

# 《计算机组成与设计（原书第5版）》

## 图书基本信息

书名：《计算机组成与设计（原书第5版）》

13位ISBN编号：9787111504828

出版时间：2015-7-1

作者：戴维 A.帕特森 (David A.Patterson), 约翰 L.亨尼斯 (John L.Hennessy)

页数：536

译者：王党辉, 康继昌, 安建峰

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu111.com](http://www.tushu111.com)

# 《计算机组成与设计（原书第5版）》

## 内容概要

《计算机组成与设计：硬件/软件接口（原书第5版）》是计算机组成与设计的经典畅销教材，第5版经过全面更新，关注后PC时代发生在计算机体系结构领域的革命性变革——从单核处理器到多核微处理器，从串行到并行。本书特别关注移动计算和云计算，通过平板电脑、云体系结构以及ARM（移动计算设备）和x86（云计算）体系结构来探索和揭示这场技术变革。

与前几版一样，本书采用MIPS处理器讲解计算机硬件技术、汇编语言、计算机算术、流水线、存储器层次结构以及I/O等基本功能。

《计算机组成与设计：硬件/软件接口（原书第5版）》特点

更新例题、练习题和参考资料，重点关注移动计算和云计算这两个新领域。

涵盖从串行计算到并行计算的革命性变革，第6章专门介绍并行处理器，每章中都涉及并行硬件和软件的相关主题。

全书采用Intel Core i7、ARM Cortex-A8和NVIDIA Fermi GPU作为实例。

增加“运行更快”这一新实例，说明正确理解硬件技术的重要性，它能使软件性能提高200倍。

讨论并强调计算机体系结构的“8个伟大思想”——通过并行提高性能、通过流水线提高性能、通过预测提高性能、面向摩尔定律的设计、存储器层次、使用抽象简化设计、加速大概率事件和通过冗余提高可靠性。

# 《计算机组成与设计（原书第5版）》

## 作者简介

David A. Patterson 加州大学伯克利分校计算机科学系教授，美国国家工程研究院院士，IEEE和ACM会士，曾因成功的启发式教育方法被IEEE授予James H. Mulligan, Jr教育奖章。他因为对RISC技术的贡献而荣获1995年IEEE技术成就奖，而在RAID技术方面的成就为他赢得了1999年IEEE Reynold Johnson信息存储奖。2000年他和John L. Hennessy分享了John von Neumann奖。

John L. Hennessy 斯坦福大学校长，IEEE和ACM会士，美国国家工程研究院院士及美国科学艺术研究院院士。Hennessy教授因为在RISC技术方面做出了突出贡献而荣获2001年的Eckert-Mauchly奖章，他也是2001年Seymour Cray计算机工程奖得主，并且和David A. Patterson分享了2000年John von Neumann奖。

## 书籍目录

出版者的话

本书赞誉

译者序

前言

作者简介

第1章 计算机概要与技术1

1.1 引言1

1.1.1 计算应用的分类及其特性2

1.1.2 欢迎来到后PC时代3

1.1.3 你能从本书学到什么4

1.2 计算机系统结构中的8个伟大思想6

1.2.1 面向摩尔定律的设计6

1.2.2 使用抽象简化设计6

1.2.3 加速大概率事件6

1.2.4 通过并行提高性能7

1.2.5 通过流水线提高性能7

1.2.6 通过预测提高性能7

1.2.7 存储器层次7

1.2.8 通过冗余提高可靠性7

1.3 程序概念入门7

1.4 硬件概念入门10

1.4.1 显示器11

1.4.2 触摸屏12

1.4.3 打开机箱12

1.4.4 数据安全15

1.4.5 与其他计算机通信16

1.5 处理器和存储器制造技术17

1.6 性能20

1.6.1 性能的定义20

1.6.2 性能的度量22

1.6.3 CPU性能及其因素23

1.6.4 指令的性能24

1.6.5 经典的CPU性能公式25

1.7 功耗墙27

1.8 沧海巨变：从单处理器向多处理器转变29

1.9 实例：Intel Core i7基准31

1.9.1 SPEC CPU基准测试程序31

1.9.2 SPEC功耗基准测试程序32

1.10 谬误与陷阱33

1.11 本章小结35

1.12 历史观点和拓展阅读36

1.13 练习题36

第2章 指令：计算机的语言40

2.1 引言40

2.2 计算机硬件的操作43

2.3 计算机硬件的操作数44

2.3.1 存储器操作数45

- 2.3.2 常数或立即数操作数47
- 2.4 有符号数和无符号数48
- 2.5 计算机中指令的表示53
- 2.6 逻辑操作58
- 2.7 决策指令60
  - 2.7.1 循环61
  - 2.7.2 case/switch语句63
- 2.8 计算机硬件对过程的支持64
  - 2.8.1 使用更多的寄存器66
  - 2.8.2 嵌套过程67
  - 2.8.3 在栈中为新数据分配空间69
  - 2.8.4 在堆中为新数据分配空间70
- 2.9 人机交互72
- 2.10 MIPS中32位立即数和寻址75
  - 2.10.1 32位立即数75
  - 2.10.2 分支和跳转中的寻址76
  - 2.10.3 MIPS寻址模式总结78
  - 2.10.4 机器语言解码79
- 2.11 并行与指令：同步81
- 2.12 翻译并执行程序83
  - 2.12.1 编译器83
  - 2.12.2 汇编器84
  - 2.12.3 链接器85
  - 2.12.4 加载器87
  - 2.12.5 动态链接库87
  - 2.12.6 启动一个Java程序89
- 2.13 以一个C排序程序作为完整的例子90
  - 2.13.1 swap过程90
  - 2.13.2 sort过程91
- 2.14 数组与指针96
  - 2.14.1 用数组实现clear96
  - 2.14.2 用指针实现clear97
  - 2.14.3 比较两个版本的clear97
- 2.15 高级内容：编译C语言和解释Java语言98
- 2.16 实例：ARMv7(32位)指令集98
  - 2.16.1 寻址模式99
  - 2.16.2 比较和条件分支100
  - 2.16.3 ARM的特色100
- 2.17 实例：x86指令集102
  - 2.17.1 Intel x86的改进102
  - 2.17.2 x86寄存器和数据寻址模式103
  - 2.17.3 x86整数操作105
  - 2.17.4 x86指令编码107
  - 2.17.5 x86总结108
- 2.18 实例：ARMv8(64位)指令集108
- 2.19 谬误与陷阱109
- 2.20 本章小结110
- 2.21 历史观点和拓展阅读111
- 2.22 练习题112

## 第3章 计算机的算术运算117

- 3.1 引言117
- 3.2 加法和减法117
- 3.3 乘法121
  - 3.3.1 顺序的乘法算法和硬件121
  - 3.3.2 有符号乘法124
  - 3.3.3 更快速的乘法124
  - 3.3.4 MIPS中的乘法124
  - 3.3.5 小结125
- 3.4 除法125
  - 3.4.1 除法算法及其硬件结构125
  - 3.4.2 有符号除法128
  - 3.4.3 更快速的除法128
  - 3.4.4 MIPS中的除法129
  - 3.4.5 小结129
- 3.5 浮点运算130
  - 3.5.1 浮点表示131
  - 3.5.2 浮点加法135
  - 3.5.3 浮点乘法138
  - 3.5.4 MIPS中的浮点指令139
  - 3.5.5 算术精确性145
  - 3.5.6 小结146
- 3.6 并行性和计算机算术：子字并行148
- 3.7 实例：x86中流处理SIMD扩展和高级向量扩展149
- 3.8 加速：子字并行和矩阵乘法150
- 3.9 谬误与陷阱153
- 3.10 本章小结155
- 3.11 历史观点和拓展阅读158
- 3.12 练习题159

## 第4章 处理器162

- 4.1 引言162
- 4.2 逻辑设计的一般方法165
- 4.3 建立数据通路167
- 4.4 一个简单的实现机制173
  - 4.4.1 ALU控制173
  - 4.4.2 主控制单元的设计175
  - 4.4.3 为什么不使用单周期实现方式181
- 4.5 流水线概述182
  - 4.5.1 面向流水线的指令集设计186
  - 4.5.2 流水线冒险186
  - 4.5.3 对流水线概述的小结191
- 4.6 流水线数据通路及其控制192
  - 4.6.1 图形化表示的流水线200
  - 4.6.2 流水线控制203
- 4.7 数据冒险：旁路与阻塞206
- 4.8 控制冒险214
  - 4.8.1 假定分支不发生215
  - 4.8.2 缩短分支的延迟215
  - 4.8.3 动态分支预测216

- 4.8.4 流水线小结220
- 4.9 异常221
  - 4.9.1 MIPS体系结构中的异常处理221
  - 4.9.2 在流水线实现中的异常222
- 4.10 指令级并行226
  - 4.10.1 推测的概念227
  - 4.10.2 静态多发射处理器227
  - 4.10.3 动态多发射处理器231
  - 4.10.4 能耗效率与高级流水线233
- 4.11 实例：ARM Cortex-A8和Intel Core i7流水线234
  - 4.11.1 ARM Cortex-A8235
  - 4.11.2 Intel Core i7 920236
  - 4.11.3 Intel Core i7 920的性能238
- 4.12 运行更快：指令级并行和矩阵乘法240
- 4.13 高级主题：通过硬件设计语言描述和建模流水线来介绍数字设计以及更多流水线示例242
- 4.14 谬误与陷阱242
- 4.15 本章小结243
- 4.16 历史观点和拓展阅读243
- 4.17 练习题243
- 第5章 大容量和高速度：开发存储器层次结构252
  - 5.1 引言252
  - 5.2 存储器技术255
    - 5.2.1 SRAM技术256
    - 5.2.2 DRAM技术256
    - 5.2.3 闪存258
    - 5.2.4 磁盘存储器258
  - 5.3 cache的基本原理259
    - 5.3.1 cache访问261
    - 5.3.2 cache缺失处理265
    - 5.3.3 写操作处理266
    - 5.3.4 一个cache的例子：内置FastMATH处理器267
    - 5.3.5 小结269
  - 5.4 cache性能的评估和改进270
    - 5.4.1 通过更灵活地放置块来减少cache缺失272
    - 5.4.2 在cache中查找一个块275
    - 5.4.3 替换块的选择276
    - 5.4.4 使用多级cache结构减少缺失代价277
    - 5.4.5 通过分块进行软件优化280
    - 5.4.6 小结283
  - 5.5 可信存储器层次283
    - 5.5.1 失效的定义283
    - 5.5.2 纠正一位错、检测两位错的汉明编码（SEC/DED）284
  - 5.6 虚拟机287
    - 5.6.1 虚拟机监视器的必备条件289
    - 5.6.2 指令集系统结构（缺乏）对虚拟机的支持289
    - 5.6.3 保护和指令集系统结构289
  - 5.7 虚拟存储器290
    - 5.7.1 页的存放和查找293
    - 5.7.2 缺页故障294

- 5.7.3 关于写297
  - 5.7.4 加快地址转换：TLB297
  - 5.7.5 集成虚拟存储器、TLB和cache 300
  - 5.7.6 虚拟存储器中的保护302
  - 5.7.7 处理TLB缺失和缺页303
  - 5.7.8 小结307
  - 5.8 存储器层次结构的一般框架309
    - 5.8.1 问题1：一个块可以被放在何处309
    - 5.8.2 问题2：如何找到一个块310
    - 5.8.3 问题3：当cache缺失时替换哪一块311
    - 5.8.4 问题4：写操作如何处理311
    - 5.8.5 3C：一种理解存储器层次结构行为的直观模型312
  - 5.9 使用有限状态机来控制简单的cache314
    - 5.9.1 一个简单的cache314
    - 5.9.2 有限状态机315
    - 5.9.3 一个简单的cache控制器的有限状态机316
  - 5.10 并行与存储器层次结构：cache一致性317
    - 5.10.1 实现一致性的基本方案318
    - 5.10.2 监听协议319
  - 5.11 并行与存储器层次结构：冗余廉价磁盘阵列320
  - 5.12 高级内容：实现cache控制器320
  - 5.13 实例：ARM Cortex-A8和Intel Core i7的存储器层次结构320
  - 5.14 运行更快:cache分块和矩阵乘法324
  - 5.15 谬误和陷阱326
  - 5.16 本章小结329
  - 5.17 历史观点和拓展阅读329
  - 5.18 练习题329
- 第6章 从客户端到云的并行处理器340
- 6.1 引言340
  - 6.2 创建并行处理程序的难点342
  - 6.3 SISD、MIMD、SIMD、SPMD和向量机345
    - 6.3.1 在x86中的SIMD：多媒体扩展346
    - 6.3.2 向量机346
    - 6.3.3 向量与标量的对比347
    - 6.3.4 向量与多媒体扩展的对比348
  - 6.4 硬件多线程350
  - 6.5 多核和其他共享内存多处理器352
  - 6.6 图形处理单元简介355
    - 6.6.1 NVIDIA GPU体系结构简介356
    - 6.6.2 NVIDIA GPU存储结构357
    - 6.6.3 GPU展望358
  - 6.7 集群、仓储级计算机和其他消息传递多处理器360
  - 6.8 多处理器网络拓扑简介363
  - 6.9 与外界通信：集群网络366
  - 6.10 多处理器测试集程序和性能模型366
    - 6.10.1 性能模型368
    - 6.10.2 Roofline模型369
    - 6.10.3 两代Opteron的比较370
  - 6.11 实例：评测Intel Core i7 960和NVIDIA Tesla GPU的Roofline模型373



- 6.12 运行更快：多处理器和矩阵乘法376
- 6.13 谬误与陷阱378
- 6.14 本章小结379
- 6.15 历史观点和拓展阅读381
- 6.16 练习题382
- 附录A 汇编器、链接器和SPIM仿真器389
- 附录B 逻辑设计基础437
- 索引494

# 《计算机组成与设计（原书第5版）》

## 精彩短评

- 1、感觉跟《性能设计》各有千秋
- 2、常见句式：“本节的剩余内容在配套网站上。”
- 3、写得比 CS:APP 清楚明了很多。
- 4、翻译之糟糕，把原本的好书弄得艰涩难懂，习题中出现的大量翻译错误问题更是让人无语。希望对于此类书籍的翻译更加严谨一些！
- 5、与时俱进是好事，但这版那么多东西不印出来，做在线资源合适么。人家荷兰老头津津乐道的各种小历史、小发展、小趣事，这里去的连影子都没有，确实，读起来就不显得划水啰嗦，只是连raid都去掉了。不过看完后终于明白，为啥mips如此适合教学了。这书例子确实易懂，然而习题完全变了个画风。

-----  
另外，如果想精读cpu设计部分，去看原版，中文版插图以及排版简直可怕，铁工机真是名不虚传。

# 《计算机组成与设计（原书第5版）》

## 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:[www.tushu111.com](http://www.tushu111.com)