



普通高等教育“十一五”国家级规划教材
普通高等教育电子信息类规划教材

系统芯片(SOC) 设计原理

罗胜钦 编著



免费
电子课件



机械工业出版社
CHINA MACHINE PRESS

TN430. 2/5

2007

普通高等教育“十一五”国家级规划教材
普通高等教育电子信息类规划教材

系统芯片 (SOC) 设计原理

罗胜钦 编著
毛志刚 张申科 审

机械工业出版社

本书为普通高等教育“十一五”国家级规划教材。

系统芯片 (System On a Chip, 简称 SOC) 是微电子技术发展的一个新的里程碑。本书介绍在 EDA 工具的平台, 进行以系统级设计为核心的系统芯片的设计方法。本书从基本单元电路设计出发, 以 VHDL 语言为基本设计手段, 讨论了各种典型的数字集成系统的设计, 以及系统芯片实现的两种基本途径: 半定制的高密度可编程逻辑器件 (HDPLD) 的实现和全定制的专用集成电路 (ASIC) 的实现。

本书主要内容包括: 集成电路工艺及版图基础, CMOS 数字电路, 硬件描述语言 VHDL 及数字系统的设计, 系统集成芯片的体系结构, 高密度可编程逻辑器件, 可编程系统芯片 (SOPC), 专用集成电路设计和可测试结构设计。全书语言流畅, 循序渐进地讨论了系统芯片各方面的内容。每章后附有习题, 供课后练习。

本书配有免费电子课件, 欢迎选用本书作教材的老师索取, 电子邮箱: wbj@mail.machineinfo.gov.cn。

本书可作为高等院校电子信息类高年级本科生与研究生的教材, 也可作为相关领域工程技术人员的参考资料。

图书在版编目 (CIP) 数据

系统芯片 (SOC) 设计原理/罗胜钦编著. —北京: 机械工业出版社, 2007.7
普通高等教育“十一五”国家级规划教材. 普通高等教育电子信息类规划教材
ISBN 978-7-111-21861-6

I. 系… II. 罗… III. 集成电路—芯片—设计—高等学校—教材 IV. TN402

中国版本图书馆 CIP 数据核字 (2007) 第 103542 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 王保家 版式设计: 张世琴 责任校对: 张晓蓉

封面设计: 张静 责任印制: 洪汉军

北京瑞德印刷有限公司印刷 (三河市明辉装订厂装订)

2007 年 10 月第 1 版第 1 次印刷

184mm × 260mm · 30.25 印张 · 750 千字

标准书号: ISBN 978-7-111-21861-6

定价: 42.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

销售服务热线电话: (010) 68326294

购书热线电话: (010) 88379639 88379641 88379643

编辑热线电话: (010) 88379727

封面无防伪标均为盗版

序

在过去的三十年里，微电子技术一直按照摩尔定律高速发展。目前，工艺水平进入到纳米尺度时代，单芯片集成度达到上亿个元器件，实现了融模拟/数字、软件/硬件于一体的完整应用系统，称为 SOC (System On a Chip)。SOC 技术涉及诸多方面，既包括传统的工艺制造、器件原理、电路原理和数字逻辑等，又包括当代的体系结构设计、软硬件协同设计和设计方法学的内容。因此，系统地介绍 SOC 设计技术和方法，使得不同知识背景的读者能够了解从系统到芯片的各种实现方法和各个层次的设计技术，是十分有价值的。本书正是为了这样的目的，将作者多年的教学经验整理成文，以飨广大读者。

本书的特点可以概括为以下几个方面：

1. 涉及了 SOC 技术的各个层面，包括了系统、模块、逻辑、电路、版图等多方面的设计技术，使得读者能够系统地了解 and 掌握相关技术。

2. 结合了数字逻辑的基本知识，由浅入深地介绍先进的设计方法和理念，方便不同专业领域的读者学习。

3. 以硬件描述语言为核心，结合大量的应用实例，在讲授原理和方法的同时，给读者以主动实践的机会。

目前，在应用需求的推动和政府的大力扶持下，我国集成电路行业的发展正处在上升阶段；相伴随的是企业需要大量的集成电路设计工程师，而现实情况是掌握先进集成电路设计技术的人才缺口越来越大。相信本书的出版能够为更多的学生提供帮助，成为学习和掌握相关课程内容的重要参考资料；同时希望本书能够为有志了解芯片设计技术的相关学科工程技术人员提供帮助，成为原理性和实践性结合的入门指导书籍。

毛志刚

上海交通大学教授

上海交通大学微电子学院院长

前 言

20 世纪下半叶，微电子技术得到了迅速发展。由于集成电路设计和工艺水平的提高，目前集成电路的微细加工工艺的最小尺度已经达到 60nm，可以在一个单片上集成 $10^9 \sim 10^{10}$ 个晶体管，从而有可能将原先由许多集成电路 (Integrated Circuits, 简称 IC) 组成的电子系统集成在一个单片上，构成所谓系统芯片 (System On a Chip, 简称 SOC)。SOC 与 IC 相比，不再是一种功能单一的单元电路，而是将信号采集、处理和输出等完整地系统集成在一起，成为一个有专用目的的电子系统单片。其设计思想也有别于 IC，SOC 把系统的处理机制、模型算法、芯片结构、各层次电路直至器件的设计紧密结合，在一个或若干个单片上完成整个系统的功能。SOC 的出现是电子系统设计领域的一场革命，它对电子信息产业的影响将不亚于 20 世纪 60 年代集成电路的出现所产生的影响。正因为如此，当今电子系统的设计已不再是利用各种通用 IC，进行印制电路板 (Printed Circuits Board, 简称 PCB) 级的设计和调试，而是转向以高密度可编程逻辑器件 (High Density Programmable Logic Device, 简称 HDPLD) 或专用集成电路 (Application Specific Integrated Circuit, 简称 ASIC) 为物理载体的系统芯片的设计。另一方面，由于集成电路工艺的成熟和电子设计自动化 (Electronic Design Automation, 简称 EDA) 工具的迅速发展，使得电子系统的设计者并不需要过多地关注半导体集成工艺，完全可以利用现有的成熟工艺，在 EDA 工具的帮助下完成整个系统从行为层到物理层的全部设计，并最终在 HDPLD 上实现，或委托 IC 制造商进行 ASIC 的生产。

综上所述，系统芯片的出现是电子设计领域的一场革命，同时也对电子工程技术人员提出了新的挑战。SOC 的设计以 IP 核为基础，以硬件描述语言为主要设计手段，借助于以计算机为平台的 EDA 工具进行，其设计方法和手段都与传统的设计不同，要求电子系统设计者必须掌握新的设计方法和技术。近几年来，SOC 在 IT 领域的应用日益广泛，预计若干年后，SOC 将可能占到 IC 全部产量的 50%。因此，今后几年将是 SOC 迅速发展的阶段。

本书共分 10 章。第 1 章 绪论，介绍 SOC 和 EDA 技术的发展概况。第 2 章 CMOS 数字集成电路，介绍 CMOS 集成电路制造的主要工艺、设计规则和工艺参数、CMOS 电路信号传输和延迟、容性负载及其影响和 CMOS 电路的功耗，并介绍典型的 CMOS 数字单元电路。第 3 章 硬件描述语言 VHDL，介绍 VHDL 的基本知识和用 VHDL 语言进行数字系统设计的基本方法；第 4 章 基本数字逻辑单元的设计，介绍用 VHDL 语言进行组合逻辑电路、时序逻辑电路的基本单元模块的设计和状态机的设计。第 5 章 数字系统的层次结构设计，介绍利用层次结构设计的方法来构造大型系统的方法及其仿真测试；第 6 章 介绍 SOC 的体系结构，讨论了嵌入式精简指令集处理器 (RISC)、嵌入式处理器 ARM 和 MIPS32 4Kc 处理器核、片上系统总线和 SOC 的互连机制，以及嵌入式实时操作系统 (RTOS)；第 7 章 可编程逻辑器件，介绍 CPLD、FPGA 等高密度可编程逻辑器件的结构特点和数字系统面向 CPLD 和面向 FPGA 的设计实现；第 8 章 可编程系统芯片，介绍了以 Xilinx 的 Virtex-II Pro 为代表的可编程系统芯片，以及 Virtex-II Pro 内的 32 位嵌入式 RISC 处理器硬核 PowerPC 405 和 32 位嵌入式 RISC 处理器软核 MicroBlaze；第 9 章 专用集成电路的设计，介绍基于单元电

路的版图生成、参数提取、后仿真和版图验证；第 10 章 可测试结构设计，介绍可测试性分析、可测试性结构设计、测试矢量生成、JTAG 标准化测试体系结构设计方法。

本书配有免费电子课件，欢迎选用本书作教材的老师索取，电子邮箱：wbj@mail.machineinfo.gov.cn。

本书中很多电路图源自国外资料，许多图形符号采用的是国外流行符号，请读者注意。

上海交通大学毛志刚教授、同济大学张申科教授在百忙中审阅了全稿，提出了许多宝贵的意见，毛志刚教授还为本书撰写了序，在此表示衷心的感谢。

编著者

目 录

序

前言

第 1 章 绪论 1

1.1 系统芯片是微电子技术发展的必然 1

1.2 电子设计自动化技术和硬件描述语言 3

1.2.1 电子设计自动化技术发展概述 3

1.2.2 Top-Down 设计方法 4

1.2.3 硬件描述语言 6

第 2 章 CMOS 数字集成电路 8

2.1 引言 8

2.2 集成电路的主要生产工艺 8

2.2.1 晶片准备 9

2.2.2 制版 9

2.2.3 光刻 9

2.2.4 氧化 9

2.2.5 淀积 10

2.2.6 腐蚀 10

2.2.7 扩散 10

2.2.8 导体和电阻 10

2.3 CMOS 反相器及其版图 11

2.3.1 MOS 晶体管及其版图 11

2.3.2 CMOS 反相器的结构及其版图 12

2.4 设计规则与工艺参数 16

2.4.1 设计规则的内容与作用 16

2.4.2 几何规则 16

2.4.3 电学规则 17

2.5 CMOS 数字电路的特征 25

2.5.1 标准逻辑电平 25

2.5.2 逻辑扇出特性 25

2.5.3 容性负载及其影响 26

2.5.4 CMOS 电路的噪声容限 26

2.6 CMOS 逻辑门 28

2.6.1 CMOS 或非门 28

2.6.2 CMOS 与非门 30

2.6.3 多输入 CMOS 逻辑门 31

2.7 NMOS 传输晶体管与 CMOS 传输门 31

2.7.1 NMOS 传输晶体管 31

2.7.2 CMOS 传输门 32

习题 32

第 3 章 硬件描述语言 VHDL 34

3.1 引言 34

3.2 VHDL 的基础知识 35

3.2.1 VHDL 程序的结构 35

3.2.2 VHDL 常用资源库中的程序包 42

3.2.3 VHDL 的词法单元 48

3.2.4 数据对象和类型 50

3.2.5 表达式与运算符 57

3.3 VHDL 结构体的描述方式 59

3.3.1 结构体的行为描述 59

3.3.2 结构体的 RTL 描述 61

3.3.3 结构体的结构化描述 63

3.4 结构体的子结构形式 66

3.4.1 进程 66

3.4.2 复杂结构体的多进程组织方法 69

3.4.3 块 70

3.4.4 子程序 72

3.5 顺序语句和并发语句 75

3.5.1 顺序语句 76

3.5.2 并发语句 85

3.6 VHDL 中的信号和信号处理 91

3.6.1 信号的驱动源 91

3.6.2 信号的延迟 92

3.6.3 仿真周期和信号的 δ 延迟 94

3.6.4 信号的属性函数 96

3.6.5 带属性函数的信号 98

3.7 VHDL 的其他语句 103

3.7.1 ATTRIBUTE 描述与定义语句 103

3.7.2 ASSERT 语句 110

3.7.3 TEXTIO 110

3.8 多值逻辑	112	5.4.1 概述	191
3.8.1 三态数值模型	112	5.4.2 仿真程序的设计方法	191
3.8.2 多值逻辑	113	5.4.3 TEXTIO 建立测试程序	195
3.9 元件例化	116	习题	197
3.9.1 设计通用元件	116	第 6 章 SOC 的体系结构	198
3.9.2 构造程序包	119	6.1 SOC 的结构	198
3.9.3 元件的调用	120	6.1.1 引言	198
3.10 配置	121	6.1.2 SOC 的硬件结构	198
3.10.1 默认配置	121	6.1.3 嵌入式软件	200
3.10.2 元件的配置	123	6.2 SOC 中的嵌入式精简指令集处理 器	201
3.10.3 块的配置	126	6.2.1 概述	201
3.10.4 结构体的配置	128	6.2.2 RISC 的定义与特点	202
习题	129	6.2.3 RISC 的指令特点	204
第 4 章 基本数字逻辑单元的设计	131	6.2.4 RISC 的并行处理技术	206
4.1 组合逻辑电路设计	131	6.2.5 RISC/DSP 结构	208
4.1.1 门电路	131	6.2.6 RISC 核的设计	212
4.1.2 三态缓冲器和总线缓冲器	133	6.3 嵌入式处理器 ARM 的体系结构	212
4.1.3 编码器、译码器和选择器	135	6.3.1 概述	212
4.1.4 运算器的设计	138	6.3.2 ARM7 系列处理器	215
4.1.5 算术逻辑运算单元	143	6.3.3 ARM9 系列处理器	222
4.2 时序逻辑电路设计	145	6.3.4 ARM9E 系列处理器	224
4.2.1 触发器	145	6.3.5 ARM10 系列处理器	225
4.2.2 锁存器	149	6.3.6 ARM11 系列处理器	226
4.2.3 寄存器	149	6.4 嵌入式处理器 MIPS32 4Kc 的体系 结构	227
4.2.4 计数器	151	6.4.1 概述	227
4.3 存储器	153	6.4.2 MIPS32 4Kc 嵌入式处理器	229
4.3.1 概述	153	6.5 SOC 的互连机制	236
4.3.2 只读存储器	154	6.5.1 概述	236
4.3.3 随机存取存储器	155	6.5.2 AMBA 总线	238
4.3.4 先进后出堆栈	156	6.5.3 CoreConnect 总线	244
4.4 有限状态机	157	6.5.4 Wishbone 总线	245
习题	162	6.5.5 OCP 总线	246
第 5 章 数字系统的层次结构设计	163	6.5.6 虚拟元件接口	247
5.1 硬件的算法模型	163	6.6 带 ARM 核的嵌入式系统芯片 举例	251
5.1.1 先进先出堆栈的算法模型	163	6.6.1 LPC2100 系列高性能微控制器	251
5.1.2 布思一位补码乘法器的算法 模型	166	6.6.2 AT91SAM7X 系列高性能微 控制器	253
5.2 芯片系统的划分	168	6.6.3 AT91RM9200 高性能微控制 器	257
5.2.1 并行接口 8255	169	6.7 嵌入式实时操作系统	259
5.2.2 布思二位补码乘法器的结构化 设计	175		
5.3 系统间互连的表示	183		
5.4 系统的仿真和测试	191		

6.7.1 实时操作系统	259	8.1 可编程系统芯片概述	350
6.7.2 嵌入式实时操作系统概述	260	8.2 Virtex-II 系列 FPGA 的结构和性能	351
6.7.3 实时多任务调度	261	8.2.1 概述	351
6.7.4 信号与信号量	262	8.2.2 Virtex-II 系列 FPGA 的总体结构	352
习题	263	8.2.3 Virtex-II 系列 FPGA 的可构造逻辑模块	353
第 7 章 可编程逻辑器件	264	8.2.4 18Kbit 可选 RAM 模块	360
7.1 概述	264	8.2.5 嵌入式乘法器	361
7.1.1 可编程逻辑器件的发展	264	8.2.6 全局时钟多路缓冲器	362
7.1.2 用户再构造电路和可编程 ASIC 电路	264	8.2.7 数字时钟管理器	363
7.1.3 可编程逻辑器件的分类	265	8.2.8 输入输出模块	364
7.2 可编程逻辑器件的编程元件	267	8.2.9 有源互连技术	367
7.2.1 熔丝型开关	267	8.3 嵌入式 RISC 处理器软核 MicroBlaze	368
7.2.2 反熔丝开关	267	8.3.1 嵌入式处理器软核 MicroBlaze 概况	368
7.2.3 浮栅编程技术	268	8.3.2 嵌入式处理器软核 MicroBlaze 的结构	369
7.3 PAL 与 GAL 器件的电路结构	270	8.3.3 嵌入式处理器软核 MicroBlaze 的接口信号	370
7.3.1 PLD 的电路表示方法	270	8.4 Virtex-II Pro 系列可编程片上系统芯片	372
7.3.2 PLD 的基本电路结构	273	8.4.1 Virtex-II Pro 系列 SOPC 概况	372
7.3.3 PAL 器件的电路结构	276	8.4.2 嵌入式 PowerPC 405 处理器核	373
7.3.4 通用阵列逻辑 GAL	278	8.4.3 极速双向串行传送器	385
7.4 ispLSI 系列 CPLD	287	习题	390
7.4.1 概述	287	第 9 章 专用集成电路设计	391
7.4.2 ispLSI1000 系列 CPLD 的结构特点	288	9.1 引言	391
7.4.3 ispLSI CPLD 的测试和编程特性	299	9.2 门阵列和门海阵列设计	392
7.4.4 ispLSI 2000 系列 CPLD 的结构	301	9.2.1 门阵列设计	392
7.4.5 ispLSI 3000 系列 CPLD	301	9.2.2 门海阵列	393
7.4.6 ispLSI 5000V 系列 CPLD 的结构和工作原理	306	9.2.3 门阵列和门海阵列的设计流程	395
7.4.7 ispLSI 8000/V 系列 CPLD 的结构和工作原理	313	9.3 标准单元设计	395
7.5 现场可编程门阵列	318	9.4 设计检验	397
7.5.1 概述	318	9.4.1 设计规则检查	398
7.5.2 XC4000 系列 FPGA 的结构和工作原理	320	9.4.2 电学规则检查	398
7.5.3 Spartan 系列 FPGA	344	9.4.3 版图与电路图一致性检查	400
7.6 基于 HDPLD 的系统设计实现	346	9.5 后仿真	401
7.6.1 设计实现概述	346	习题	402
7.6.2 器件的选择	347	第 10 章 可测试性结构设计	403
7.6.3 HDPLD 的设计流程	348	10.1 大规模集成电路可测试设计的	
习题	348		
第 8 章 可编程系统芯片	350		

意义	403	10.3.3 内建自测试技术	424
10.2 可测试性基础	404	10.3.4 系统级测试技术——边界扫描 测试技术	425
10.2.1 故障模型	404	习题	429
10.2.2 可测试性分析	406	附录	430
10.2.3 测试矢量生成	411	附录 A VHDL 标准包集合文件	430
10.2.4 故障模拟	418	附录 B IP 核一览表	460
10.3 集成电路的可测试性结构设计	420	参考文献	473
10.3.1 专门测试设计	421		
10.3.2 扫描测试设计	422		

第 1 章 绪 论

1.1 系统芯片是微电子技术发展的必然

微电子技术是 20 世纪以来发展最为迅速的高新技术。1948 年, 巴丁·布莱连等人发明了世界上第一个晶体管。十年后, 美国科学家在一块硅片上做成了相移振荡器, 该振荡器器件之间的连线不再是传统的导线, 而是通过很薄的刻蚀获得的金属条。这便是世界上第一块集成电路。1967 年诞生了单片集成度在 1000 个晶体管以上的集成电路, 集成电路由中小规模进入大规模时期。20 世纪 70 年代以来, 由于 CMOS 工艺的发展, 尤其是高密度短沟道 CMOS 技术的发展, 使得 CMOS 器件不仅具有很高的集成度, 而且器件的工作速度达到甚至超过了 TTL 电路的速度。1977 年, 美国科学家在 30mm^2 的硅片上集成了 13 万个晶体管, 制作出了世界上第一块 64Kbit 的动态随机存储器 (DRAM), 标志着集成电路跨入了超大规模 (VLSI) 时代。集成电路微细加工的最小尺度几乎每 18 个月缩小 50%, 集成规模每三年增加 4 倍。20 世纪 80 年代, 集成电路加工的最小尺度是微米级, 20 世纪 90 年代初是亚微米 ($0.5 \sim 1\mu\text{m}$), 现在是深亚微米 ($0.3\mu\text{m}$ 以下), $0.25 \sim 0.09\mu\text{m}$ 工艺已成为产品的主导工艺。目前, 最先进的微细加工工艺已达到 60nm。在深亚微米工艺中, 连线引起的延迟已与门电路相当而不可忽略, 传统的电路设计方法必须予以改进。在加工尺度不断细化的同时, 用以生产集成电路的硅片面积不断扩大。现在集成电路生产中大量使用 5in、8in 乃至 12in 的大直径硅片, 单片芯片的面积达到 $2\text{cm} \times 2\text{cm}$ 的尺度, 集成电路已进入 3G 时代 ($1\text{G} = 10^9$), 即单片集成度达到 1G 个晶体管, 工作速度达到 GHz 数量级, 最高数据传输速率达到 Gbit/s 数量级。表 1-1 回顾了集成电路技术发展的历史。预计在本世纪上半叶, 微电子技术仍将保持快速发展的趋势, 并将进入 3T 时代 ($1\text{T} = 10^{12}$)。微细加工的进一步发展有可能达到 $0.01\mu\text{m}$, 这相当于 30 个原子排成一列的长度, 是纳米级加工的水平, 这时, 现有的关于 PN 结的半导体理论可能不再适用, 需要建立新的理论和发展新的技术。

表 1-1 集成电路技术发展简况

年 份	1948	1950	1961	1966	1971	1980	1990	1998	2000
规 模	发明晶体管		分立元件	SSI	MSI	LSI	VLSI	ULSI	GSI
理论集成度 每芯片晶体管				10 ~ 100	10 ~ 1000	1000 ~ 100000	$10^5 \sim 10^6$	$10^6 \sim 10^9$	16×10^9
商业集成度	1		1	10	100 ~ 1000	1000 ~ 20000	20000 ~ 500000	> 500000	> 10^8
代表产品			二极管 晶体管	门电路 触发器	计数器 加法器	8 位微 处理器	16 位、32 位微处理器	图像处理器, SOC 高档微处理器	

由于集成规模的扩大, 原先由许多 IC 组成的电子系统有可能集成在一个单片上, 构成所谓系统芯片 (System On a Chip, 简称 SOC)。系统芯片与集成电路 (IC) 相比, 不再是一

种功能单一的单元电路,而是将信号采集、处理和输出等完整地系统集成在一起,成为一个有某种应用目的的电子系统单片。电子系统传统的设计方法是在 PCB (Printed Circuit Board) 级完成的。系统设计人员利用各 IC 制造商生产的通用集成电路,在 PCB 上构成系统,系统的调试也在 PCB 上进行。这种开发设计方法要求设计者具有丰富的硬件知识和调试能力,产品开发周期长,投资较大,设计修改困难。此外,由于 PCB 连线的延时、空间尺度、重量和可靠性等的制约,整机性能受到很大限制。如果能将整个系统最终集成在一个单片上,无疑对于提高产品性能、缩小产品体积具有极大帮助。因而, SOC 是电子系统开发设计的合理选择。

微电子技术的近期发展成果,为 SOC 的实现提供了多种途径。对于经过验证而又具有批量的系统芯片,可以做成专用集成电路 (Application Specific Integrated Circuit, 简称 ASIC) 大量生产。而对于一些仅小批量应用或处于开发阶段的 SOC,若马上投入流片生产,需要投入较多的资金,承担较大的试制风险。近十几年发展起来的高密度可编程逻辑器件 (High Density Programmable Logic Device, 简称 HDPLD),尤其是近几年迅速发展的可编程系统芯片 (System On Programmable Chip, 简称 SOPC),提供了另一种低成本的实现途径。可编程逻辑器件 (PLD) 是一种由 IC 制造商大批量定型生产的半定制产品,器件内部的逻辑功能由用户设计和构造,是一种价格低廉,但硬件功能可多次编程重构的器件。可编程逻辑器件的出现,使得系统设计人员有可能在不改变系统硬件结构的前提下,修改完善甚至重新设计系统的硬件功能,使电子系统的硬件具有了“柔性”,极大地改变了硬件的刚性结构状态,甚至可以使电子系统的硬件功能动态调整,以适应外界使用环境的变化。

可编程逻辑器件产生于 20 世纪 70 年代,其出现的最初目的是为了用较少的 PLD 品种替代种类繁多的各种中小规模逻辑电路。在 30 多年的发展过程中,PLD 的结构、工艺、功耗和工作速度等性能都得到了重大改进。尤其是在 20 世纪 80 年代,出现了 HDPLD,可编程逻辑器件的单片集成度由原来的数百到数千门,发展到数万、数十万甚至数百万门,芯片的 I/O 引脚也由 20~24 个引脚发展到 400~1000 个引脚,为用户提供了大量的可编程逻辑资源和触发器,可以实现各种逻辑功能 (包括组合逻辑和时序逻辑)。近年来,有的 PLD 制造商还推出了嵌入系统级功能模块的核,使之具有强大功能。因此,完全有可能将一个电子系统集成在一个 HDPLD 单片上,为 SOC 的实现提供了一种简单易行而又成本低廉的手段,这极大地促进了 SOC 的发展。

经过多年的研究,目前关于 SOC 的定义已趋一致。这个定义虽然在形式上不那么严格,但是明确了 SOC 的内涵和表征。一般认为,如果一个集成电路芯片具有如下特征的话,即可称其为 SOC,这些特征是:

- 1) 采用超深亚微米工艺技术实现复杂系统功能的 VLSI;
- 2) 使用一个或多个嵌入式处理器或数字信号处理器 (DSP);
- 3) 具备外部对芯片进行编程的功能;
- 4) 主要采用第三方的 IP (Intellectual Property) 核进行设计。

SOC 的设计理念与传统 IC 不同。SOC 把系统的处理机制、模型算法、芯片结构、各层次电路直到器件的设计紧密结合,在一个或若干个单片上完成整个系统的功能。与普通 IC 的设计不同, SOC 的设计以 IP 核为基础,以硬件描述语言为系统功能的主要描述手段,借助于以计算机为平台的 EDA 工具进行。SOC 的出现是电子设计领域的一场革命。如果说在

20 世纪, 电子系统的设计主要是在 PCB 层次上将各种元器件合理连接, 那么进入 21 世纪后, 电子系统的设计将主要是以 HDPLD 或 ASIC 为物理载体的系统级芯片的设计, 它对电子信息产业的影响将不亚于 20 世纪 60 年代集成电路的出现所产生的影响。

1.2 电子设计自动化技术和硬件描述语言

1.2.1 电子设计自动化技术发展概述

众所周知, 微电子技术和计算机技术是近半个世纪来发展最为迅速的高新技术。微电子技术的发展, 使当今各种电子产品功能完善、系统复杂。一片 VLSI 或 SOC 上往往集成了数十万到数千万个晶体管, 如此规模巨大而又复杂的系统, 其设计和方案的验证也变得十分繁复。用人工方法进行设计, 工作效率十分低下, 有时还难以进行。正是在这种背景下, 人们试图运用计算机辅助设计方法, 建立起完整的电子系统设计、分析、模拟、仿真、综合等手段, 以使整个电子系统的设计能够在计算机平台上自动进行, 促进了电子设计自动化 (Electronic Design Automation, 简称 EDA) 技术的迅猛发展。

层次化设计是设计大型电子系统的常用方法。一个完整的集成电子系统从概念的提出到最后物理实现, 可以分为如下几个设计层次, 即系统级、行为级、寄存器传输级、逻辑门级、电路级和版图级。

版图级也称为物理级, 是集成电路描述的最低层次。在版图级, 以几何图形描述晶体管、MOS 管、二极管、电阻、连线等, 硬件的功能隐含于器件的物理特性关系中。在这一层次, 系统的特性不仅与器件的互连方式有关, 而且与器件的加工工艺有关。

版图级以上的层次是电路级。在电路级, 基本的元器件是晶体管、MOS 管、二极管和电阻、电容等, 系统的功能由电路的微分方程描述。

电路级上面的层次是逻辑门级, 简称为门级, 它是数字系统的主要层次。门级设计中的基本单元是与门、或门、非门、三态门等各种门电路, 以及少量的触发器。门电路的互连方式构成了门级的结构描述, 逻辑图和布尔方程是这一层次的主要描述形式。门级描述也是硬件描述语言最低的层次。

门级之上的层次是寄存器传输级 (Register Transfer Level, 简称为 RTL), 这一层次的描述较之门级更为抽象。在寄存器传输级设计中, 大量采用触发器、寄存器、计数器、多路选择器、算术逻辑运算单元 (ALU) 等功能块。这些功能块的规模比基本门电路单元要大得多, 因而往往又把它们称之为宏单元。寄存器传输级设计的结构描述是宏单元的互连, 反映其功能的是真值表和状态图。

寄存器传输级之上的层次是行为级, 行为级又称为算法级, 这一层次对系统功能的描述是抽象的算法模型, 或者是控制流图和数据流图。

描述电子系统功能的最高层次是系统级, 在这一层次, 系统行为的描述往往是一些性能指标, 如计算机的总线宽度、每秒执行指令的次数、数据的传输速率等。

一般, 把门级以上各层次的设计, 称为前端设计, 而把版图设计以后的工作称为后端设计。前端设计与工艺无关, 而后端设计必须在某一确定的生产工艺下进行。

EDA 技术最初是从物理层的计算机辅助设计开始, 以后逐步向高层次的设计自动化发

展。从 20 世纪 70 年代到目前为止的三十多年中, EDA 技术的发展大致经历了四个阶段。第一代 EDA 工具在 20 世纪 70 年代中期投入实际应用, 主要的功能是 PCB 绘图、版图绘图和图形编辑的计算机辅助设计, 是物理层的 EDA 工具。到了 20 世纪 80 年代中, 推出了第二代 EDA 工具。第二代 EDA 工具除了能够进行物理级的计算机辅助设计之外, 还可以进行逻辑功能仿真、门阵列和标准逻辑单元版图的自动设计和验证, 人机交互采用图形语言, 即电路图输入的方法。20 世纪 80 年代后期, EDA 技术迅速发展, 出现了一批著名的 EDA 工具制造商, 如美国的 Cadence、Mentor、Synopsys、Viewlogic 等, 推出了一大批第三代 EDA 工具。第三代 EDA 工具不仅支持图形输入, 而且支持硬件描述语言输入, 可以对用硬件描述语言设计的寄存器传输级的系统进行功能仿真、时序分析和逻辑综合, 自动生成门级网表文件, 进行门级功能仿真, 并在此基础上进一步进行自动布局布线、时序分析, 生成 PLD 编程所需的 JEDEC (Joint Electron Device Engineering Council, 电子器件工程联合委员会) 文件, 或自动进行版图设计和验证、版图后功能仿真和时序分析, 形成从设计输入 (包括原理图输入、语言输入和状态图输入) 到设计验证 (包括功能仿真、时序仿真、设计规则验证、版图验证、版图后仿真)、到设计实现 (包括优化、映射、布局布线、版图生成) 完整的 EDA 平台。这一代 EDA 工具往往都可提供采用主流工艺制造的 IC 标准单元设计数据库, 供版图设计和后仿真时使用。第三代 EDA 工具是目前使用的主要设计平台。

从 20 世纪 90 年代中期开始, 致力于发展第四代 EDA 工具。第四代 EDA 工具围绕深亚微米工艺特点展开, 试图在行为级对系统进行描述、模拟和综合, 将前端设计和后端设计以及测试融为一体。同时, 研究开发模拟电路设计自动化技术。表 1-2 回顾了 EDA 技术发展概况。如果一个 EDA 工具能够自动完成从系统的行为描述开始、到系统的物理实现为止的全部设计工作, 则称其为全程 EDA 工具。目前, 全程 EDA 技术还在继续发展中。

表 1-2 EDA 技术发展概况

年 代	技术特点	EDA 工具	级 别	典型系统
第一代 (20 世纪 70 年代中)	布局布线设计 PCB 设计	版图图形编辑及绘图 PCB 绘图	物理级	Computer Vision
第二代 (20 世纪 80 年代中)	逻辑模拟和半定制版图设计	电路图输入、逻辑模拟、门阵列、标准单元版图设计及验证	门 级	Mentor Valid
第三代 (20 世纪 90 年左右)	Top-Down 设计实现从寄存器传输级→门级→物理级的设计自动化	VHDL 语言描述、编译及模拟、逻辑综合、时序分析、IC 设计数据库、通用电路分析程序	寄存器传输级	Cadence Mentor Synopsys Spice
第四代 (20 世纪 90 年代中开始)	围绕深亚微米工艺特点展开, 将前端设计和后端设计及测试融为一体	行为级的描述和模拟、高层次的综合、布局布线规划和布图模拟电路设计自动化	系统级	Cadence Mentor Synopsys

1.2.2 Top-Down 设计方法

所谓 Top-Down 设计方法, 就是自顶向下的设计方法, 这是一种从系统的概念出发, 最终到系统的物理实现, 逐步细化的层次化设计方法。这种设计方法与传统的电子系统设计方法不同。

电子系统的传统设计方法中, 首先根据系统的要求, 建立起系统框图, 将整个系统适当

划分, 然后从确定单元电路开始, 沿着单元电路—部件—整机的过程进行样机的设计、制作和调试。系统功能的测试必须待样机完成后 (也就是物理实现后) 才能进行。这种设计是从底层开始, 按照由简到繁、由底向上的步骤进行, 称为 Bottom-Up 设计方法。在设计开始阶段, 对系统的划分、部件功能的定义及相互间的接口必须周密考虑。然而, 由于认识的局限和一些不可预计的因素, 同时由于在样机实现之前, 难以对部件和系统功能进行模拟和仿真, 因此难免在设计过程和样机制作过程中产生一些偏离设计要求的问题, 需要在设计过程和样机试制过程中不断修改完善, 有时甚至推倒重来。显然, 这种设计方法要求系统设计人员必须具备丰富的硬件知识和调试经验, 设计修改比较困难, 产品开发周期长, 投资风险比较大。

为了提高产品研发的效率, 减小投资风险, 现代设计方法不再是从底向上进行, 而是由抽象到具体、由顶向下地进行, 如图 1-1 所示。Top-Down 设计首先从系统的行为层开始,

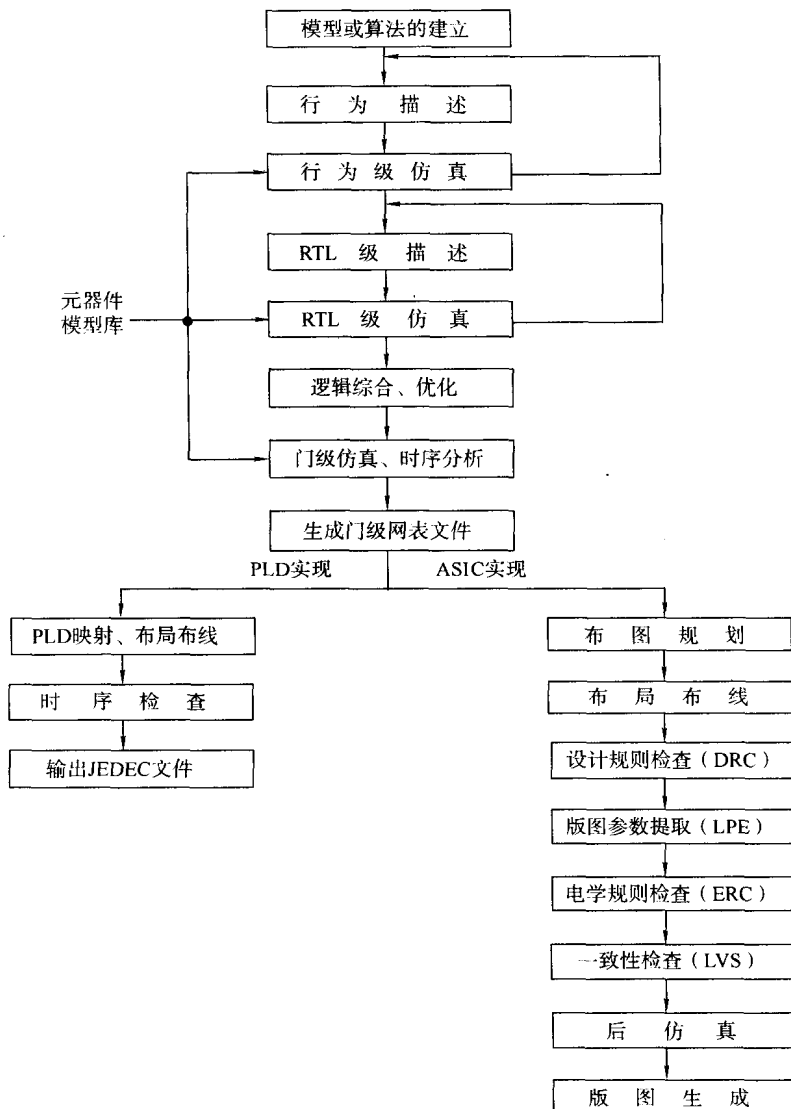


图 1-1 系统芯片设计的 Top-Down 设计方法流程

对系统的模型和算法进行模拟和仿真, 检验系统模型和算法是否正确。一般来说, 目前的 EDA 工具还难以对行为层的描述进行逻辑综合。因此, 在确认系统的模型或算法正确后, 应将系统的行为描述转化成可进行逻辑综合的寄存器传输级 (或者门级) 描述, 同时在这一层次进行模拟、仿真。寄存器传输级描述被验证后, 就可以在计算机平台上由 EDA 工具自动进行逻辑综合和优化, 生成门级网表, 并进行门级仿真和时序分析。上述三个层次的设计工作统称为系统的前端设计。前端设计完成后, 系统的逻辑设计工作全部完成。后端设计是物理层的版图设计, 包括布局、布线、版图参数提取、一致性检查、后仿真和版图生成。在整个设计过程中, 任何一个层次有问题, 都可以及时予以修改和完善。显然, Top-Down 设计中, 采用系统早期仿真, 而系统的物理实现是在设计的结束, 各个层次的模拟和仿真均在以计算机为平台的虚拟样机上进行, 从而有效地降低了研发成本, 缩短了开发周期, 大大节省设计的人力和物力。

1.2.3 硬件描述语言

在利用 EDA 工具进行电子系统设计时, 首先要解决的问题是利用适当的方式描述系统的功能, 并输入计算机, 以便计算机进行识别, 根据设计要求进行有关项目的自动设计。这种描述硬件功能的方式就是硬件描述语言 (Hardware Description Language, 简称为 HDL)。硬件描述语言有文字硬件描述语言 (Word HDL) 和图形硬件描述语言 (Graphic HDL), 图形硬件描述语言即广泛使用的逻辑电路图和状态流程图, 是为广大电子系统设计人员所熟悉的描述方法。在现有的 EDA 工具中, 两种描述方法均可采用。比较而言, 计算机对于图形硬件描述语言的识别比较困难, 同时图形描述难以抽象, 无法描述不同层次的电子系统。而由文字符号构成的形式语言, 已在计算机系统中广泛应用, 相应的编译技术十分成熟, 计算机执行这种由文字符号组成的形式语言程序具有很高的工作效率。故在 EDA 工具中, 文字硬件描述语言应用得更为普遍。一般地说, 硬件描述语言 HDL 都是泛指文字硬件描述语言。本书所指的硬件描述语言也是文字硬件描述语言。

硬件描述语言有多种, 自 20 世纪 80 年代后期起, 逐步开始推广使用。应用比较普遍的有美国国防部提出的 VHDL, Candence 公司开发的 Verilog HDL, Menter Graphics 公司开发的 BLM 硬件描述语言, DATA I/O 公司开发的 ABEL 硬件描述语言等。目前已成为国际标准的硬件描述语言只有两种, 即 VHDL 和 Verilog HDL。

VHDL 的英文全称是 Very high speed integrated circuit Hardware Description Language, 即超高速集成电路硬件描述语言, 是美国国防部为支持超高速集成电路 (Very High Speed Integrated Circuits, 简称 VHSIC) 项目的研发于 20 世纪 80 年代初提出的, 其目的是为了在各个开发商之间统一设计标准, 便于资源共享。1986 年, IEEE 标准化组织开始审订 VHDL, 并于 1987 年 12 月通过标准审查, 宣布实施, 即 IEEE STD 1076—1987 [LRM87]。1993 年 VHDL 作了修订, 形成了新的标准, 即 IEEE STD 1076—1993 [LRM93], 本书所采用的 VHDL 语言即这个标准。

Verilog HDL 最初由 Gate Way Design Automatic (GDA) 公司的 Phil Moorby 首创, 1989 年 Candence 公司收购了 GDA 公司, Verilog HDL 遂成为 Candence 公司 EDA 工具中的硬件描述语言。1995 年, Verilog HDL 成为 IEEE 的标准, 即 Verilog HDL 1364—1995。Verilog HDL

是一种商用硬件描述语言。

VHDL 和 Verilog HDL 都支持从行为级到门级的系统描述，适合于电子系统自顶向下的层次化设计。目前，硬件描述语言正在向模数混合电路设计和系统级描述的方向发展。