

System-on-a-Chip: Design and Test

SoC

设计与测试

[美] Rochit Rajsuman 著

于敦山 盛世敏 田泽 译



北京航空航天大学出版社

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

SoC 设计与测试

System-on-a-Chip: Design and Test

[美] Rochit Rajsuman 著
于敦山 盛世敏 田 泽 译

北京航空航天大学出版社

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

内容简介

本书分为设计和测试两部分,分别介绍 SoC 的设计方法和测试方法。在设计部分,介绍 SoC 设计会遇到的问题和与传统的 ASIC 设计流程的差别,并介绍逻辑核、存储器核及模拟核的设计方法和需要注意的问题,以及 SoC 系统的验证方法。在测试部分,介绍 SoC 中逻辑核、存储器核及模拟核的测试结构与测试方法,还介绍 Iddq 测试在 SoC 测试中的应用,最后介绍产品测试中需要注意的问题。

本书内容全面,可以作为教材,对 ASIC 设计工程师及系统设计工程师都有较高的参考价值。

图书在版编目(CIP)数据

SoC 设计与测试/(美)雷斯曼(Rajsuman, R.)著;
于敦山等译. —北京:北京航空航天大学出版社,2003.8

书名原文: System-on-a-Chip: Design and Test

ISBN 7-81077-308-9

I. S… II. ①雷… ②于… III. 单片微型计算机
—系统设计 IV. TP368.1

本书英文版原名: System-on-a-Chip: Design and Test

Copyright 2000 Advantest America R&D Center, Inc. All rights reserved.

This translation of *System-on-a-Chip: Design and Test* is published by arrangement with
Advantest America R&D Center, Inc.

3201 Scott Boulevard, Santa Clara, CA 95054, USA

本书中文简体字版由美国 Advantest America R&D Center, Inc. 公司授权北京航空航天大学出版社在中华人民共和国境内(不包括香港特别行政区)独家出版发行。版权所有。

北京市版权局著作权登记号:图字:01-2002-5038

SoC 设计与测试

System-on-a-Chip: Design and Test

[美] Rochit Rajsuman 著

于敦山 盛世敏 田 泽 译

责任编辑 王 实

责任校对 陈 坤

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:(010)82317024 传真:(010)82328026

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

E-mail: bhpress@263.net

北京市松源印刷有限公司印装 各地书店经销

*

开本: 787×1092 1/16 印张: 14 字数: 358 千字

2003 年 8 月第 1 版 2003 年 8 月第 1 次印刷 印数: 5 000 册

ISBN 7-81077-308-9 定价: 35.00 元

译者序

随着设计与制造技术的发展,集成电路设计从晶体管的集成发展到逻辑门的集成,现在又发展到 IP 的集成,即 SoC(System-on-a-Chip)设计技术。SoC 可以有效地降低电子/信息系统产品的开发成本,缩短开发周期,提高产品的竞争力,是未来工业界将采用的最主要的产品开发方式。

虽然 SoC 一词多年前就已出现,但到底什么是 SoC 则有各种不同的说法。在经过了多年的争论后,专家们就 SoC 的定义达成了一致意见。这个定义虽然不是非常严格,但明确地表明了 SoC 的特征:

- 实现复杂系统功能的 VLSI;
- 采用超深亚微米工艺技术;
- 使用一个以上嵌入式 CPU/数字信号处理器(DSP);
- 外部可以对芯片进行编程;
- 主要采用第三方 IP 进行设计。

从上述 SoC 的特征来看,SoC 中包含了微处理器/微控制器、存储器以及其他专用功能逻辑,但并不是包含了微处理器、存储器以及其他专用功能逻辑的芯片就是 SoC。SoC 技术被广泛认同的根本原因,并不在于 SoC 可以集成多少个晶体管,而在于 SoC 可以用较短时间被设计出来。这是 SoC 的主要价值所在——缩短产品的上市周期,因此,SoC 更合理的定义为:SoC 是在一个芯片上由于广泛使用预定制模块 IP(Intellectual Property)而得以快速开发的集成电路。从设计上来说,SoC 就是一个通过设计复用达到高生产率的硬件-软件协同设计的过程。从方法学的角度来看,SoC 是一套极大规模集成电路的设计方法学,包括 IP 核可复用设计/测试方法及接口规范、系统芯片总线式集成设计方法学、系统芯片验证和测试方法学。

虽然 SoC 的概念已被提出多年,并且关于 SoC 设计、验证及测试方法的研究取得了很大的进展,但是仍存在很多尚未解决的问题,因此,SoC 仍是集成电路设计方法学领域里的一大热点。对于很多有经验的 IC 设计工程师来说,SoC 设计仍然是一个具有挑战性的任务。我国的 IC 产业界和学术界也有很多人在讨论并设计 SoC,但是在有关 SoC 的一些问题上,我们的认识还不够全面,对有些概念还比较模糊。为此,北京大学信息科学技术学院一些从事 SoC 设计方法研究与教学的师生翻译了本书——《SoC 设计与测试》一书。

本书介绍 SoC 系统设计方法/流程、系统验证方法、测试方法及 IP 可复用设计和测试方法,并介绍 SoC 设计过程中会遇到的问题及解决方法。本书内容系统、全面,可作为硕士研究生 SoC 设计相关课程的参考书,对 ASIC 设计工程师和系统设计工程师也有较高的参考价值。

本书由于敦山、盛世敏及田泽主译。参加本书翻译的还有北京大学微电子学系 SoC 研究所的研究生杨红、李夏青、陈佳、桂少华、张怡浩和张师群。

译 者

2003 年 4 月

序

关于本书

这个项目开始时只是作为一个临时性的报告,目的是就片上系统(SoC: System-on-a-Chip)设计、测试的主要问题以及通用的工业惯例与 Advantest 公司内的各个部门进行交流。在近一年的时间里,许多人协力完成了这个报告。

在这段时间,我还参与了 VSI 联盟关于 SoC 设计和测试的各种规范文献的撰写,也参与了 IEEE P1500 工作组制定 SoC 核测试标准的工作。在工作过程中,我注意到 SoC 的概念已广为人知,同时在业界也有着很多误解,从错误的用语到完全的概念性错误。显然,出版这篇临时报告会对业界有所帮助。

基于这种想法,我联系了 Artech House 出版社。Artech House 出版社的编辑们已经读过很多关于 SoC 方面的书,并且对这篇报告很感兴趣。考虑到科技发展的速度,我制定了一个四个月的时间表,并决定在 1999 年底之前完成原稿。虽然我有这篇报告的原始资料,但并不是简单地整理一下就行了。除了要从报告中删除一些部分外,还有很多部分甚至是整个章节都需要全面的检查或者重写。此外,还需要再增加几章。由于时间仓促以及其他原因,我一度感到非常乏味和疲劳。这可能导致了书中有几部分讨论不够完备。我曾根据 ARD 的同事和 Artech House 的评论家们的反馈意见加以弥补,但读者也许还会在书中找到较多的漏洞。

本书的目的是提供一个 SoC 设计和测试技术现状的概貌。由于我想介绍的是关于 SoC 设计和测试的基本问题,所以尽量避免讨论一般的 VLSI 设计和测试问题,所讲的内容也是专门适用于 SoC 的。SoC 现在正处于发展的最初阶段,因此本书介绍的知识决不是完全的。本书可以划分为两个独立的部分:(1)设计;(2)测试。

在第一部分设计的第 1 章绪论中,介绍了 SoC 产生的背景及相关术语的定义,还讨论了 SoC 设计中的困难。软-硬件协同设计、设计复用及核是 SoC 的基本问题。在第 2 章从产品定义(说明)到交付的要求与系统集成的角度讨论了这些题目。这些方法中有一些已经在少数公司使用,其他的部分正在由其他公司和标准化组织评估。对于设计的复用,需要一套严格的 RTL 规则。附录 A 包括了 RTL 编码的参考规则,还有基于 Lint 的对这些准则的检查方法。

第 2 章的内容只是针对数字逻辑核的。第 3 章讨论了使用大规模片上存储器的优点和相关问题,以及使用存储器编译器设计存储器核的方法,并给出了一些常用的模拟/混合信号核的规范,如 DAC、ADC 和 PLL。第 4 章介绍了单个的核和整个 SoC 的设计确认,还提出了开发核和 SoC 的测试平台(testbench)的方针。第一部分的最后是

第 5 章,给出了核、核的连接和 SoC 的几个具体例子。

第二部分的第 6 章讨论了测试的困难。SoC 的主要组成部分之一是数字逻辑核,因此,在第 6 章讲解了嵌入式数字逻辑核的测试方法学。与数字逻辑核的设计方法类似,有些测试方法已经在少数公司使用,其他部分正在由其他公司和标准化组织评估。第 6 章还提供了微处理器和微控制器核的测试方法。这些核可以看成是数字逻辑核,但是,由于它们的体系结构和功能的特点,它们又是 SoC 的大脑,因而,有少数内容超出了一般逻辑核的范畴,是专门用于微处理器和微控制器核的。这些内容也在第 6 章讲述。

除了逻辑核,大的存储器块是 SoC 的又一个重要组成部分。第 7 章讨论了嵌入式存储器的测试。嵌入式模拟和混合信号电路的测试在第 8 章讨论。

Iddq 测试一直在引起广泛的注意。除了与制造相关的问题外,SoC 的 Iddq 测试还有其他一些特殊的问题。第 9 章讨论了这些问题以及 Iddq 的可测性设计和测试向量的生成方法。

许多关于 SoC 测试的重要问题是与 SoC 的制造环境和产品测试相关的。这些问题包括全速测试、多测试仪的测试后勤学以及生产线的一般问题,如材料传送、快速分级和生产流程等。第 10 章讨论了这些问题。最后,在第 11 章中给出了结论。

致 谢

首先,我要向 Artech House 的编辑们表示感谢,感谢他们快速的反应、热诚、充满活力的工作和对事务完美的处理。特别要感谢 Mark Walsh, Barbara Lovenvirth, Jessica McBride, Tina Kolb, Bridget Maddalena, Sean Flannagan 和 Lynda Fishbourne。同时还要感谢 Artech House 的评论家们阅读本书的草稿并提出了宝贵的意见。

不必说,还要感谢 ARD 的很多以不同方式帮助我完成本书的人们。如果没有 Shigeru Sugamori, Hiro Yamoto 及 Robert Sauer 的持续支持与鼓励,这本书是不会面世的。我特别想向 Robert Sauer 致谢,他花费了大量的晚上和周末的时间对本书进行校对,并及时给我反馈意见。这些帮助对于改正书中的许多错误和遗漏起了不可估价的作用。他和 Artech House 的评论家提出的宝贵意见,帮我改正了原书中的很多不足。

我还要对所有从事 SoC 设计和测试的人们表示感谢,没有他们的工作,就不可能写出这本书。我要感谢 VSI 联盟,他们制定了一系列 SoC 设计和测试的规范。还要感谢的是 IEEE P1500 工作组正在进行的工作和 IEEE 与 Computer Society Press 的出版工作。最后还要感谢 IEEE 允许我引用各种论文中大量的图表。

目 录

第一部分 设计

第 1 章 绪 论

1.1 当前 SoC 的结构	4
1.2 SoC 设计中的问题	7
1.3 硬件-软件协同设计	11
1.3.1 协同设计流程	12
1.3.2 协同设计工具	15
1.4 核库、EDA 工具和网址	16
1.4.1 核 库	17
1.4.2 EDA 工具和提供商	18
1.4.3 网上站点	22
参考文献	23

第 2 章 逻辑核的设计方法

2.1 SoC 设计流程	25
2.2 设计复用的一般原则	27
2.2.1 同步设计	27
2.2.2 存储器和混合信号设计	28
2.2.3 片上总线	29
2.2.4 时钟分配	29
2.2.5 清零/置位/复位信号	31
2.2.6 物理设计	31
2.2.7 可交付模型	32
2.3 软核和固核的设计流程	33
2.3.1 设计流程	33
2.3.2 软核/固核的开发流程	34
2.3.3 RTL 设计规则	35
2.3.4 软核/固核产品化	35
2.4 硬核设计流程	36
2.4.1 硬核设计中的特有问题的	36
2.4.2 硬核开发流程	37

2.5 交付检查表与可交付的核	38
2.5.1 交付检查表	38
2.5.2 软核交付	39
2.5.3 硬核交付	40
2.6 系统集成	40
2.6.1 使用硬核设计	40
2.6.2 使用软核设计	41
2.6.3 系统验证	41
参考文献	42

第3章 存储器与模拟核的设计方法

3.1 使用大容量的嵌入式存储器的原因	43
3.2 嵌入式存储器的设计方法	46
3.2.1 电路技术	46
3.2.2 存储器编译器	52
3.2.3 仿真模型	54
3.3 模拟电路的技术要求	55
3.3.1 模/数转换器	55
3.3.2 数/模转换器	58
3.3.3 锁相环	59
3.4 高速器件	60
3.4.1 Rambus ASIC 单元	60
3.4.2 IEEE 1394 串行总线(Firewire)PHY 层	61
3.4.3 高速 I/O	61
参考文献	63

第4章 设计的确认

4.1 核级确认	64
4.1.1 核的确认方案	64
4.1.2 测试平台	66
4.1.3 核级时序验证	68
4.2 核接口的验证	70
4.2.1 协议验证	70
4.2.2 门级仿真	71
4.3 SoC 的设计确认	71
4.3.1 协同仿真	72
4.3.2 硬仿真	75
4.3.3 硬件原型	76
参考文献	77

第 5 章 核及 SoC 设计实例

5.1 微处理器核	78
5.1.1 V830R/AV 超标量 RISC 核	82
5.1.2 PowerPC 603e G2 核的设计	83
5.2 关于存储器核生成器	85
5.3 核的集成和片上总线	85
5.4 SoC 设计实例	87
5.4.1 媒体处理器	87
5.4.2 机顶盒 SoC 系统的可测性	91
参考文献	92

第二部分 测 试**第 6 章 数字逻辑核的测试**

6.1 SoC 测试问题	95
6.2 访问、控制及隔离	96
6.3 IEEE P1500 的成果	98
6.3.1 无边界扫描的核	99
6.3.2 核测试语言	101
6.3.3 带有边界扫描的核	102
6.4 核测试和 IP 保护	107
6.5 用于设计复用的测试方法	109
6.5.1 核可测性的方针	109
6.5.2 高层次测试综合	109
6.6 微处理器核的测试	110
6.6.1 内建自测试方法	110
6.6.2 举例:ARM 处理器核的可测性	114
6.6.3 对微处理器核的调试支持	115
参考文献	116

第 7 章 嵌入式存储器的测试

7.1 存储器的故障模型和测试算法	118
7.1.1 故障模型	118
7.1.2 测试算法	120
7.1.3 测试算法的有效性	121
7.1.4 用多数据背景来修改测试	122
7.1.5 多端口存储器时的修改	123
7.1.6 用于双缓冲存储器的算法	123

7.2 嵌入式存储器的测试方法	125
7.2.1 用 ASIC 功能测试方法进行测试分析	125
7.2.2 直接访问的测试应用	125
7.2.3 扫描寄存器或环绕寄存器的测试应用	126
7.2.4 存储器内建自测试	126
7.2.5 通过片上微处理器进行测试	130
7.2.6 嵌入式存储器测试算法小结	132
7.3 存储器的冗余和修复	133
7.3.1 硬修复	135
7.3.2 软修复	135
7.4 检错和纠错编码	135
7.5 含大型嵌入式存储器的 SoC 的生产测试	136
参考文献	137

第 8 章 模拟和混合信号核的测试

8.1 模拟参数及特性	140
8.1.1 数/模转换器	140
8.1.2 模/数转换器	142
8.1.3 锁相环	145
8.2 用于模拟核的可测性设计和内建自测试方法	147
8.2.1 Fluence Technology 公司的模拟 BIST 方案	147
8.2.2 LogicVision 公司的模拟 BIST 方案	149
8.2.3 通过片上微处理器进行测试	150
8.2.4 IEEE P1149.4	152
8.3 特殊模拟电路的测试	154
8.3.1 Rambus ASIC 单元	154
8.3.2 1394 串行总线/Firewire 的测试	154
参考文献	157

第 9 章 Iddq 测试

9.1 物理缺陷	159
9.1.1 桥接(短路)	159
9.1.2 栅氧化缺陷	163
9.1.3 开路(断线)	163
9.1.4 Iddq 测试的有效性	166
9.2 SoC 中 Iddq 测试的困难	168
9.3 基于 Iddq 测试的设计	172
9.4 Iddq 测试设计规则	176
9.5 Iddq 测试向量的产生	177

参考文献	180
第 10 章 生产测试	
10.1 生产测试流程	183
10.2 全速测试	184
10.2.1 RTD 和无效周期	184
10.2.2 Fly-by	186
10.2.3 速度分类	187
10.3 产能和材料传送	188
10.3.1 测试后勤	188
10.3.2 测试仪器设置	189
10.3.3 多 DUT 测试	189
参考文献	190
第 11 章 总结与结论	
11.1 总 结	191
11.2 未来的前景	193
附录 A 设计复用的 RTL 指导原则	
A.1 命名习惯	194
A.2 编码的一般指导原则	195
A.3 面向综合的 RTL 开发	196
A.4 RTL 检查	197
作者简介	
中英文名词对照表	

第一部分

设计

- 第1章 绪论
- 第2章 逻辑核的设计方法
- 第3章 存储器与模拟核的设计方法
- 第4章 设计的确认
- 第5章 核及 SoC 设计实例

第 1 章 绪 论

在 20 世纪 90 年代中期,ASIC 技术从芯片集成的理念发展成为基于嵌入式核的片上系统(SoC)概念。简单地说,SoC 是一种将多个独立的 VLSI 设计拼合在一起,来完成某一应用所需的全部功能的集成电路。在 SoC 的定义中,明确地强调了服务于多种应用的称为“核”(也可称为知识产权模块、虚拟单元和宏单元)的预先设计的复杂功能模型。在一个 SoC 芯片中,ASIC 制造商可以使用自己设计的核库,也可以使用来自无加工线(fabless)或者无芯片(chipless)设计公司设计的核。后者通常被称为知识产权(IP)公司。今天 SoC 的设计方案主要有以下 3 种形式^[1]:

1. 由 ASIC 销售商设计 这种方式指芯片所有部件的设计和生产都由 ASIC 销售商完成。
2. 集成设计 这种方式指由 ASIC 销售商设计的芯片,但其中的部件不全是由他们自己设计的。这意味着使用了从其他途径,如核/IP 提供商或其他半导体生产厂商提供的一个或多个核。芯片的生产既可以是 ASIC 销售商,也可以是其他半导体厂商来完成。
3. 桌面设计 这种设计方式指由 fabless 公司设计的芯片。其中大部分核通过如 IP 设计公司、EDA 公司、设计服务公司或者半导体生产厂商等途径来获取。通常情况下,芯片的生产由独立的芯片代工厂完成。

由于 SoC 芯片内核的集成数量越来越多,以及嵌入式软件的使用,使 SoC 设计的复杂性极大地增加了,而且预计还会继续以很高的速率增加。这种趋势如图 1.1 所示。

按照摩尔定律,芯片的集成度每 3 年增加 4 倍。由于核规模的增大和工艺几何尺寸的缩小使芯片的复杂性增加了,因而使设计规范中必须包括更多的参量。例如,几年前,芯片的设计仅仅考虑功能、延迟、功耗和可测性就足够了。现在,对信号完整性、电迁移、封装影响、电磁耦合和射频效应的考虑变得越来越必要了。

除了硅基的 IP 核复杂性增加之外,嵌入式软件的容量也以大大高于摩尔定律的速度增加。因此,整个系统复杂性的增长趋势要比硅基的 IP 核复杂性的增长趋势明显得多。

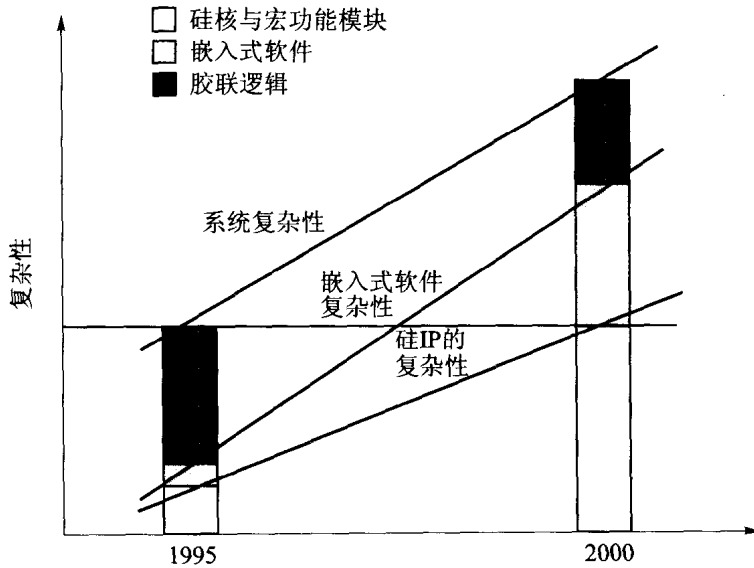


图 1.1 设计复杂性的增长趋势

1.1 当前 SoC 的结构

在所有 SoC 的设计中,预先设计的核是基本的部件。系统芯片由各种满足片上功能的嵌入核组合而成。这些核包括微处理器、大型存储器阵列、音频和视频控制器、调制解调器、因特网调谐器、2 维和 3 维的图像控制器、DSP 功能模块等。这些核通常可以由可综合的高级描述语言(HDL),如 Verilog/VHDL,或经过优化的晶体管级的版图,如 GDSII 来实现。核的形式决定了核使用的灵活性。下面给出软核、固核和硬核的定义^[1~3]:

- **软核** 是用可综合的 RTL 描述或者通用库元件的网表形式表示的可复用模块。这意味着软核(宏单元)的用户须负责实际的实现和版图。
- **固核** 是指在结构和拓扑方面针对性能和面积通过版图规划,甚至可能用某种工艺技术进行过优化的可复用模块。它们以综合好的代码或通用库元件的网表形式存在。
- **硬核** 是指在性能、功率和面积上经过优化,并映射到特定工艺技术的可复用模块。它们以完整的布局布线后的网表和诸如 GDSII 格式的固定版图形式存在。

软核、固核及硬核间的权衡要依据可复用性、灵活性、可移植性、性能优化、成本及面市时间等进行综合考虑。图 1.2 提供了这种权衡的量化表示。

基于核的 SoC 设计实例,包括最新的高端微处理器、多媒体处理器、GPS 控制器、单片蜂窝电话、GSM 电话及智能呼机 ASIC,甚至单片计算机。注意,有些人并不把微处理器归到 SoC 芯片之列。但是,无论从哪个角度讲,像 Alpha 21264, Power PC,

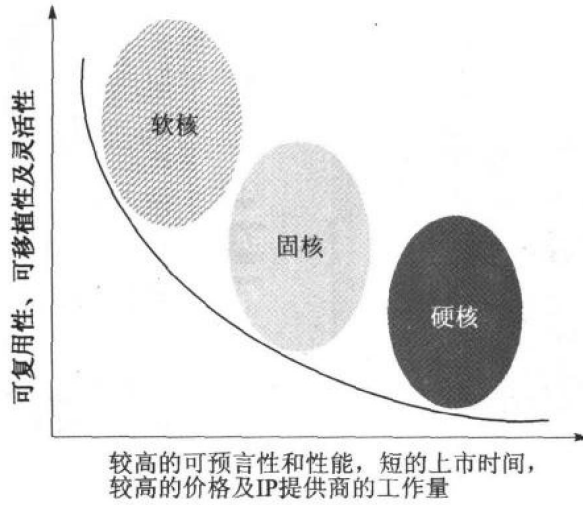


图 1.2 软核、固核及硬核间的权衡

Pentium III 这些微处理器的结构和设计的复杂性绝不亚于 SoC 芯片。

为了帮助读者理解 SoC 的一般结构，图 1.3 给出了一个高端微处理器的示例。图 1.4 是两个 SoC 设计示例。这两个图中都给出了最新设计的 SoC 芯片所使用的各种部件类型。

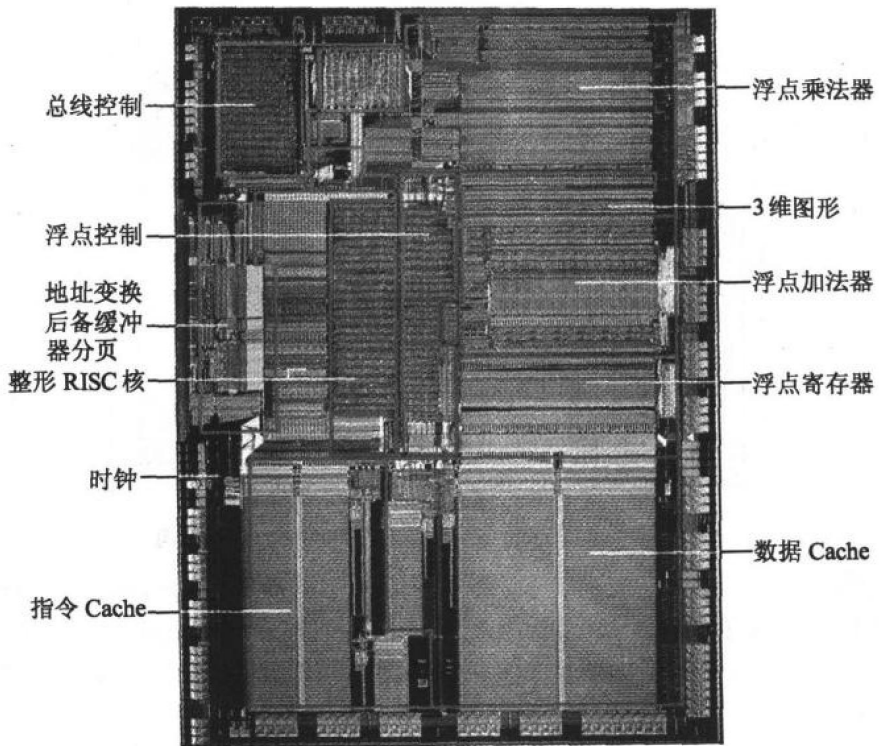


图 1.3 Intel 的 i860 微处理器
(来自文献[4], © IEEE 1989, 并被授权复制)