


Contents

Preface xv

CHAPTERS

1



Computer Abstractions and Technology 2

- 1.1 Introduction 3
- 1.2 Eight Great Ideas in Computer Architecture 11
- 1.3 Below Your Program 13
- 1.4 Under the Covers 16
- 1.5 Technologies for Building Processors and Memory 24
- 1.6 Performance 28
- 1.7 The Power Wall 40
- 1.8 The Sea Change: The Switch from Uniprocessors to Multiprocessors 43
- 1.9 Real Stuff: Benchmarking the Intel Core i7 46
- 1.10 Fallacies and Pitfalls 49
- 1.11 Concluding Remarks 52
-  1.12 Historical Perspective and Further Reading 54
- 1.13 Exercises 54


2

Instructions: Language of the Computer 60


- 2.1 Introduction 62
- 2.2 Operations of the Computer Hardware 63
- 2.3 Operands of the Computer Hardware 66
- 2.4 Signed and Unsigned Numbers 73
- 2.5 Representing Instructions in the Computer 80
- 2.6 Logical Operations 87
- 2.7 Instructions for Making Decisions 90
- 2.8 Supporting Procedures in Computer Hardware 96
- 2.9 Communicating with People 106
- 2.10 MIPS Addressing for 32-Bit Immediates and Addresses 111
- 2.11 Parallelism and Instructions: Synchronization 121
- 2.12 Translating and Starting a Program 123
- 2.13 A C Sort Example to Put It All Together 132
- 2.14 Arrays versus Pointers 141


-  2.15 Advanced Material: Compiling C and Interpreting Java 145
- 2.16 Real Stuff: ARMv7 (32-bit) Instructions 145
- 2.17 Real Stuff: x86 Instructions 149
- 2.18 Real Stuff: ARMv8 (64-bit) Instructions 158
- 2.19 Fallacies and Pitfalls 159
- 2.20 Concluding Remarks 161
-  2.21 Historical Perspective and Further Reading 163
- 2.22 Exercises 164

3**Arithmetic for Computers 176**




- 3.1 Introduction 178
- 3.2 Addition and Subtraction 178
- 3.3 Multiplication 183
- 3.4 Division 189
- 3.5 Floating Point 196
- 3.6 Parallelism and Computer Arithmetic: Subword Parallelism 222
- 3.7 Real Stuff: x86 Streaming SIMD Extensions and Advanced Vector Extensions 224
- 3.8 Going Faster: Subword Parallelism and Matrix Multiply 225
- 3.9 Fallacies and Pitfalls 229
- 3.10 Concluding Remarks 232
-  3.11 Historical Perspective and Further Reading 236
- 3.12 Exercises 237

4**The Processor 242**


- 4.1 Introduction 244
- 4.2 Logic Design Conventions 248
- 4.3 Building a Datapath 251
- 4.4 A Simple Implementation Scheme 259
- 4.5 An Overview of Pipelining 272
- 4.6 Pipelined Datapath and Control 286
- 4.7 Data Hazards: Forwarding versus Stalling 303
- 4.8 Control Hazards 316
- 4.9 Exceptions 325
- 4.10 Parallelism via Instructions 332
- 4.11 Real Stuff: The ARM Cortex-A8 and Intel Core i7 Pipelines 344
- 4.12 Going Faster: Instruction-Level Parallelism and Matrix Multiply 351
-  4.13 Advanced Topic: An Introduction to Digital Design Using a Hardware Design Language to Describe and Model a Pipeline and More Pipelining Illustrations 354


- 4.14 Fallacies and Pitfalls 355
- 4.15 Concluding Remarks 356
-  4.16 Historical Perspective and Further Reading 357
- 4.17 Exercises 357

5**Large and Fast: Exploiting Memory Hierarchy 372**

- 5.1 Introduction 374
- 5.2 Memory Technologies 378
- 5.3 The Basics of Caches 383
- 5.4 Measuring and Improving Cache Performance 398
- 5.5 Dependable Memory 418
- 5.6 Virtual Machines 424
- 5.7 Virtual Memory 427
- 5.8 A Common Framework for Memory Hierarchy 454
- 5.9 Using a Finite-State Machine to Control a Simple Cache 461
- 5.10 Parallelism and Memory Hierarchies: Cache Coherence 466
-  5.11 Parallelism and Memory Hierarchy: Redundant Arrays of Inexpensive Disks 470
-  5.12 Advanced Material: Implementing Cache Controllers 470
- 5.13 Real Stuff: The ARM Cortex-A8 and Intel Core i7 Memory Hierarchies 471
- 5.14 Going Faster: Cache Blocking and Matrix Multiply 475
- 5.15 Fallacies and Pitfalls 478
- 5.16 Concluding Remarks 478
-  5.17 Historical Perspective and Further Reading 483
- 5.18 Exercises 483

6**Parallel Processors from Client to Cloud 500**

- 6.1 Introduction 502
- 6.2 The Difficulty of Creating Parallel Processing Programs 504
- 6.3 SISD, MIMD, SIMD, SPMD, and Vector 509
- 6.4 Hardware Multithreading 516
- 6.5 Multicore and Other Shared Memory Multiprocessors 519
- 6.6 Introduction to Graphics Processing Units 524
- 6.7 Clusters, Warehouse Scale Computers, and Other Message-Passing Multiprocessors 531
- 6.8 Introduction to Multiprocessor Network Topologies 536
-  6.9 Communicating to the Outside World: Cluster Networking 539
- 6.10 Multiprocessor Benchmarks and Performance Models 540
- 6.11 Real Stuff: Benchmarking Intel Core i7 versus NVIDIA Tesla GPU 550

- 6.12 Going Faster: Multiple Processors and Matrix Multiply 555
- 6.13 Fallacies and Pitfalls 558
- 6.14 Concluding Remarks 560
-  6.15 Historical Perspective and Further Reading 563
- 6.16 Exercises 563

A P P E N D I C E S



Assemblers, Linkers, and the SPIM Simulator A-2

- A.1 Introduction A-3
- A.2 Assemblers A-10
- A.3 Linkers A-18
- A.4 Loading A-19
- A.5 Memory Usage A-20
- A.6 Procedure Call Convention A-22
- A.7 Exceptions and Interrupts A-33
- A.8 Input and Output A-38
- A.9 SPIM A-40
- A.10 MIPS R2000 Assembly Language A-45
- A.11 Concluding Remarks A-81
- A.12 Exercises A-82



The Basics of Logic Design B-2

- B.1 Introduction B-3
- B.2 Gates, Truth Tables, and Logic Equations B-4
- B.3 Combinational Logic B-9
- B.4 Using a Hardware Description Language B-20
- B.5 Constructing a Basic Arithmetic Logic Unit B-26
- B.6 Faster Addition: Carry Lookahead B-38
- B.7 Clocks B-48
- B.8 Memory Elements: Flip-Flops, Latches, and Registers B-50
- B.9 Memory Elements: SRAMs and DRAMs B-58
- B.10 Finite-State Machines B-67
- B.11 Timing Methodologies B-72
- B.12 Field Programmable Devices B-78
- B.13 Concluding Remarks B-79
- B.14 Exercises B-80

Index I-1

ONLINE CONTENT**Graphics and Computing GPUs C-2**

- C.1 Introduction C-3
- C.2 GPU System Architectures C-7
- C.3 Programming GPUs C-12
- C.4 Multithreaded Multiprocessor Architecture C-25
- C.5 Parallel Memory System C-36
- C.6 Floating Point Arithmetic C-41
- C.7 Real Stuff: The NVIDIA GeForce 8800 C-46
- C.8 Real Stuff: Mapping Applications to GPUs C-55
- C.9 Fallacies and Pitfalls C-72
- C.10 Concluding Remarks C-76
- C.11 Historical Perspective and Further Reading C-77

**Mapping Control to Hardware D-2**

- D.1 Introduction D-3
- D.2 Implementing Combinational Control Units D-4
- D.3 Implementing Finite-State Machine Control D-8
- D.4 Implementing the Next-State Function with a Sequencer D-22
- D.5 Translating a Microprogram to Hardware D-28
- D.6 Concluding Remarks D-32
- D.7 Exercises D-33


**A Survey of RISC Architectures for Desktop, Server, and Embedded Computers E-2**


- E.1 Introduction E-3
- E.2 Addressing Modes and Instruction Formats E-5
- E.3 Instructions: The MIPS Core Subset E-9
- E.4 Instructions: Multimedia Extensions of the Desktop/Server RISCs E-16
- E.5 Instructions: Digital Signal-Processing Extensions of the Embedded RISCs E-19
- E.6 Instructions: Common Extensions to MIPS Core E-20
- E.7 Instructions Unique to MIPS-64 E-25
- E.8 Instructions Unique to Alpha E-27
- E.9 Instructions Unique to SPARC v.9 E-29
- E.10 Instructions Unique to PowerPC E-32
- E.11 Instructions Unique to PA-RISC 2.0 E-34
- E.12 Instructions Unique to ARM E-36
- E.13 Instructions Unique to Thumb E-38
- E.14 Instructions Unique to SuperH E-39

E.15 Instructions Unique to M32R E-40

E.16 Instructions Unique to MIPS-16 E-40

E.17 Concluding Remarks E-43

 Glossary G-1

 Further Reading FR-1