

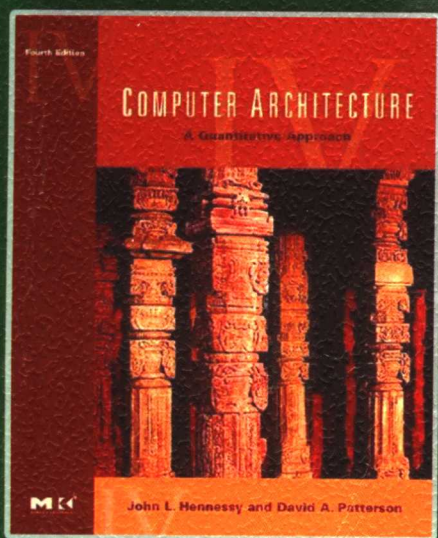


权威作者 经典教材

# 计算机系统结构

—— 量化研究方法 (第四版)

Computer Architecture:  
A Quantitative Approach, Fourth Edition



[美] John L. Hennessy 著  
David A. Patterson

白跃彬 译 钱德沛 审校



电子工业出版社  
Publishing House of Electronics Industry  
<http://www.phei.com.cn>

## 内 容 简 介

本书堪称计算机系统结构学科的“圣经”，是计算机体系结构方向的学生们的必读教材。全书系统地介绍了计算机系统的设计基础、指令集系统结构、流水线与指令级并行技术、层次化存储系统与存储设备、互连网络以及多处理器系统等重要内容。在这一最新版本中，作者更新了从单核处理器到多核处理器的历史发展过程的相关内容，同时使用了广受好评的“量化研究方法”进行计算设计，并阐述了多种可以实现并行的技术，这些技术恰恰是展现多处理器系统结构威力的关键。在介绍多处理器时，作者不仅讲述了处理器的性能，而且还介绍了处理器性能之外的其他设计要素，包括功耗、可靠性、可用性和可信性等。

本书可作为计算机专业计算机系统结构方向的高年级本科生及研究生的教材，也可以作为相关技术人员的参考书。

Authorized translation from the English language edition published by Elsevier Science(USA). Copyright © 2007 by Elsevier Science(USA).

Translation Copyright © 2007 by Publishing House of Electronics Industry.

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 the Publisher.

本书中文简体专有翻译出版权由 Elsevier Science(USA) 授予电子工业出版社。其原文版权及中文翻译出版权受法律保护。未经许可，不得以任何形式或手段复制或抄袭本书内容。

版权贸易合同登记号 图字：01-2006-7674

### 图书在版编目 (CIP) 数据

计算机系统结构：量化研究方法：第4版 / (美) 亨尼西 (Hennessy, J. L.) 等著；白跃彬译. - 北京：电子工业出版社，2007.8

(国外计算机科学教材系列)

书名原文：Computer Architecture: A Quantitative Approach, Fourth Edition

ISBN 978-7-121-04769-5

I. 计... II. ①亨... ②白... III. 计算机体系结构-教材 IV. TP303

中国版本图书馆 CIP 数据核字 (2007) 第 114263 号

责任编辑：谭海平

印 刷：北京市海淀区四季青印刷厂

装 订：涿州市桃园装订有限公司

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：32.5 字数：915 千字

印 次：2007 年 8 月第 1 次印刷

定 价：69.80 元 (附光盘 1 张)

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

服务热线：(010) 88258888。

## 出版说明

21世纪初的5至10年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入WTO后的今天,培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如Pearson Education培生教育出版集团、麦格劳-希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

## 教材出版委员会

- 主任** 杨芙清 北京大学教授  
中国科学院院士  
北京大学信息与工程学部主任  
北京大学软件工程研究所所长
- 委员** 王 珊 中国人民大学信息学院院长、教授
- 胡道元 清华大学计算机科学与技术系教授  
国际信息处理联合会通信系统中国代表
- 钟玉琢 清华大学计算机科学与技术系教授、博士生导师  
清华大学深圳研究生院信息学部主任
- 谢希仁 中国人民解放军理工大学教授  
全军网络技术研究中心主任、博士生导师
- 尤晋元 上海交通大学计算机科学与工程系教授  
上海分布计算技术中心主任
- 施伯乐 上海国际数据库研究中心主任、复旦大学教授  
中国计算机学会常务理事、上海市计算机学会理事长
- 邹 鹏 国防科学技术大学计算机学院教授、博士生导师  
教育部计算机基础课程教学指导委员会副主任委员
- 张昆藏 青岛大学信息工程学院教授

多处理器时代的到来已经势不可挡。当我们告别单核处理器时代进入芯片多处理时代之际，Hennessy和Patterson著作最新版本的推出无疑是逢时之作。很少有其他著作能像本书一样对计算机系统结构产生过如此巨大的影响，而这一最新版本必定在未来很长一段时间内是这一领域的权威之作。

—— Luiz André Barroso, Google 公司

如果你问下列哪个可算做经典：甲壳虫乐队，HP 计算器，巧克力曲奇饼，还是这本书？我会告诉你，它们都很经典，因为它们都经受住了时间的考验！

—— Robert P. Colwell, Intel 公司首席架构师

这本书不仅提供了一个应当被计算机系统架构师了熟于心的权威参考，同时也对业界的趋势进行了积极探索。

—— Krisztián Flautner, ARM 公司

青出于蓝而胜于蓝！新版的内容更新后，与当今计算机系统结构领域的核心问题更加密切相关。而且，其中新的练习题对学生和老师都更加实用了。

—— Norman P. Jouppi, HP 实验室

本书以导致 RISC 技术革命的基本原理为基础，其中包括导致 CISC 转换的因素。现在，在这个新版本中，它清晰地解释并阐述了最近出现的新一代多线程多核处理器所需的微系统结构技术。

—— Marc Tremblay, Fellow & VP, 首席架构师, Sun 微系统公司

这本著作具备了优秀教科书的所有优点：它介绍的计算机组织和设计的观点和技术，符合教学法，易于接受；它的内容吸引人；同时还涉及到非常多的领域。该书的第一版堪称内容精良的典范，而最新版又再次做到了这一点。

—— Milos Ercegovac, 加州大学洛杉矶分校

他们再次做到了这一点。Hennessy 和 Patterson 有力地证明了为什么他们会成为这个博大精深却又日新月异的领域中的泰斗。谬误：计算机系统结构在信息时代并非必不可少。陷阱：你不需要本书。

—— Michael D. Smith, 哈佛大学

Hennessy 和 Patterson 再次做到了这一点！第四版是一次经典重现，通过调整，它很好地应对了由所谓‘后 CMOS 时代’技术不断变化而带来的一些制约条件。书中对真实产品所做的详细的

案例分析对教学非常有益,而且其行文流畅让人爱不释手。这本书对学生和专业人士等人来说不容错过。

—— Pradip Bose, IBM 公司

本书最新版向学生们讲述了系统结构的框架和基础知识,这些都是未来的优秀架构师所必需的。

—— Ravishankar Iyer, Intel 公司

随着技术的进步,设计方面机遇与约束条件的变化,使得这本书也需要与时俱进。第四版秉承以前的风格,在呈现了商业产品革新的同时描述了以下一些基本的概念:高级处理器和存储器系统设计技术,多线程和芯片多处理器,存储系统,虚拟机,等等。该书对于那些想要学习实际商业产品所用的系统结构概念的人来说是绝佳的资源。

—— Gurindar Sohi, 威斯康星大学麦迪逊分校

非常高兴能够让我的学生使用这本好书来学习计算机系统结构方面的知识,不过,我也因为自己没有能写出这样的好书而羡慕该书的作者。

—— Mateo Valero, UPC, 巴塞罗那

Hennessy和Patterson继续根据当前计算机系统设计领域的变化对他们的教学方式进行了改进。学生们由此得以深入理解影响计算机系统结构设计因素以及计算机系统领域潜在的研究方向。

—— Dan Connors, 科罗拉多大学布德尔分校

经过再版,本书仍将是以后计算机系统结构领域学生们的必读书目。

—— Wen-mei Hwu, 伊利诺伊大学香槟分校

第四版秉承了其一贯采用的内容精挑细选、方法新颖的风格,这非常受学生、研究人员以及计算机系统设计者们的欢迎。新版中的课程对读者将一直都会有很大的帮助。

—— David Brooks, 哈佛大学

在第四版中, Hennessy 和 Patterson 使得本书回归到突出个别重点的做法,因而第四版再次像第一版一样经典。

—— Mark D. Hill, 威斯康星大学麦迪逊分校

## 计算机系统结构相关公式

1. CPU 时间 = 指令数 × 每条指令的时钟周期数 × 时钟周期所占时间

2. Amdahl 定律 (阿姆达尔定律):

$$\text{总加速比} = \frac{\text{原执行时间}}{\text{改进后执行时间}} = \frac{1}{(1 - \text{可改进任务占任务总量的百分比}) + \frac{\text{可改进任务占任务总量的百分比}}{\text{改进部分加速比}}}$$

3. 动态功率 =  $1/2 \times$  电容性负载  $\times$  电压的平方  $\times$  交换频率

4. 静态功率 = 静态电流  $\times$  电压

5. 平均存储器访问时间 = 命中时间 + 缺失率  $\times$  缺失代价

6. 有效性 = 平均失败时间 / (平均失败时间 + 平均修复时间)

7. 晶片产量 = 晶片成品率  $\times \left(1 + \frac{\text{单位面积缺陷} \times \text{晶片面积}}{\alpha}\right)^{-\alpha}$

式中, 晶片成品率表示因已经报废而无须测试的晶片数,  $\alpha$  表示掩膜层数, 是影响到晶片产量的重要参数 (通常  $\alpha = 4.0$ )

8. 每条指令缺失数 = 缺失率  $\times$  每条指令存储器访问数

9. Cache 索引空间:  $2^{\text{index}} = \text{Cache 容量} / (\text{块大小} \times \text{组相关性})$

10. 算数平均 (AM)、加权平均 (WAM) 和几何平均 (GM):

$$\text{AM} = \frac{1}{n} \sum_{i=1}^n \text{Time}_i, \quad \text{WAM} = \sum_{i=1}^n \text{Weight}_i \times \text{Time}_i, \quad \text{GM} = \sqrt[n]{\prod_{i=1}^n \text{Time}_i} = \exp\left(\frac{1}{n} \times \sum_{i=1}^n \ln(\text{Time}_i)\right)$$

式中,  $\text{Time}_i$  是  $n$  个程序中第  $i$  个的执行时间,  $\text{Weight}_i$  是第  $i$  个程序的权重。

$$11. \text{几何标准偏差} = \exp\left(\sqrt{\frac{\sum_{i=1}^n \ln(\text{Time}_i) - \ln(\text{几何平均})^2}{n}}\right)$$

## 经验规律

1. Amdahl/Case 定律: 要使计算机系统较为平衡, 每 MIPS 的 CPU 性能需要对应 1 MB 的主存容量和 1 Mb/s 的 I/O 带宽。

2. 90/10 局部性原理: 一个程序中 10% 的代码执行其 90% 的指令。

3. 带宽定律: 带宽的增长速度至少是其时延性能提高速度的两倍。

4. 2:1 Cache 定律: 大小为  $N$  的直接映射 Cache 的缺失率大致和大小为  $N/2$  的双路组相连 Cache 缺失率相同。

5. 可靠性定律: 进行毫无错误的设计。

## MIPS 64 常用指令子集

指令类型 / 操作码	指令含义
数据传输指令	在寄存器和存储器间,或在整型和浮点型或专用寄存器间传输数据;只有存储器地址模式是16位位移量+通用寄存器的值
LB, LBU, SB	装载一个字节(到整型寄存器),装载一个无符号字节(到整型寄存器),(从整型寄存器)存储一个字节
LH, LHU, SH	装载一个半字(到整型寄存器),装载一个无符号半字(到整型寄存器),(从整型寄存器)存储一个半字
LW, LWU, SW	装载一个字(到整型寄存器),装载一个无符号字(到整型寄存器),(从整型寄存器)存储一个字
LD, SD	装载一个双字(到整型寄存器),(从整型寄存器)存储一个双字
L.S, L.D, S.S, S.D	装载一个单精度浮点数,装载一个双精度浮点数,存储一个单精度浮点数,存储一个双精度浮点数
MFC0, MTC0	在通用寄存器和专用寄存器之间复制数据
MOV.S, MOV.D	复制一个单精度或双精度浮点数到另一个浮点寄存器
MFC1, MTC1	在浮点寄存器和整型寄存器之间复制32位数据
算术 / 逻辑指令	在GPR中对整型和逻辑数据进行操作;溢出会引发有符号算术陷阱
DADD, DADDI, DADDU, DADDIU	加,加立即数(所有的立即数为16位);有符号和无符号数
DSUB, DSUBU	减,有符号和无符号
DMUL, DMULU, DDIV, DDIVU, MADD	乘和除,有符号和无符号;乘-加;所有操作数占用和生成64位值
AND, ANDI	与,和立即数相与
OR, ORI, XOR, XORI	或,和立即数相或,异或,和立即数异或
LUI	装载高位立即数,即装载保存立即数的寄存器的32-47位,然后进行符号扩展
DSLL, DSRL, DSRA, DSLLV, DSRLV, DSRAV	移位:立即数(DS_) and 变量(DS_V)形式;包括逻辑左移、逻辑右移、算术右移
SLT, SLTI, SLTU, SLTIU	有符号小于置位、有符号小于立即数置位;无符号小于置位、无符号小于立即数置位;
控制指令	条件转移和跳转;基于PC或通过寄存器
BEQZ, BNEZ	根据GPR内容等于/不等于0进行转移;16位位移量从PC+4开始
BEQ, BNE	根据GPR内容等于/不等于进行转移;16位位移量从PC+4开始
BCIT, BCIF	测试浮点数状态寄存器中的比较位然后转移,16位位移量从PC+4开始
MOVN, MOVZ	如果第三个GPR为负、0,将GPR的内容复制到另一个GPR中
J, JR	根据从PC+4开始的26位位移量(J)或寄存器中的目标(JR)跳转
JAL, JALR	跳转并链接:将PC+4保存到R31,转移目标是相对的PC(JAL)或某个寄存器(JALR)
TRAP	根据一个向量式地址转移到操作系统
ERET	从一个异常中返回到用户代码;恢复用户模式
浮点运算指令	在双精度和单精度格式上进行浮点操作
ADD.D, ADD.S, ADD.PS	双精度浮点数加,单精度浮点数加,一对单精度浮点数相加
SUB.D, SUB.S, ADD.PS	双精度浮点数减,单精度浮点数减,一对单精度浮点数相减
MUL.D, MUL.S, MUL.PS	双精度浮点数乘,单精度浮点数乘,一对单精度浮点数相乘
MADD.D, MADD.S, MADD.PS	双精度浮点数乘加,单精度浮点数乘加,一对单精度浮点数乘加
DIV.D, DIV.S, DIV.PS	双精度浮点数除,单精度浮点数除,一对单精度浮点数相除
CVT.__	转换指令:CVT.x.y从x类型转换到y类型,x和y是L(64位整型)、W(32位整型)、D(DP)或S(SP)。两个操作数都是FPR
C._.D, C._.S	双精度和单精度比较:“_”=LT,GT,LE,GE,EQ,NE;将浮点数状态寄存器中的相应位置位



# 译 者 序

我第一次看到美国 John L. Hennessy 教授和 David A. Patterson 教授所著的《计算机系统结构——量化研究方法（第一版）》一书是在 1991 年，当时，我正在德国汉诺威大学计算机系统结构和操作系统研究所做访问学者。我立刻被这本书的崭新内容吸引住了，恨不得立即将它据为己有，一“读”为快。可惜，此书当时实验室里只有一本，教授要用它备课，博士生要用它当参考书，粥少僧多，不能独占。想咬咬牙，自己买一本吧，谁知在汉诺威的书店里竟寻觅不到它的踪迹。当时，Internet 还没有像今天这样普及，网上购书更是没有的概念，所以只能插空，陆陆续续地读，直到 1992 年春回国，我也未能把它读完。

1992 年夏，周松年教授回国访问来到西安。当时正值他从多伦多大学拿到博士学位不久，刚刚自己创立了 Platform 公司。在他的交谈中，我提到本书，流露出渴求的意思。过了一段时间，忽然接到松年从加拿大寄来的包裹，打开一看，正是渴望已久的书，当时的激动和喜悦至今难忘。于是，这本书成了我炫耀的宝贝，当然，在我的教学和科研生涯中，它发挥了莫大的作用。

再往后，本书出了第二版、第三版，又有了中译版，国内的条件也越来越好，我的那本第一版已不再是奇货可居。但是，我对它的感情却始终没变。

本书堪称是计算机系统结构学科的“圣经”。有多少学子在它的引导下进入了计算机系统结构的殿堂，从一无所知，到略知一二，到较为精通。本书不但有精辟的论述，还有大量习题，帮助学生理解、掌握书中所讲授的知识。多年来，本书在本科生和研究生教学中发挥的作用是不可估量的。更为难能可贵的是，两位作者笔耕不辍，每一次再版，都有新内容，给人以新的启示和教诲。这恐怕也是本书长盛不衰的原因。

此次第四版问世，更是给人以耳目一新的惊喜。全书内容经过重新安排，大量章节经过了重写。作者全面更新了与单核处理器到多核处理器历史发展相关的内容，以其独特的量化研究方法，阐述了多种体现多处理器系统结构优势的并行技术，这种“与时俱进”的态度令人肃然起敬。中国正处在研究开发我们自己的高性能通用处理器、向研制千万亿次高效能计算机冲刺的关键时期，本书第四版及其中译本的问世对正在从事这方面研究的中国学者来说，无疑是一个福音。

我与本书译者相识多年。他在师从西安交通大学郑守淇先生攻读计算机系统结构博士学位期间，就深受本书的影响。此次第四版中译本的问世，反映了他对本书的偏爱，也凝聚了他的很多心血。感谢他为读者做了一件大好事。我相信，严肃的读者定能从阅读本书中得到新的启示，吸取新的营养，在自己的事业中作出新的贡献。

钱德沛

2007 年 7 月 1 日于北京

## 作者简介

**John L. Hennessy:** 斯坦福大学校长, 1977年开始在斯坦福大学电子工程系和计算机系任教。他是IEEE和ACM会士, 美国国家工程院院士及美国科学与艺术院院士。由于其在RISC技术领域的杰出贡献, 于2001年被授予Eckert-Mauchly奖; 他获得的其他奖项还包括2001年度Seymour Cray计算机工程奖以及2000年度同David Patterson共同获得的John von Neumann奖。同时他还获得了荣誉博士学位。

1981年, Hennessy带领几个研究生开始其在斯坦福大学的MIPS项目。在1984年完成该项目之后, 他从学校离开了1年时间去开发一个MIPS计算机系统, 在开发该系统的过程中研制出了世界上第一个商用的RISC微处理器。MIPS Technologies公司从1991年被SGI公司收购, 后来在1998年脱离出来成为一个独立的公司, 并将其战略重点转向嵌入式微处理器。截至2006年, 已经有超过5亿个MIPS微处理器被用于视频游戏、掌上电脑、激光打印机和网络交换机等设备中。

**David A. Patterson:** 1977年开始在加州大学伯克利分校任职, 并讲授计算机系统结构课程。因其在教育方面的杰出贡献, 他荣获了Upsilon Pi Epsilon的Abacus奖、加州大学杰出教育奖、ACM颁发的Karlstrom奖、Mulligan教育奖章以及IEEE颁发的本科生教育奖。Patterson曾由于其对RISC技术的杰出贡献而获得IEEE技术贡献奖, 另外, 由于他在RAID技术方面的成就, 他赢得了IEEE Johnson信息存储奖。之后, 他和John L. Hennessy一起获得了John von Neumann奖。Patterson是美国科学与艺术院院士以及IEEE和ACM会士, 同时, 他被选入美国国家工程院、美国国家科学院以及硅谷工程名人堂。Patterson服务于美国总统信息技术顾问委员会, 同时也是伯克利大学EECS系计算机科学分部的主任, 计算研究协会主席以及ACM主席。这使他获得了CRA的杰出服务奖。

在加州大学伯克利分校, Patterson领导了RISC的设计与研发工作, 这几乎是第一台VLSI精简指令系统计算机。该研究工作成为后来SPARC系统结构的基础, SPARC系统结构目前被Sun微系统公司、富士公司及其他公司使用。他曾是RAID项目的领导者之一, 该项目使得许多公司得以应用其在可靠存储系统方面的研究成果。同时, 他还曾参加过网络工作站(NOW)项目, 成功研究出了目前许多Internet公司在使用的集群技术。上述这些项目获得了ACM颁发的三个论文奖。Patterson当前的研究项目在RAD实验室进行, 主要开发可靠、自适应、用于分布式Internet服务的相关技术; 另外, 还有多处理器研究加速器(RAMP)项目, 目的是开发基于FPGA的低成本、高可靠的并行计算机和开源硬件与软件。

# 序 言

Fred Weber, MetaRAM 公司总裁兼 CEO

我非常荣幸能够为这本在计算机系统结构领域极为重要的著作的第四版作序。在第一版中,我在产业界的第一位老师 Grodon Bell 就预言,该书将成为在计算机系统结构和设计领域中处于中心位置的一本经典著作。他的观点非常正确。我清晰地记得在该书刚出版时自己那种激动的心情。如今重读这本在后续三个版本中不断丰富的著作,我再次感受到了兴奋的感觉。坦率地说,在计算机系统结构领域中,甚至在任何我所了解的领域中,没有一本著作能够以这样迅速有效的方式使读者对计算机系统结构的知识从陌生到精通。

这本书数据翔实,图表清晰,理论与实践相结合,实例和描述丰富。该书用到了非常多的缩写、技术用语、趋势描述、公式、示例图和表格。这些对于计算机系统结构方面的著作来说再恰当不过。系统结构设计者的角色不像科学家或发明家那样,需要去深入研究一个特殊的现象或者创造新的基础性的材料或技术;也不像系统实施者那样,需要精通一些工具来精确地构建系统。系统结构设计者要做的工作应该是这样的:他们首先需要深刻理解最新的技术和做法,以及从过去到现在所期望的设计风格,然后他们需要拥有一种设计理念来和谐地构造完整的系统,最后要拥有信心和精力来合理搭配这些知识与资源从而完成构建工作。要做到这些,一名系统结构设计者需要大量的信息,同时深入理解基本概念并掌握定量的研究方法以便作为基础。而这正是这本书所要讲述的。

随着计算机系统的演变——从大型机、小型机和微处理器并存的时代,到以微处理器为主的时代,再到当前微处理器引入大型机的复杂技术的时代——Hennessy 和 Patterson 不断适时地更新这本著作。这本书的第一版介绍了 IBM 360, DEC VAX 以及 Intel 80x86, 这些是那一代计算机的典型代表。后续版本主要讨论了 80x86 和 RISC 处理器的细节,这在当时是处于统治地位的。最近的这一版本,讨论了线程与多处理技术、虚拟化与存储器层次结构以及存储系统,这为读者提供了一个与当前发展方向相吻合的背景,同时也为后十年的设计奠定了基础。这其中着重讨论了 AMD Opteron 和 SUN Niagara, 这是新时代多处理和 SoC 系统结构中, x86 与 SPARC(RISC)系统结构的典型代表,同时,它们也是创新的科技思想在真实的商业产品中的实际运用。

本书的第 1 章,用很少的篇幅介绍了计算机设计技术的分类以及计算机系统结构的基本概念,概述了驱动产业界发展的技术趋势,并给出了一种定量的方法帮助计算机系统设计。后两章主要讨论了传统 CPU 的设计,并深入讲述了该核心领域的一些可能方法和限制因素。最后三章主要讨论与多处理技术、存储器层次结构和存储系统相关的系统问题。这些领域的知识对于计算机系统结构设计者来说是非常重要的,同时在当前片上系统设计时代,这些知识对每一个 CPU 设计者来说是必需的。最后,在附录中详细阐述了一些具体实例来帮助加深理解。

在设计工作中,同时以宏观和微观两种角度来分析问题并在这两种角度之间灵活转换的能力非常重要。在阅读本书时,读者会频繁地发现这两种角度。一个伟大的系统结构设计成果,无论是计算机系统设计,还是建筑设计,抑或是教科书的设计,都要考虑客户的需求和期望并给出一个设计成果,使得客户能够满意地说:“哇,我不知道这样也是可行的。”本书在这一点上做得非常成功,我希望它可以给读者带来快乐和价值。

# 前 言

## 关于此书的目的

从这本书的第一版开始一直到现在的第四版,我们都一直致力于对未来科技发展所基于的最基本原则进行阐述。我们对于计算机系统结构的热情从来都没有消退过,这里我们想引用第一版当中的一句话来说明:“这并不是什么徒有其表的纸模型,让人提不起兴趣。恰恰相反,这是一门研究人员相当感兴趣的学科,它需要在市场和成本性能之间找到平衡点。在人们寻找的过程中,既有可圈可点的失败案例,也有令人称赞的成功案例。”

在写这本书的第一版时,我们的初衷是希望改变人们对计算机系统结构原有的认知方式。现在,我们觉得这一初衷依然有存在的需要,而且相当重要。这一领域的变化日新月异,对它的研究必须基于真实计算机的真实案例和测量数据,简单地罗列一些永远都无法实现的定义和设计是毫无意义的。我们热情地欢迎我们的老朋友,同样也欢迎那些新加入的新朋友。不论怎样,我们都承诺对真实的系统采用相同的量化研究和分析方法。

和之前的版本一样,我们竭尽所能使这本书能够满足不论是专业工程师、系统架构师,还是那些参加高级计算机系统结构和设计课程的人的需求。与之前版本类似的还有一点,那就是本书希望能够通过强调成本、性能、功耗之间的关系以及优秀的工程设计来把看似复杂的计算机系统结构解释清楚。我们相信这一领域正在日趋成熟,而且会像其他经典的理工学科一样,将建立起强大的量化研究基础。

## 关于第四版

《计算机系统结构——量化研究方法》的第四版可能是其第一版以来最具特殊意义的一版。就在我们刚刚开始编写这一版不久,Intel宣布它将和IBM和Sun一起研究单芯片上多处理器或多核所能实现的高性能设计。以下是这本书的第一组数字,在连续十六年保持每十八个月性能双倍提升的纪录之后,单核处理器性能改良的频率降到了每年幅度不大的提高。在计算机系统结构发展历程上的这一转折点意味着有史以来,第一次没有人能够给出速度更快的串行处理器。如果希望程序能够运行得更快,比如,对新增加的功能进行调整,那么就必须对程序进行并行处理。

因此,除了和前三个版本一样,继续关注指令级并行(ILP)所带来的高性能表现之外,这一版还对线程级并行(TLP)和数据级并行(DLP)给予了同样的关注。在之前的版本中,也有一些关于大型多处理器服务器上的TLP和DLP的内容,但是这一版会在单芯片的多核处理器中提到TLP和DLP。这一历史性的转变让我们相应更改了各个章节的顺序:在上一版本中第6章是关于多处理器的内容,在这一版本中将这一内容提前到第4章中来讲述。

不断发展的技术也促使我们将一些原本在稍后章节中的内容提前到第1章中。按照科学家们的预言,随着计算机产业向具有65nm或更小特性的半导体处理器的方向发展,将导致更高的硬件和软件出错率。于是,我们决定将有关依赖性的一些基本概念从第三版中的第7章前移到第1章。除此之外,由于电源成为决定在一个芯片上能够放置多少内容的决定性因素,我们在第1章中增加了相关部分的内容。当然,所有章节的内容和举例都做了更新,下面将一一给出解释。

在这一版中,除了因技术领域的不断革新而带来内容的大量更新之外,我们还采用了新的方法来编写书中的习题。要编写出有趣、准确而且无歧义的习题是一件难度大、耗费时间的工作,而且这些习题必须要能够测试到一个章节中方方面面的内容。令人无奈的是,网络的使用使得这些习题的可用期减少了一半,最多只有几个月。学生不用自己费劲地找出答案,只需要在网络上搜索一下就能找到一本刚刚出版的教材上的习题的答案。于是,我们所付出的巨大努力很快便付之东流。对教师而言,他们也无法利用这些习题来检测学生对知识的掌握程度。

为了能够较好地解决这一问题,在这一版本中我们尝试了两个新的做法。第一,我们在学术界和业界聘请了各个主题的专家来编写这些习题。也就是说,每个领域的一些顶尖人物在帮助我们通过生动活泼的方式来探讨每个章节中的核心概念,并检测读者对于内容的理解程度。第二,每组习题都围绕一组范例分析展开。我们希望每个范例分析中的量化举例都能够在很长一段时间里保持其生命力。这些案例具有相当的弹性和足够的信息量,只要教师愿意,他们就能够方便地写出自己的练习题。当然,关键是每年我们都会为每个范例分析出版新的习题集。这些新的习题会在一些参数上做重大更改,这样老习题的答案就不再有效了。

另外一个重大的变化是我们沿袭了 *Computer Organization and Design (COD)* 一书第三版的做法,将文字内容进行缩减,只包含任何一个读者都可能会希望看到的内容。对于那些某些读者认为是可有可无或者是参考内容的附录,我们把它放到了随书附带的光盘中。这样做,出于以下一些原因:

1. 学生抱怨这本书越来越重。第一版中主要章节的内容共有 594 页,另外还有 160 页的附录;到第二版,已经变成了 760 页的主要章节加上 223 页的附录;在第三版,这两个数字已经分别变成了 883 页和 209 页,除此之外,还有 245 页的在线附录<sup>①</sup>。按照这样的速度,第四版将会有超过 1500 页的印刷内容和在线内容!
2. 基于同样的原因,教师们担心在这一门课程中会因内容过多而无法全部涵盖。
3. 和 *Computer Organization and Design* 一样,本书将部分内容放到了光盘中。它能使学生无论是否能够访问 Elsevier 的网站,都可以迅速地找到任何相关信息。而且,这一版中的附录即使在以后的版本出版以后,也能够继续为读者所用。
4. 这种灵活性让我们能够将关于流水线、指令系统、存储层次结构这些部分的回顾内容从各个章节里面抽取出来放到附录 A, B, C 中。对于教师和读者而言,这样做的好处在于他们可以快速地浏览这些回顾内容,然后将更多的时间和精力放到第 2 章、第 3 章和第 5 章的深层话题上。另外,我们还可以把一些重要但不是核心的课程内容放到光盘的附录中。这样一来,资料的量没有任何变化,但是印刷的书却精简了。在这一版中,我们一共有六个章节,所有的章节都不超过 80 页。在上一版中,我们一共有八个章节,最长的章节就有 127 页。
5. 这次的书加上光盘与以前的版本相比,制作成本更低。这样出版商也能够更多地降低售价。在目前的价格范围,已经没有必要为欧洲的读者单独制作一个海外学生版了。

还有一个与前一版本不同的重大变化是我们把第三版中的一些嵌入式系统资料移到了它自己的附录 D 中。我们觉得这些嵌入式系统材料并不总是和其余材料的量化评价方法相符合,而且还使得很多原本就已经很长的章节更加冗长。我们相信将所有嵌入式系统资料放到一个单独的附录中应该还会给教学带来一定的便利。

这一版本沿袭了使用真实世界中的实例来说明概念的做法。“综合”部分是全新的内容。事实上,有些内容是在我们这本书送交出版商之后才宣布的。这一版本中的“综合”部分包括了 Intel

<sup>①</sup> 注意,本页的页数是指英文版的页数,不是指中文版的页数。——编者注

Pentium 4 和 AMD Opteron 的流水线组织和存储系统结构, 以及 Sun T1 (Niagara) 8 位处理器、32 位线程微处理器、最新的 NetApp Filer、Internet Archive 集群和 IBM Blue Gene/L 大规模并行处理器系统 (MPP)。

## 主题的选择和组织

和以前一样, 我们对主题的选择采取了保守的做法。毕竟, 在这个领域让人感兴趣的话题多如牛毛, 在这样一本以基本原理为核心的书中, 保守的做法是唯一合理的。我们没有对读者可能遇到的所有系统结构做详尽的调查说明。相反, 我们把重点放在那些任何一台新机器中都可能用到的核心概念上。我们的标准始终如一, 即选择那些已经被验证过且被成功实施过的技术, 这样才能对它们进行量化分析。

我们一直试图把精力放在那些无法从其他资源得到的内容上。因此只要有可能, 我们就不断地加强更高级的内容。事实上, 这本书里对好几个系统的描述都是首次公布 (对于想要了解关于计算机系统结构更为基础的一些概念的读者, 建议阅读 *Computer Organization and Design, The Hardware/Software Interface* 一书的第三版)。

## 内容概述

在这一版中, 第 1 章的内容更为丰富。其中包括了静态电源、动态电源、集成电路成本、可靠性以及可用性的计算公式。和此前的版本相比, 我们更深入地讨论了几何平均值和几何标准偏差在获取平均值变化方面的应用。我们希望这些概念可以在整本书中都得到体现和使用。除了经典的计算机设计和性能测量方面的量化原则之外, 对通用标准部分还进行了更新, 并使用了新的 SPEC2006 套件。

我们认为指令集系统结构 (ISAs) 在今天所扮演的角色已经没有 1990 年时那么重要了, 所以我们将这一部分内容移到了附录 B 中。其中使用的仍然是 MIPS64 系统结构。对于 ISAs 的支持者, 附录 J 涵盖了有关 RISC 系统结构、80x86、DEC VAX 和 IBM 360/370 方面的内容。

第 2 章和第 3 章讲述了指令级并行在高性能处理器中的运用, 其中包括超标量执行、转移<sup>①</sup>预测、推测、动态调度以及相关的编译器技术。前文已经提到, 附录 A 对流水线进行了回顾, 需要时可作为参考。第 3 章对 ILP 的局限性进行了分析。这一版本中新增了对多线程的量化评估。第 3 章还包括了对以下不同处理器的逐项对比: AMD Athlon, Intel Pentium 4, Intel Itanium 2 和 IBM Power 5。这些处理器对 ILP 和 TLP 都有各自不同的应用。在上一版中有大量的篇幅对 Itanium 进行了讨论, 在这一版中, 这部分内容被移到了附录 G。原因是我们认为 Itanium 现在看来并没有当时所预计的那样有潜力。

由于业界现在已经从只对 ILP 进行研究转向了同时还研究线程级并行和数据级并行, 我们把与多处理器系统的相关内容提前到了第 4 章, 重点讲述共享存储式系统结构。这一章就是以这样一个系统结构的性能表现来引入主题的。这一章接着讨论了对称和分布存储层次结构, 研究了组织原则和性能。再下来是关于同步和存储一致性模型指令级并行的内容。其中用 Sun T1 (Niagara), 一个商业产品的初始设计来举例说明。这一设计回归到了单一指令发射、6 段流水微系统结构。它将 8 个这样的设计放到一个芯片上, 每个支持 4 个线程。于是软件会在这个低电源的单芯片上有 32 个线程。

上文已经提到, 附录 C 中对 Cache 原理进行了回顾, 需要时可作为参考。通过这一调整, 第 5 章可以一开始就讨论 11 种对 Cache 进行优化的方法。这一章新加入了有关虚拟机的内容。虚拟机在计算机保护、软件及硬件管理方面有着很多的优势。实例部分采用的是 AMD Opteron, 它最近刚刚扩展到了 64 位地址, 在 Cache 系统结构和虚拟内存模式方面很有代表性。

<sup>①</sup> Branch 一词在全书中翻译为“转移”, 也可译为“分支”, 但译者认为前者更为贴切。——译者注

第6章进一步增加了对可靠性和可用性的讨论,并以 RAID 为例描述了 RAID 的6个模式,以及在真实系统上极少会发现的失效数据。接着介绍了排队论和 I/O 性能基准。与上一版本不同,我们没有一步一步地介绍构建一个假想集群的过程,取而代之的是对真实集群(Internet Archive)中成本、性能以及可靠性的评估。“综合”部分以 NetApp FAS60000 过滤器为例,它使用的是 AMD Opteron 微处理器。

下面是附录 A 到 L。上文已经提过,附录 A 和 C 是关于流水线和缓存概念的基本介绍。对于流水线还比较生疏的读者,应该在阅读第2章和第3章之前先阅读附录 A。对缓存不太了解的读者应该在阅读附录 C 之后再阅读第5章。

附录 B 讲述了 ISA 的基本原理,其中包括 MIPS64。附录 J 描述了 Alpha, MIPS, PowerPC 和 SPARC 的 64 位版本,以及它们各自的多媒体扩展。除此之外,这一部分还包括了一些经典的系统结构(80x86, VAX, IBM 360/370)以及比较主流的嵌入式指令系统(ARM, Thumb, SuperH, MIPS16, Mitsubishi M32R)。附录 G 也和这部分内容相关,主要讲述了 VLIW ISA 的系统结构。

附录 D 是由 Thomas M. Conte 来负责更新的,主要将嵌入的资料全部集中到了一起。

附录 E 是关于网络的。Timothy M. Pinkston 和 José Duato 对其做了大量的修改。附录 F 是由 Krste Asanovic 来修改的,其中包括了对向量处理器的描述。我们认为这两个附录是我们所了解的关于它们各自主题的最好的学习资料之一。

附录 H 描述了并行处理应用和一致性协议在大型共享存储的多处理过程中的应用。附录 I 是由 David Goldberg 完成的,介绍了计算机算术运算。

附录 K 将第三版中各个章节中的“历史回顾与参考”部分集中到了一起。这样做是为了更好地展示每个章节中的各种创新想法,让读者更好地了解这些发明背后的历史。我们希望呈现给读者一个有关计算机设计的演变进程。同时,它还给主修计算机系统结构的学生提供了很好的参考资料。如果有时间的话,我们建议读者可以读一些在这些章节当中提到的这一领域的经典文章。直接从创新者那里了解他们的想法是一件获益匪浅的乐事。“历史回顾”是以前版本中最受欢迎的章节。

附录 L(也可以通过 [textbooks.elsevier.com/0123704901](http://textbooks.elsevier.com/0123704901) 访问)中是有关本书中范例分析习题的答案。

## 关于阅读顺序

阅读这本书没有一个最佳的顺序。但是所有的读者都应该从第1章开始阅读。如果不想通读这本书,则这里建议读者以这样一些顺序来阅读:

- ILP: 附录 A, 第2章和第3章, 附录 F 和 G
- 存储系统结构: 附录 C 和第5章、第6章
- 线程级和数据级并行: 第4章, 附录 H 和 E
- ISA: 附录 B 和 J

附录 D 可以随时阅读,但是在读完 ISA 和 Cache 后再阅读可能效果最好。在任何时候,只要读者对计算机算法感兴趣,都可以去参考附录 I。

## 关于章节结构

在每个章节里面,我们所选择的材料都是按照既定的结构进行组织的。每一章以这一章当中的主要概念介绍开始,接着是“相关问题”部分,它围绕着这一章中的内容是如何与其他章节中的内容相互关联而展开。第三部分称为“综合”,它通过真实的机器来把所有这些概念组织到一起。

第四部分是“谬误和易犯的错误”，其目的是希望学生能从别人的错误中汲取经验。我们给出的都是一些常见的误解和极难避免的系统结构陷阱。“谬误和易犯的错误”部分是这本书中最受欢迎的内容。每一章的内容均以“结论”结束。

## 范例分析及习题

每一章的最后都有范例分析和相关习题。这些范例和习题由业界和学术界的专家编撰而成，目的是对每一章的核心概念进行探讨，并通过越来越具挑战性的习题来考察理解的程度。教师们应该会发现这些范例分析都非常详尽，而且给予了他们编写自己的习题的很大空间。

每个习题中用尖括号括起的部分(<章节>)表示这部分内容对于完成习题非常重要。我们希望通过这种方式能让读者避免在没有阅读相关内容的情况下去试图做习题。而且这样读者也能够方便地找到需要回顾的内容。请注意所有习题的答案都在附录L中。我们根据每个习题可能需要的解答时间给习题划分了不同的等级：

- [10] 不到5分钟可以完成阅读和理解
- [15] 得出完整的答案需要5~15分钟
- [20] 得出完整的答案需要15~20分钟
- [25] 写出完整的答案需要1个小时
- [30] 小型编程项目：需要不到一天时间进行编程
- [40] 大型编程项目：需要两个星期的时间
- [讨论] 需要与其他人就主题进行讨论

那些在 [textbooks.elsevier.com/0123704901](http://textbooks.elsevier.com/0123704901) 注册的教师，可以得到另外一套范例分析习题。这套习题会在每年的夏季做更新，这样到秋季新学期开始时，教师们就可以下载一套全新的配套习题和答案了。

## 补充材料

随书附带的光盘中包含了以下内容：

- 参考附录：一些由主题专家们撰写的猜想，其内容涵盖了许多高级话题
- 历史回顾资料：主要探讨了每一章中核心概念的发展历程
- 正文和光盘内容的搜索引擎

其他的附加内容可以在 [textbook.elsevier.com/012374901](http://textbook.elsevier.com/012374901) 处访问到。而在教师专用站点(需要在 [textbook.elsevier.com](http://textbook.elsevier.com) 注册)，还包括以下内容：

- 另外一套完整的配套习题和答案(每年更新一次)
- 演示文档格式的教学演示稿
- 书中用到的图，JPEG和PPT格式

在本书的配套网站(所有读者均可访问)上包括以下内容：

- 书中范例分析及习题的答案
- 相关资料的链接
- 勘误表

我们会定期对新的资料和网络上相关资源的链接进行更新。



## 给出修改建议

最后，基于与成本-性能类似的考虑，当读者在读这本书时，别忘了还可以赚钱！如果读者看看下面的“致谢”这一部分，就会了解我们在更正错误方面所做出的努力。因为这本书经过了多次出版，我们可以有很多机会来改正其中的错误。如果读者发现了任何遗留的错误，请通过电子邮件和出版商联系（邮件地址为 [ca4bugs@mkp.com](mailto:ca4bugs@mkp.com)）。第一位报告错误且给出更正的读者，如果更正在我们以后的出版当中被采纳，这位读者将获得 1 美元的奖励。请在报告错误之前浏览我们的主页（[textbooks.elsevier.com/0123704901](http://textbooks.elsevier.com/0123704901)）上的勘误表，查看该错误是否已经被报告过。我们会处理这些错误并在大概一年之内寄出奖金的支票，请读者耐心等待。

我们同时也欢迎对于这本书的任何建议和意见，请将你的意见发送到邮箱 [ca4comments@mkp.com](mailto:ca4comments@mkp.com)。

## 结束语

这里我们想再次强调的是这本书完全是合作的结果，我们每个人负责编写每章以及附录的一半内容。我们无法想象如果没有另外的人来负责另一半的工作，这本书将会需要多久才能完成。我们彼此感谢对方在自己感到任务的完成遥遥无期时所给予的激励，当自己遇到晦涩的概念时提供解释这些概念的好方法，在周末花时间帮对方做内容的审核，当在其他工作的繁重压力下自己感到无法再提起笔时对方给予的安慰（随着这本书的一版再版，其他的工作急剧增多。特别是我们其中一个人是斯坦福大学的校长，而另外一位是美国计算机学会的主席）。当然，我们也共同承担读者因书中的内容而产生的任何不满和抱怨。

John L. Hennessy 和 David A. Patterson