的 40 多年来, 为满足电子整机系统及其终端产品日新月异的发展需求, 在技术及其产品市场需求的驱动下, 已经历了孕育期、开发期、发展期和成熟期。在整个集成电路发展历程中, 一直遵从摩尔定律, 追求着集成度的提高和成本降低的变化规律。

在 20 世纪 90 年代中末期, 随着互联网的广泛普及, 数字化、个性化、微型化、低功耗的计算机和通信及消费类电子 (即 3C) 相融合的产品已经成为全球性的发展趋势。针对这新一轮的发展, 集成电路除了提升工艺技术、减小特征尺寸、增大晶圆片直径及完善封装测试技术外, 更重要的是必须改善乃至变革集成电路的设计方法, 以适应集成电路产品创新速度加快及设计效率提高的需求。

对于系统芯片这个新鲜事物, 至今还没有完整、权威的定义, 业界普遍认为是指单个芯片上实现一个系统所具有的信号采集、转换、储存、处理和输入/输出 (I/O) 等功能的电路, 是在单个芯片上广泛采用预先设计好的知识产权 (IP) 模块、通过各种重用技术而快速开发出来的集成电路。系统芯片的基本特征是内嵌 CPU 核和诸如 DSP 之类的核心 IP、一定容量的存储器、强大的信号和数据处理能力、具有片内操作系统和应用软件、采用软件/硬件协同设计等。

与传统芯片相比较, 系统芯片以更低的功耗、更高的性价比、更快的上市时间等特点, 极大地提高了系统整机的性能, 降低了成本和体积, 成为将集成电路设计与电子系统设计及整机应用结合的纽带, 以达到省时、省钱、提高效益的目的。

系统芯片的出现, 既带来了各方面的机遇, 也为设计带来了诸多挑战。它将系统概念设计、架构设计与系统建模、寄存器传输级电路设计、物理设计等原本属于不同学科和应用领域的问题集中到一块系统芯片的设计之中。所有设计环节无一不需要经过缜密而严格的仿真和验证, 这正是系统芯片设计所面临的重大挑战和难点课题之一。正如原书序所言: “验证不仅是很棘手, 而且是非常棘手”, “验证至少占据了 70% 的设计工作量。”

译者在多年的芯片设计工作中, 也深切体会到了验证工作对芯片设计具有决定性的作用。从制定系统设计规约开始, 验证工作就已经开始。在架构设计、行为级的系统建模到模块化的设计和实现, 以及最终的网表和后端布局、布线等一系列过程中, 验证工作都一直是如影相随, 总是要不断地进行仿真和验证。在一个设计团队中, 专门的验证小组和验证部门也已经成为芯片设计团队主要的组成部分, 验证方法和验证工具也成为重要的研究和开发方向。

因此, 对于国内集成电路设计人员而言, 全面了解和深入理解验证的重要性和各种验证方法极为必要。本书几位作者都是长期研究系统芯片设计和验证工作的资深专家, 对于系统芯片的验证有着最切身的体会和全面准确的把握, 总结和归纳了很多卓有成效的验证方法和

经验。从最高层次的系统级验证直至最终的物理实现和签付，本书详尽地阐述了在各种设计抽象级别和各个阶段所涉及到的各种验证方法及其主要特点和工具。全书还有大量结合实际的示例、代码和脚本可供读者学习和参考，帮助读者进一步深入理解。

本书内容全面、翔实，可以作为从事系统芯片设计的工程设计人员、相关专业师生的参考资料，书中很多内容对于从事传统集成电路设计和验证的人员也有着很强的指导和借鉴作用。

译者在翻译过程中保留了原书的部分书写规则，并且修正了原著中的错误和疏漏之处。

本书由孙海平和丁健主译；张睿卿、林爽和齐海鹏参与了部分章节的翻译和修订，孙海平统稿。在此，谨向为本书的出版付出辛勤劳动的所有人员致以诚挚的感谢！还要特别感谢本书的责任编辑高买花老师的细致、耐心和敬业精神！

在翻译过程中，我们力求译意准确、术语恰当，但限于水平，必然存在错误和不足之处，恳请读者及时通过电子邮件（hp.sun@263.net 和 johnsond@public3.bat.net.cn）向我们提出批评指正。

译 者
2004 年 8 月

前 言

近几年来，半导体工业面临的主要挑战是如何才能花费比以往少得多的时间设计出功能复杂度和应用领域的多样性满足需求的系统芯片。在有待解决的各种挑战中，真正居首位的是验证。众多学者一致认为验证至少占据了 70% 的设计工作量。

对最终设计验证其正确性，被视为对设计更复杂的系统芯片（SoC）和开发应用前沿的加工工艺起到了关键性的保障作用。在虚拟插座接口联盟（VSIA）于 1999 年成立验证专题研究组（Verification Workshop）之际，众多世界级的验证专家达成“验证很棘手”的结论。数周之后，经过相当深入的探讨，最终的结论是“验证不仅是很棘手，而且是非常棘手”。另外一项调查结论是：没有任何一种设计工具能够单独用来解决此问题。相反地，需要结合包括传统仿真、定向验证和随机验证以及形式技术在内的一系列工具和技术来将设计错误减少到最少。

第三届 VSIA 验证会议得出“验证不仅是非常棘手，而且是非常、非常棘手”的结论。这一趣闻进而激发出这本讨论系统芯片验证的专题著作。有效验证是设计重用的基础，通过设计重用许可来获取生产率收益对于开发利用先进的加工工艺是必不可少的。每一个可重用的设计功能块都必须附带可重用的、用来在功能块内实现透彻验证的完备的测试平台或成套验证材料。此外，设计团队必须懂得“重用而非返工”的原则，即无须修改地将各个设计功能块及其相关的成套验证纳入系统芯片设计中。

Rashinkar、Paterson 和 Singh 的这本著作是一本全面介绍系统芯片验证方法的综合指南，也是介绍验证工程师和设计工程师能够采用的各种工具、技术和方法的知识宝库。本书作者以清晰的逻辑思路介绍验证的方方面面，从系统级验证到各个功能块的验证，既有数字信号功能块，也有模拟/混合信号功能块，随后介绍仿真、软件/硬件协同验证、网表静态验证、物理验证等技术。需要特别留意各种新技术，如系统级设计途径、测试平台迁移、形式模型与等价性检查、代码静态检查与代码覆盖状况分析、定向随机测试、用于测试平台的基于事务的验证技术、各种原型和仿效途径等。本书结合一种支持蓝牙消费电子应用的架构来讲解各种验证方法和技术的实际应用。

本书将为系统芯片和复杂 ASIC 设计项目在开发恰当的验证策略方面提供重大帮助。使用这些最先进的技术将有助于引领大家更快地迈向“设计与验证重用而无须返工”这一梦想。让我们向这本意义深远的著作致以热烈的喝彩！

Grant Martin

系统级设计与验证领域研究员

Larry Rosenberg

VSIA 技术委员会主席

作者简介

Prakash Rashinkar 在通信卫星、运载火箭、太空船地面系统、高性能计算机、网络交换、多媒体和无线应用等领域拥有超过 15 年的嵌入式系统设计和验证工作经验。他毕业于印度 Warangal 的 Regional Engineering College，获电气工程理学硕士学位。他在 Cadence Design Systems 公司领导一个团队负责系统芯片验证方法的研究工作，是 VSIA 功能验证研发工作组成员。目前，他是 Cadence 公司纵向市场和设计环境工作组的核心设计师。

Peter Paterson 在 ASIC 和计算机系统设计领域拥有超过 20 年的工作经验。他毕业于苏格兰 Robert Gordon 大学，获电气工程理学学士学位。他在 Unisys 公司时带领团队设计出了第一块主机芯片 SCAMP 和一款单芯片砷化镓 (GaA) 处理器，这两种器件是当今系统芯片器件的早期雏形。在 Cadence 公司工作期间，他缔造的基于平台的系统芯片设计方法交付给 Scottish Enterprise 公司作为 ALBA 工程的一部分。他也是 VSIA 功能验证研发工作组成员。目前，他是 Vixel 公司的 ASIC 研发总监。

Leena Singh 在多媒体、无线通信、过程控制等应用领域拥有超过 9 年的 ASIC 设计和验证的工作经验。Leena 毕业于印度 Chandigarh 的 Punjab 大学，获电气工程理学学士学位。她是 Cadence 公司系统芯片验证方法研发成员。目前，她是 Cadence 公司纵向市场和设计环境工作组的主任设计工程师。

目 录

第 1 章 绪言	(1)
1.1 工艺的挑战	(1)
1.1.1 时序收敛	(2)
1.1.2 设计能力	(2)
1.1.3 物理属性	(2)
1.1.4 设计生产率鸿沟	(3)
1.1.5 面市时间的发展趋势	(3)
1.1.6 系统芯片技术	(4)
1.2 可供选用的验证技术	(5)
1.2.1 仿真技术	(5)
1.2.2 静态技术	(7)
1.2.3 形式技术	(8)
1.2.4 物理验证与分析	(9)
1.2.5 各种验证做法的比较	(9)
1.3 验证方法	(11)
1.3.1 系统级验证	(11)
1.3.2 系统芯片硬件寄存器传输级验证	(12)
1.3.3 系统芯片软件验证	(12)
1.3.4 网表验证	(12)
1.3.5 物理验证	(12)
1.3.6 器件测试	(12)
1.4 测试平台的建立	(13)
1.4.1 采用硬件描述语言建立测试平台	(13)
1.4.2 采用编程语言接口建立测试平台	(14)
1.4.3 基于波形建立测试平台	(14)
1.4.4 基于事务建立测试平台	(14)
1.4.5 基于(设计)规约建立测试平台	(14)
1.5 测试平台的迁移	(14)
1.5.1 将测试平台从功能性迁移至寄存器传输级	(15)
1.5.2 测试平台从寄存器传输级向网表的迁移	(15)
1.6 验证语言	(15)
1.7 验证 IP 重用	(16)
1.8 验证途径	(16)
1.8.1 自顶而下的设计与验证途径	(16)
1.8.2 自底而上的验证途径	(17)
1.8.3 基于平台的验证途径	(19)

1.8.4 受系统接口驱动验证的途径	(19)
1.9 验证和器件测试	(20)
1.9.1 器件测试的挑战	(20)
1.9.2 测试策略	(20)
1.10 验证计划	(21)
1.11 蓝牙系统芯片的参考设计	(24)
1.11.1 蓝牙器件中的元素	(24)
1.11.2 蓝牙网络	(25)
1.11.3 蓝牙系统芯片	(25)
参考文献	(28)
第2章 系统级验证	(30)
2.1 系统设计	(30)
2.1.1 功能/行为设计	(31)
2.1.2 架构映射	(31)
2.2 系统验证	(32)
2.2.1 功能验证	(32)
2.2.2 性能验证	(32)
2.2.3 系统级测试平台	(33)
2.2.4 建立系统级测试平台	(33)
2.2.5 系统测试平台的评价尺度	(33)
2.2.6 运用系统级测试平台	(34)
2.2.7 系统测试平台的迁移	(36)
2.3 蓝牙系统芯片	(37)
参考文献	(42)
第3章 功能块级验证	(43)
3.1 IP 功能块	(43)
3.2 功能块级验证	(44)
3.3 蓝牙系统芯片的功能块细节	(45)
3.3.1 仲裁器	(45)
3.3.2 仲裁器的测试平台	(51)
3.3.3 译码器	(59)
3.3.4 ASB 主方	(59)
3.3.5 ASB 从方	(59)
3.3.6 ASB/APB 桥	(60)
3.4 代码静态检查	(60)
3.5 模型形式检查	(61)
3.5.1 模型检查的时间	(62)
3.5.2 模型检查的局限	(62)
3.5.3 模型检查方法	(62)
3.5.4 模型检查的步骤	(65)

3.6	功能验证与仿真	(71)
3.6.1	黑盒验证途径	(71)
3.6.2	白盒验证途径	(72)
3.6.3	灰盒验证途径	(72)
3.6.4	仿真	(73)
3.7	协议检查	(73)
3.7.1	存储器/寄存器访问信号	(74)
3.7.2	协议检查示例	(74)
3.8	定向随机测试	(80)
3.9	代码覆盖状况分析	(82)
3.9.1	覆盖状况分析的类型	(82)
3.9.2	代码覆盖率分析	(84)
	参考文献	(87)
第4章	模拟与混合信号仿真	(89)
4.1	混合信号仿真	(89)
4.2	设计抽象层次	(90)
4.3	仿真环境	(91)
4.3.1	选择仿真环境	(92)
4.3.2	现行环境的局限	(93)
4.4	使用 SPICE	(93)
4.5	仿真方法	(93)
4.6	蓝牙系统芯片中的数模转换器	(95)
4.6.1	DAC 的测试平台	(99)
4.6.2	建立网表	(101)
4.6.3	仿真	(104)
4.6.4	响应	(104)
4.7	含模拟混合信号功能块的芯片级验证	(104)
	参考文献	(105)
第5章	仿真	(106)
5.1	功能仿真	(106)
5.2	测试平台壳	(107)
5.2.1	蓝牙系统芯片的功能块细节	(110)
5.2.2	测试向量转换	(110)
5.2.3	激励生成	(111)
5.2.4	激励的截取	(115)
5.2.5	结果检查	(116)
5.2.6	从方的测试平台壳	(117)
5.3	基于事件的仿真	(124)
5.3.1	基于事件的仿真工具	(125)
5.3.2	基于事件的仿真环境	(125)

5.3.3	选择基于事件的仿真方案	(125)
5.3.4	基于事件的仿真流程	(126)
5.4	基于周期的仿真	(126)
5.4.1	何时采用基于周期的仿真	(127)
5.4.2	基于周期的仿真环境	(127)
5.4.3	选择基于周期的仿真方案	(128)
5.4.4	基于周期的仿真的局限性	(128)
5.4.5	基于周期的仿真流程	(128)
5.4.6	基于事件和基于周期的仿真的比较	(129)
5.5	ASB/APB 桥的仿真	(130)
5.5.1	ASB/APB 功能块	(130)
5.5.2	设计寄存器传输级代码	(131)
5.5.3	基于事件的仿真测试平台	(138)
5.5.4	运行仿真	(138)
5.6	基于事件和基于周期相混合的仿真	(140)
5.7	基于事务的验证	(141)
5.7.1	基于事务的验证中的元素	(141)
5.7.2	基于事务的验证环境	(142)
5.7.3	建立测试平台	(143)
5.7.4	事务分析	(147)
5.7.5	基于事务的验证中的功能覆盖状况	(147)
5.7.6	基于事务的验证流程	(148)
5.7.7	蓝牙系统芯片	(149)
5.8	仿真加速	(162)
5.8.1	仿效	(162)
5.8.2	何时采用仿效	(164)
5.8.3	仿效环境	(165)
5.8.4	选择仿效方案	(166)
5.8.5	仿效的局限性	(166)
5.8.6	仿效的流程	(167)
5.8.7	快速原型系统	(167)
5.8.8	硬件加速器	(168)
5.8.9	设计划分	(168)
	参考文献	(168)

第 6 章	软件/硬件协同验证	(170)
6.1	软件/硬件协同验证环境	(170)
6.2	仿效	(171)
6.3	软原型或虚拟原型	(172)
6.3.1	软原型的局限性	(173)
6.3.2	建立软原型的流程	(173)

6.3.3	蓝牙系统芯片的软原型	(174)
6.4	协同验证	(190)
6.4.1	协同验证环境	(191)
6.4.2	选择协同验证环境	(191)
6.4.3	协同验证方法	(192)
6.4.4	UART 的协同验证	(193)
6.5	快速原型系统	(212)
6.5.1	快速原型系统的局限性	(212)
6.5.2	可重配置快速原型系统	(213)
6.5.3	专用快速原型系统	(215)
6.6	软件/硬件验证方法的比较	(218)
6.7	基于 FPGA 的设计	(218)
6.7.1	基于 FPGA 的设计指导原则	(219)
6.7.2	基于 FPGA 的设计流程	(219)
6.8	开发电路印制板	(221)
6.9	软件测试	(222)
6.9.1	软件开发生命周期	(222)
6.9.2	软件开发的指导原则	(224)
6.9.3	软件测试的最佳经验	(225)
6.9.4	调试工具	(225)
6.9.5	中断的调试	(227)
	参考文献	(229)
第 7 章	网表静态验证	(232)
7.1	网表验证	(232)
7.2	蓝牙系统芯片的仲裁器	(234)
7.3	等价性检查	(234)
7.3.1	等价性检查解决方案的选择	(235)
7.3.2	等价性检查的流程	(236)
7.3.3	寄存器传输级对寄存器传输级的验证	(237)
7.3.4	寄存器传输级对门级网表的验证	(237)
7.3.5	从门级网表到门级网表验证	(239)
7.3.6	调试	(240)
7.3.7	对仲裁器进行等价性检查	(242)
7.4	时序静态验证	(244)
7.4.1	选择时序静态验证解决方案	(245)
7.4.2	时序静态验证的流程	(245)
7.4.3	对仲裁器进行时序静态验证	(246)
	参考文献	(251)
第 8 章	物理验证与设计签付	(252)
8.1	设计检查	(252)

8.2 物理效应分析	(252)
8.2.1 寄生效应的提取	(252)
8.2.2 电感效应	(253)
8.2.3 信号完整性	(253)
8.2.4 电迁移效应	(255)
8.2.5 亚波长挑战	(255)
8.2.6 工艺天线效应	(256)
8.3 设计签付	(256)
参考文献	(257)
附录 术语表	(259)

第 1 章 绪 言

在深亚微米级的工艺中，单个集成电路所能够实现的设计能力让人们可以灵活地将最终产品的所有主要功能都集成到单块系统芯片（SoC, System-on-a-Chip）之中。然而，向系统芯片设计的演进对传统的验证途径提出了种种挑战。

本章将讨论以下主题：

- 工艺的挑战
- 可供选用的验证技术
- 验证方法
- 测试平台的建立与迁移
- 验证语言
- 验证 IP 重用
- 验证途径
- 验证与器件测试
- 验证计划
- 参考设计示例

本章将介绍蓝牙参考设计，在本书的各章中将以它为例来介绍系统芯片的验证方法。

1.1 工艺的挑战

硅工艺铸片线（foundry）不断努力缩小 IC 芯片上所能够实现的硅结构的物理尺寸。随着这种尺寸上的缩小而带来的是电路容量、性能两方面的显著提升。摩尔定律（Moore's Law）刻画了这种工艺进展，该定律认为将逻辑门（晶体管）集成到单个硅芯片中的能力每 18 个月翻一番。图 1-1 表明了目前硅芯片工艺的发展态势。

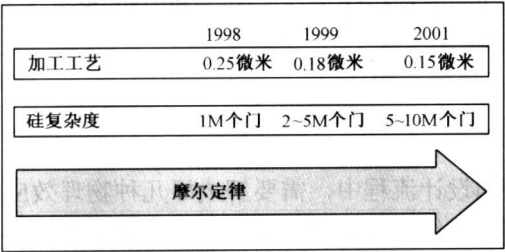


图 1-1 硅芯片工艺的发展

因为硅工艺达到了 0.25 微米或更低的线宽，设计界正面临着若干重大的挑战性难题。这些难题大致可以归为以下三类：

- 时序收敛；
- 设计能力；
- 物理属性。

1.1.1 时序收敛

传统的硅设计流程使用连线负载统计模型来为版图之前的时序分析并估算金属互连线（延迟）。采用这种途径，估算某个特定节点上的负载使用的是受该节点驱动的各个逻辑门的输入电容之和，以及根据功能块的规模与受驱动的逻辑门的数目所做出的连线的统计估算值。对于 0.25 微米或更宽的几何尺寸，采用这种途径，倘若在版图之前的阶段满足了时序目标和约束，那么往往能够在物理设计之后就达到同样的结果。

成功地实现时序收敛的原因在于：在这样的几何尺寸下，逻辑门传播延迟和逻辑门负载电容支配着总延迟，而不太受互连线延迟的影响。在深亚微米级的工艺中，互连线延迟变得更加重要，如果要实现时序收敛就必须对其进行精确估算。连线负载统计模型之所以不精确，是因为它们代表的是根据功能块规模而做出的统计值。连线负载分布曲线在平均值处可能会差异很大，因此在分布曲线的“尾部”互连线延迟被低估了很多。

1.1.2 设计能力

采用 0.15 微米或更小的深亚微米工艺，将一千万个以上逻辑门集成到单块芯片中是切实可行的，然而这给设计流程中很多工具的设计能力带来了挑战性难题。要控制这种复杂程度，深亚微米设计系统必须采用以下解决方案：

- 层次化设计；
- 设计重用。

层次化设计流程支持在设计对象内实现多个结构层次。顶层是功能块之间的互连，其下的各个层次为那些功能块提供了设计细节，既可以使用互连起来的子功能块，也可以使用库元素。采用这种方式来划分设计，就能够限制住设计对象在特定层次上的复杂度。要支持这种思路，就必须有可能生成功能块在每一个设计层次上的抽象模型，以便在更高的层次上加以使用。

设计重用将既有的功能块和新创的功能块集成起来。这样将有助于深亚微米设计的研发。首先，因为设计对象中的一个或多个功能块已经事先设计好了，因而减少了原创性的设计工作量。其次，事先设计好的功能块已经得到证明（certify）或证实（validate），可以将它们视为黑盒，因而无须再次证实。

1.1.3 物理属性

在深亚微米级工艺下的设计流程中，需要解决好几种物理效应。向深亚微米的发展导致器件几何尺寸更小、金属互连线层数更多、供电电压更低、器件阈值更低、时钟频率更高，以及数十万个器件可以集成在单个芯片之中。这些因素导致信号完整性（SI, Signal Integrity）和设计完整性（DI, Design Integrity）比在几何尺寸更为宽松时得到更多的关注。信号完整性问题包括串扰、IR 电压降以及电源和地的反弹跳动*。设计完整性问题包括电迁移、热电子和

* 译者注：地的反弹跳动（ground bounce）亦有学者译为“地弹”或“地弹反射”。

导线自热。在更为宽松的几何尺寸下，这些信号完整性和设计完整性问题是次要的因素，因而可以忽略不计。

因为几何尺寸在缩小，这些问题将变得更为普遍，因此需要有“签付”（sign-off）这一幕，在送去制造（fabricate）之前检查各方面的设计违规（violation），一旦检测出现违规，马上加以修正。在深亚微米级，这些问题上升为设计过程需要自发地检测和修正各种设计违规。设计流程中缺少了“找寻并修正”的能力，将会出现数以万计的设计违规，这使在后续处理步骤中修正这些违规是不可能的。任何一种系统芯片验证方法都必须解决深亚微米设计中的设计能力和物理属性这两类挑战性难题。

1.1.4 设计生产率鸿沟

设计生产率拖累了工艺进步可能带来的设计密度的提高。图 1-2 体现了设计生产率鸿沟。其中，逻辑门密度的单位是门数/芯片，设计生产率的单位是门数/小时。

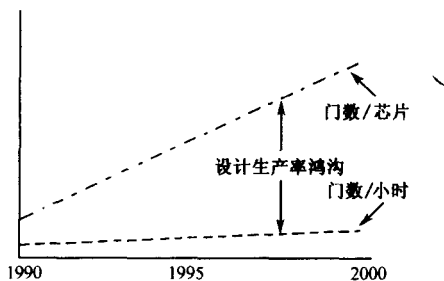


图 1-2 设计生产率鸿沟

IC 复杂度的日益增长对系统工程师和验证工程师两者都提出了挑战。设计生产率鸿沟不能够在遇到问题时通过简单的投入更多的工程师来得到解决。首先，没有足够多的称职工程师来解决问题，即使有，每个设计团队究竟能壮大到何种规模也将受到实际的限制。随着设计团队的壮大，因此需要协调到让每一个人都步调一致的程度。换句话说，需要新的方法让设计过程变得更具有生产率。

工业界采用设计重用策略来回应这种挑战。利用既有功能块，也被称为知识产权（IP, Intellectual Property）功能块或虚元件（VC, Virtual Component），将减少实现新的设计对象时所需要的原创性设计工作量。采用基于平台的设计方法，设计重用将超越各单个功能块的重用。基于平台的设计就是确定出一族产品所公用的一组核心元素，将它们集成起来，并作为单个实体来验证。然后，通过向该芯核添加一个个设计元素来设计出实际的产品。各个元素既可以是其他的 IP 功能块，也可以是新创作出来的元素。这一设计重用理念不仅减少了设计工作量，也可观地减少了在实现新设计对象时所付出的验证工作量。

1.1.5 面市时间的发展趋势

随着设计复杂度的爆炸式增长，电子器件的面市时间却要求以令人震惊的速度缩短。图 1-3 是军用器件、工业用器件和消费类电子器件的面市时间的发展趋势。

不仅所有的应用领域都面临设计周期缩短的现象，而且还伴随着从传统的应用（军用和工业用）向生命周期最短的消费类产品的转移。这种发展变化使其设计周期缩短得比其他任

何细分市场都快。

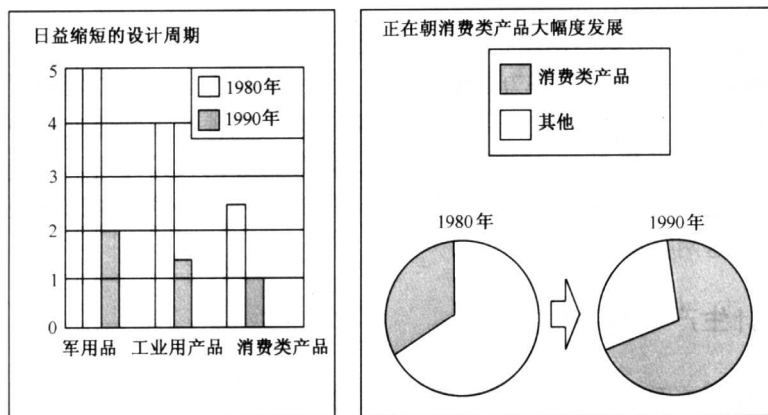


图 1-3 面市时间的发展趋势

这些工艺和市场方面的挑战对验证方法和工具将产生惊人的影响力。据估计，总的开发工作量中有 40%~70% 用于验证任务。很显然，如果要满足市场的全面挑战，就必须更高效地完成这些验证举动。

1.1.6 系统芯片技术

向系统芯片设计方向的发展还带来了传统验证途径的挑战。系统芯片含有可编程元素（控制处理器和数字信号处理器）、硬件元素（数字信号功能块和模拟/混合信号功能块）、软件元素、复杂总线架构、时钟分布、功耗分布、测试结构以及总线，如图 1-4 所示。

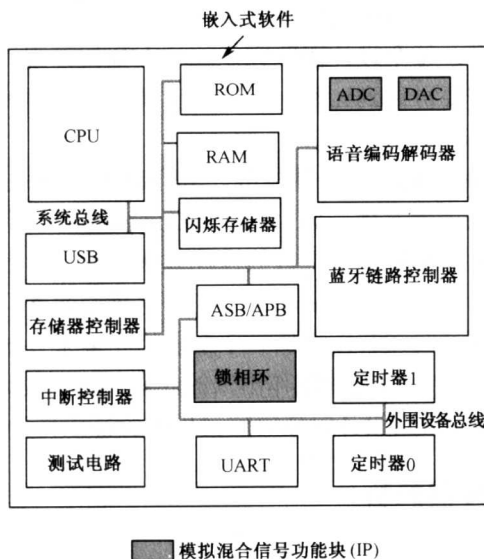


图 1-4 系统芯片

传统的设计方法，即采用该学科专门的方法和工具各自独立地对各种不同学科领域的设计（如数字电路、模拟混合信号电路、嵌入式软件）加以开发，已遭到挑战。在系统芯片中，