

计算机体系结构



[计算机体系结构 下载链接1](#)

著者:John L. Hennessy

出版者:机械工业出版社

出版时间:2012-1

装帧:

isbn:9787111364580

编辑推荐

“本书之所以成为永恒的经典，是因为它的每一次再版都不仅仅是更新补充，而是一次全面的修订，对这个激动人心且快速变化领域给出了最及时的信息和最独到的解读。对于我来说，即使已有二十多年的从业经历，再次阅读本书仍自觉学无止境，感佩于两位卓越大师的渊博学识和深厚功底。”

——Luiz André Barroso，Google公司

内容简介

本书堪称计算机系统结构学科的“圣经”，是计算机设计领域学生和实践者的必读经典。本书系统地介绍了计算机系统的设计基础、存储器层次结构设计、指令级并行及其开发、数据级并行、GPU体系结构、线程级并行和仓库级计算机等。

现今计算机界处于变革之中：移动客户端和云计算正在成为驱动程序设计和硬件创新的主流范型。因此在这个最新版中，作者考虑到这个巨大的变化，重点关注了新的平台（个人移动设备和仓库级计算机）和新的体系结构（多核和GPU），不仅介绍了移动计算和云计算等新内容，还讨论了成本、性能、功耗、可靠性等设计要素。每章都有两个真实例子，一个来源于手机，另一个来源于数据中心，以反映计算机界正在发生的革命性变革。

本书内容丰富，既介绍了当今计算机体系结构的最新研究成果，也引述了许多计算机系统开发方面的实践经验。另外，各章结尾还附有大量的习题和参考文献。本书既可以作为高等院校计算机专业高年级本科生和研究生学习“计算机体系结构”课程的教材或参考书，也可供与计算机相关的专业人士学习参考。

本书特色

- 更新相关内容以覆盖移动计算变革，并强调当今体系结构中最重要两个主题：存储器层次结构和各种并行技术。
- 每章中的“Putting It All Together”小节关注了业界的各种最新技术，包括ARM Cortex-A8、Intel Core i7、NVIDIA GTX-280和GTX-480 GPU，以及一种Google仓库级计算机。
- 每章都设计了常规主题：能力、性能、成本、可依赖性、保护、编程模型和新趋势。
- 书中包括3个附录，另外8个附录可以在原出版社网站在线得到。
- 每章最后都设置了由工业界和学术界专家提供的经过更新的案例研究，以及与之配套的全新练习题。

作者介绍:

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，并且和本书另外一位作者David A. Patterson分享了2000年John von Neumann奖。

David A. Patterson

加州大学伯克利分校计算机科学系主任、教授，美国国家工程院院士，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。

目录: Foreword

Preface

Acknowledgments

Chapter 1 Fundamentals of Quantitative Design and Analysis

1.1 Introduction

1.2 Classes of Computers

1.3 Defining Computer Architecture

1.4 Trends in Technology

1.5 Trends in Power and Energy in Integrated Circuits

1.6 Trends in Cost

1.7 Dependability

1.8 Measuring, Reporting, and Summarizing Performance

1.9 Quantitative Principles of Computer Design

1.10 Putting It All Together: Performance, Price, and Power

1.11 Fallacies and Pitfalls

1.12 Concluding Remarks

1.13 Historical Perspectives and References Case Studies and Exercises by Diana Franklin

Chapter 2 Memory Hierarchy Design

2.1 Introduction

2.2 Ten Advanced Optimizations of Cache Performance

2.3 Memory Technology and Optimizations

2.4 Protection: Virtual Memory and Virtual Machines

2.5 Crosscutting Issues: The Design of Memory Hierarchies

2.6 Putting It All Together: Memory Hierachies in the ARM Cortex-AS and Intel Core i7

2.7 Fallacies and Pitfalls

2.8 Concluding Remarks: Looking Ahead

2.9 Historical Perspective and References Case Studies and Exercises by Norman P. Jouppi, Naveen Muralimanohar, and Sheng Li

Chapter 3 nstruction-Level Parallelism and Its Exploitation

3.1 Instruction-Level Parallelism: Concepts and Challenges

3.2 Basic Compiler Techniques for Exposing ILP

3.3 Reducing Branch Costs with Advanced Branch Prediction

3.4 Overcoming Data Hazards with Dynamic Scheduling

3.5 Dynamic Scheduling: Examples and the Algorithm

3.6 Hardware-Based Speculation

3.7 Exploiting ILP Using Multiple Issue and Static Scheduling

3.8 Exploiting ILP Using Dynamic Scheduling, Multiple Issue, and Speculation

3.9 Advanced Techniques for Instruction Delivery and Speculation

3.10 Studies of the Limitations of ILP

3.11 Cross-Cutting Issues: ILP Approaches and the Memory System

3.12 Multithreading: Exploiting Thread-Level Parallelism to Improve Uniprocessor

Throughput

3.13 Putting It All Together: The Intel Core i7 and ARM Cortex-AS

3.14 Fallacies and Pitfalls

3.15 Concluding Remarks: What's Ahead?

3.16 Historical Perspective and References Case Studies and Exercises by Jason D. Bakos and Robert R Colwell

Chapter 4 Data-Level Parallelism in Vector, SIMD, and GPU Architectures

4.1 Introduction

4.2 Vector Architecture

4.3 SIMD Instruction Set Extensions for Multimedia

4.4 Graphics Processing Units

4.5 Detecting and Enhancing Loop-Level Parallelism

4.6 Crosscutting Issues

4.7 Putting It All Together: Mobile versus Server GPUS and Tesla versus Core i7

4.8 Fallacies and Pitfalls

4.9 Concluding Remarks

4.10 Historical Perspective and References Case Study and Exercises by Jason D. Bakos

Chapter 5 Thread-Level Parallelism

5.1 Introduction

5.2 Centralized Shared-Memory Architectures

5.3 Performance of Symmetric Shared-Memory Multiprocessors

5.4 Distributed Shared-Memory and Directory-Based Coherence

5.5 Synchronization: The Basics

5.6 Models of Memory Consistency: An Introduction

5.7 Crosscutting Issues

5.8 Putting It All Together: Multicore Processors and Their Performance

5.9 Fallacies and Pitfalls

5.10 Concluding Remarks

5.11 Historical Perspectives and References Case Studies and Exercises by Amr Zaky and David A. Wood

Chapter 6 Warehouse-Scale Computers to Exploit Request-Level and Data-Level Parallelism

6.1 Introduction

6.2 Programming Models and Workloads for Warehouse-Scale Computers

6.3 Computer Architecture of Warehouse-Scale Computers

6.4 Physical Infrastructure and Costs of Warehouse-Scale Computers

6.5 Cloud Computing: The Return of Utility Computing

6.6 Crosscutting Issues

6.7 Putting It All Together: A Google Warehouse-Scale Computer

6.8 Fallacies and Pitfalls

6.9 Concluding Remarks

6.10 Historical Perspectives and References Case Studies and Exercises by Parthasarathy Ranganathan

Appendix A Instruction Set Principles

A.1 Introduction

A.2 Classifying Instruction Set Architectures

A.3 Memory Addressing

A.4 Type and Size of Operands

A.5 Operations in the Instruction Set

A.6 Instructions for Control Flow

A.7 Encoding an Instruction Set

A.8 Crosscutting Issues: The Role of Compilers

A.9 Putting It All Together: The MIPS Architecture

A.10 Fallacies and Pitfalls

A.11 Concluding Remarks

A.12 Historical Perspective and References Exercises by Gregory D. Peterson

Appendix B Review of Memory Hierarchy

B.1 Introduction
B.2 Cache Performance
B.3 Six Basic Cache Optimizations
B.4 Virtual Memory
B.5 Protection and Examples of Virtual Memory
B.6 Fallacies and Pitfalls
B.7 Concluding Remarks
B.8 Historical Perspective and References
Exercises by Amr Zaky
Appendix C Pipelining: Basic and Intermediate Concepts
C.1 Introduction
C.2 The Major Hurdle of Pipelining--Pipeline Hazards
C.3 How Is Pipelining Implemented?
C.4 What Makes Pipelining Hard to Implement?
C.5 Extending the MIPS Pipeline to Handle Multicycle Operations
C.6 Putting It All Together: The MIPS R4000 Pipeline
C.7 Crosscutting Issues
C.8 Fallacies and Pitfalls
C.9 Concluding Remarks
C.10 Historical Perspective and References
Updated Exercises by Diana Franklin
Online Appendices
Appendix D Storage Systems
Appendix E Embedded Systems
By ThomasM Conte
Appendix F Interconnection Networks
Revised by Timothy M. Pinkston and Jose Duoto
Appendix G Vector Processors in More Depth
Revised by Krste Asanovic
Appendix H Hardware and Software for VLIW and EPIC
Appendix I Large-Scale Multiprocessors and Scientific Applications
Appendix J Computer Arithmetic
by David Goldberg
Appendix K Survey of Instruction Set Architectures
Appendix L Historical Perspectives and References
References
Index
1.1 Introduction
1.2 Classes of Computers
1.3 Defining Computer Architecture
1.4 Trends in Technology
1.5 Trends in Power and Energy in Integrated Circuits
1.6 Trends in Cost
1.7 Dependability
1.8 Measuring Reporting, and Summarizing Performance
1.9 Quantitative Principles of Computer Design
1.10 Putting It All Together: Performance, Price, and Power
1.11 Fallacies and Pitfalls
1.12 Concluding Remarks
1.13 Historical Perspectives and References
1.14 Case Studies and Exercises by Diana Franklin
• • • • • ([收起](#))

标签

计算机体系结构

计算机

计算机科学

体系结构

经典

经典著作

CS

计算机系统

评论

建议读本书前先看一下《计算机组成与设计:软硬件接口》，本书阅读会流畅许多

原来hennessy写的是这本...书没看懂但今年也算是现场听过他扯淡了

哎，感觉不适合做教材

每个CS人都应该读的书籍。

重新开读。

相对第4版：更新了每章后面的实例分析部分，增加了GPU相关章节，总的来说改动不大

当初学的死去活来。

无需多言的经典。相比之前的第四版有了大幅度更新，准备今年好好再读一遍。

13/11/22 因为蛋疼的243开始读的. 发现写的确实很赞~

中英文的间杂着看，中文的太多瑕疵了，勘误了一堆。看了一部分只有，应试。

北大CS研究生教材

一刷

教材。。。。

体系结构研究入门书 我的英语书籍阅读的第一个啃的书籍

Nice book. Clear statement, variety of examples.

高体是门神课，听完课程介绍的我瑟瑟发抖。paper多，textbook厚，presentation多，编程更多！

《软硬件接口》进阶版，紧贴时代潮流。

第5版！

这一版的笔误明显比上一版多，让人有些失望。新增加的WSC一章，对开拓学生的视野很有好处，不过把集群看做一台计算机，个人观点，略显牵强了。

这本书是普林斯顿大学EECS Computer Architecture这门课的标准教材，我当年去听过这门课，非常棒，学透的话对于CS科研和从业非常有帮助。

[计算机体系结构_下载链接1](#)

书评

Computer Architecture的另一个名字是Trade-off。看着整个体系结构的发展历程中种种新技术的使用，无不体现了Trade-off的设计思想。硬件一直都是并行工作的，不管是pipeline, out-of-order, superscale, VLIW, Instruction Level Parallelism抑或是现在的Multicore，硬件的发展...

我个人认为任何一个学计算机的，你可以不把自己的研究方向设在架构，编译器等等，

但你必须要懂架构，编译器，操作系统。我个人认为这些东西对于一个CS的人来说不亚于结构，算法，以及标准库的重要。
我一直觉得英文书比中文书好懂，因为中文书喜欢咬文嚼字装专业，也或者是译...

书籍说明 计算机体系结构领域最经典著作 两位作者都是该领域的大牛
特别是附录部分，对计算机硬件的基础知识有一个全面介绍
适合基础不好的同学认真阅读(比如我。。。) 阅读建议
计算机体系结构最经典书籍，适合和第四章结合起来阅读 另外：
还是英文原版比较靠谱。。。

看第三版是因为家里已经有了。因为实在太厚（比乔布斯传还厚），拿起来太不方便，于是在网上寻找电子版。结果找到的大都是第4版的了，也挺好的。
第三版是2002年的，第四版这是2006年出版的。从封面的照片上就可以看出不同了：从一根柱子，变成了一堆柱子。这真是一本与时...

这本书能够很全面的介绍计算机体系结构方面的知识，对于有兴趣在计算机体系结构，编译原理等方面进行发展的学生来说，个人觉得是必看的书。即使你的兴趣不在体系结构这个方面，读这本书对你也会有很大的帮助。
不过现在已经出到第四版了，建议大家还是参考最新的文献。

我只是刚刚开始读，
但是已经明白为什么L1Cache有ICache和DCache之分,为什么有N-way Cache,
为什么Pipeline让CPU更快, CPU的Pipeline的风险以及解决之道...
总之这本书让我大开眼界!

本书总的来说还是比较高级的内容，对内存模型，缓存结构方式的概念原理等都略过不讲。专心讲 quantitative 的部分。如果对体系结构不熟悉，应该先读 或者。
本书最大的特点就是...

为了图速度，就买了中文，太坑了，翻译完全没逻辑，请翻译的老师不要找你的学生翻译这种经典书籍了。还得网上下载原版英文看，更浪费时间，建议直接买英文版，如果英语不太好，那也别买中文版，汉字你是认识，但放在一起，那句子。。。fk
把我整的很凌乱。计算机类的经典书籍建...

北大东门附近交易吧。第五版，英文版，机械工业出版的，9成新，原价138，80出。
电话：133411267三七 啦啦啊啦啦啦啊啦啦啦啊啦啦啦啊啦啦
啦啦啊啦啦啦啊啦啦啦啊啦啦啦啊啦啦啦啊啦啦啦啊啦啦啊啦啦啊啦啦

计算机体系结构_下载链接1_