

# CONTENTS

## PREFACE

xviii

## 1 INTRODUCTION

1

- 1.1 STRUCTURED COMPUTER ORGANIZATION 2
  - 1.1.1 Languages, Levels, and Virtual Machines 2
  - 1.1.2 Contemporary Multilevel Machines 5
  - 1.1.3 Evolution of Multilevel Machines 8
- 1.2 MILESTONES IN COMPUTER ARCHITECTURE 13
  - 1.2.1 The Zeroth Generation—Mechanical Computers (1642–1945) 14
  - 1.2.2 The First Generation—Vacuum Tubes (1945–1955) 16
  - 1.2.3 The Second Generation—Transistors (1955–1965) 19
  - 1.2.4 The Third Generation—Integrated Circuits (1965–1980) 22
  - 1.2.5 The Fourth Generation—Very Large Scale Integration (1980–?) 23
  - 1.2.6 The Fifth Generation—Invisible Computers 26
- 1.3 THE COMPUTER ZOO 27
  - 1.3.1 Technological and Economic Forces 27
  - 1.3.2 The Computer Spectrum 29
  - 1.3.3 Disposable Computers 29
  - 1.3.4 Microcontrollers 31
  - 1.3.5 Game Computers 33
  - 1.3.6 Personal Computers 34

- 1.3.7 Servers 34
- 1.3.8 Collections of Workstations 34
- 1.3.9 Mainframes 36
  
- 1.4 EXAMPLE COMPUTER FAMILIES 37
  - 1.4.1 Introduction to the Pentium 4 37
  - 1.4.2 Introduction to the UltraSPARC III 42
  - 1.4.3 Introduction to the 8051 44
  
- 1.5 METRIC UNITS 46
  
- 1.6 OUTLINE OF THIS BOOK 47

## **2 COMPUTER SYSTEMS ORGANIZATION 51**

- 2.1 PROCESSORS 51
  - 2.1.1 CPU Organization 52
  - 2.1.2 Instruction Execution 54
  - 2.1.3 RISC versus CISC 58
  - 2.1.4 Design Principles for Modern Computers 59
  - 2.1.5 Instruction-Level Parallelism 61
  - 2.1.6 Processor-Level Parallelism 65
  
- 2.2 PRIMARY MEMORY 69
  - 2.2.1 Bits 69
  - 2.2.2 Memory Addresses 70
  - 2.2.3 Byte Ordering 71
  - 2.2.4 Error-Correcting Codes 73
  - 2.2.5 Cache Memory 77
  - 2.2.6 Memory Packaging and Types 80
  
- 2.3 SECONDARY MEMORY 81
  - 2.3.1 Memory Hierarchies 81
  - 2.3.2 Magnetic Disks 82
  - 2.3.3 Floppy Disks 86
  - 2.3.4 IDE Disks 86
  - 2.3.5 SCSI Disks 88
  - 2.3.6 RAID 89
  - 2.3.7 CD-ROMs 93

2.3.8	CD-Recordables	97
2.3.9	CD-Rewritables	99
2.3.10	DVD	99
2.3.11	Blu-Ray	102
2.4	INPUT/OUTPUT	102
2.4.1	Buses	102
2.4.2	Terminals	105
2.4.3	Mice	110
2.4.4	Printers	112
2.4.5	Telecommunications Equipment	117
2.4.6	Digital Cameras	125
2.4.7	Character Codes	127
2.5	SUMMARY	131

### **3 THE DIGITAL LOGIC LEVEL**

**135**

3.1	GATES AND BOOLEAN ALGEBRA	135
3.1.1	Gates	136
3.1.2	Boolean Algebra	138
3.1.3	Implementation of Boolean Functions	140
3.1.4	Circuit Equivalence	141
3.2	BASIC DIGITAL LOGIC CIRCUITS	146
3.2.1	Integrated Circuits	146
3.2.2	Combinational Circuits	147
3.2.3	Arithmetic Circuits	152
3.2.4	Clocks	157
3.3	MEMORY	159
3.3.1	Latches	159
3.3.2	Flip-Flops	161
3.3.3	Registers	163
3.3.4	Memory Organization	164
3.3.5	Memory Chips	168
3.3.6	RAMs and ROMs	171

- 3.4 CPU CHIPS AND BUSES 173
  - 3.4.1 CPU Chips 174
  - 3.4.2 Computer Buses 176
  - 3.4.3 Bus Width 178
  - 3.4.4 Bus Clocking 180
  - 3.4.5 Bus Arbitration 184
  - 3.4.6 Bus Operations 187
  
- 3.5 EXAMPLE CPU CHIPS 189
  - 3.5.1 The Pentium 4 189
  - 3.5.2 The UltraSPARC III 196
  - 3.5.3 The 8051 200
  
- 3.6 EXAMPLE BUSES 202
  - 3.6.1 The ISA Bus 203
  - 3.6.2 The PCI Bus 204
  - 3.6.3 PCI Express 212
  - 3.6.4 The Universal Serial Bus 217
  
- 3.7 INTERFACING 221
  - 3.7.1 I/O Chips 221
  - 3.7.2 Address Decoding 222
  
- 3.8 SUMMARY 225

## **4 THE MICROARCHITECTURE LEVEL 231**

- 4.1 AN EXAMPLE MICROARCHITECTURE 231
  - 4.1.1 The Data Path 232
  - 4.1.2 Microinstructions 239
  - 4.1.3 Microinstruction Control: The Mic-1 241
  
- 4.2 AN EXAMPLE ISA: IJVM 246
  - 4.2.1 Stacks 246
  - 4.2.2 The IJVM Memory Model 248
  - 4.2.3 The IJVM Instruction Set 250
  - 4.2.4 Compiling Java to IJVM 254

4.3	AN EXAMPLE IMPLEMENTATION	255
4.3.1	Microinstructions and Notation	255
4.3.2	Implementation of IJVM Using the Mic-1	260
4.4	DESIGN OF THE MICROARCHITECTURE LEVEL	271
4.4.1	Speed versus Cost	271
4.4.2	Reducing the Execution Path Length	273
4.4.3	A Design with Prefetching: The Mic-2	281
4.4.4	A Pipelined Design: The Mic-3	281
4.4.5	A Seven-Stage Pipeline: The Mic-4	288
4.5	IMPROVING PERFORMANCE	292
4.5.1	Cache Memory	293
4.5.2	Branch Prediction	299
4.5.3	Out-of-Order Execution and Register Renaming	304
4.5.4	Speculative Execution	309
4.6	EXAMPLES OF THE MICROARCHITECTURE LEVEL	311
4.6.1	The Microarchitecture of the Pentium 4 CPU	312
4.6.2	The Microarchitecture of the UltraSPARC-III Cu CPU	317
4.6.3	The Microarchitecture of the 8051 CPU	323
4.7	COMPARISON OF THE PENTIUM, ULTRASPARC, AND 8051	325
4.8	SUMMARY	326

## **5 THE INSTRUCTION SET ARCHITECTURE LEVEL 331**

5.1	OVERVIEW OF THE ISA LEVEL	333
5.1.1	Properties of the ISA Level	333
5.1.2	Memory Models	335
5.1.3	Registers	337
5.1.4	Instructions	339
5.1.5	Overview of the Pentium 4 ISA Level	339
5.1.6	Overview of the UltraSPARC III ISA Level	341
5.1.7	Overview of the 8051 ISA Level	345

- 5.2 DATA TYPES 348
  - 5.2.1 Numeric Data Types 348
  - 5.2.2 Nonnumeric Data Types 349
  - 5.2.3 Data Types on the Pentium 4 350
  - 5.2.4 Data Types on the UltraSPARC III 350
  - 5.2.5 Data Types on the 8051 351
  
- 5.3 INSTRUCTION FORMATS 351
  - 5.3.1 Design Criteria for Instruction Formats 352
  - 5.3.2 Expanding Opcodes 354
  - 5.3.3 The Pentium 4 Instruction Formats 357
  - 5.3.4 The UltraSPARC III Instruction Formats 358
  - 5.3.5 The 8051 Instruction Formats 359
  
- 5.4 ADDRESSING 360
  - 5.4.1 Addressing Modes 360
  - 5.4.2 Immediate Addressing 361
  - 5.4.3 Direct Addressing 361
  - 5.4.4 Register Addressing 361
  - 5.4.5 Register Indirect Addressing 362
  - 5.4.6 Indexed Addressing 363
  - 5.4.7 Based-Indexed Addressing 365
  - 5.4.8 Stack Addressing 365
  - 5.4.9 Addressing Modes for Branch Instructions 369
  - 5.4.10 Orthogonality of Opcodes and Addressing Modes 369
  - 5.4.11 The Pentium 4 Addressing Modes 371
  - 5.4.12 The UltraSPARC III Addressing Modes 373
  - 5.4.13 The 8051 Addressing Modes 373
  - 5.4.14 Discussion of Addressing Modes 374
  
- 5.5 INSTRUCTION TYPES 375
  - 5.5.1 Data Movement Instructions 375
  - 5.5.2 Dyadic Operations 376
  - 5.5.3 Monadic Operations 377
  - 5.5.4 Comparisons and Conditional Branches 379
  - 5.5.5 Procedure Call Instructions 381
  - 5.5.6 Loop Control 382
  - 5.5.7 Input/Output 383
  - 5.5.8 The Pentium 4 Instructions 386
  - 5.5.9 The UltraSPARC III Instructions 389
  - 5.5.10 The 8051 Instructions 392
  - 5.5.11 Comparison of Instruction Sets 392

- 5.6 FLOW OF CONTROL 395
  - 5.6.1 Sequential Flow of Control and Branches 395
  - 5.6.2 Procedures 396
  - 5.6.3 Coroutines 401
  - 5.6.5 Traps 404
  - 5.6.5 Interrupts 404
  
- 5.7 A DETAILED EXAMPLE: THE TOWERS OF HANOI 408
  - 5.7.1 The Towers of Hanoi in Pentium 4 Assembly Language 409
  - 5.7.2 The Towers of Hanoi in UltraSPARC III Assembly Language 409
  
- 5.8 THE IA-64 ARCHITECTURE AND THE ITANIUM 2 411
  - 5.8.1 The Problem with the Pentium 4 413
  - 5.8.2 The IA-64 Model: Explicitly Parallel Instruction Computing 414
  - 5.8.3 Reducing Memory References 415
  - 5.8.4 Instruction Scheduling 416
  - 5.8.5 Reducing Conditional Branches: Predication 418
  - 5.8.6 Speculative Loads 420
  
- 5.9 SUMMARY 421

## **6 THE OPERATING SYSTEM MACHINE LEVEL 427**

- 6.1 VIRTUAL MEMORY 428
  - 6.1.1 Paging 429
  - 6.1.2 Implementation of Paging 431
  - 6.1.3 Demand Paging and the Working Set Model 433
  - 6.1.4 Page Replacement Policy 436
  - 6.1.5 Page Size and Fragmentation 438
  - 6.1.6 Segmentation 439
  - 6.1.7 Implementation of Segmentation 442
  - 6.1.8 Virtual Memory on the Pentium 4 445
  - 6.1.9 Virtual Memory on the UltraSPARC III 450
  - 6.1.10 Virtual Memory and Caching 452
  
- 6.2 VIRTUAL I/O INSTRUCTIONS 453
  - 6.2.1 Files 454
  - 6.2.2 Implementation of Virtual I/O Instructions 455
  - 6.2.3 Directory Management Instructions 459

6.3	VIRTUAL INSTRUCTIONS FOR PARALLEL PROCESSING	460
6.3.1	Process Creation	461
6.3.2	Race Conditions	462
6.3.3	Process Synchronization Using Semaphores	466
6.4	EXAMPLE OPERATING SYSTEMS	470
6.4.1	Introduction	470
6.4.2	Examples of Virtual Memory	479
6.4.3	Examples of Virtual I/O	482
6.4.4	Examples of Process Management	493
6.5	SUMMARY	500

## **7 THE ASSEMBLY LANGUAGE LEVEL 507**

7.1	INTRODUCTION TO ASSEMBLY LANGUAGE	508
7.1.1	What Is an Assembly Language?	508
7.1.2	Why Use Assembly Language?	509
7.1.3	Format of an Assembly Language Statement	512
7.1.4	Pseudoinstructions	515
7.2	MACROS	517
7.2.1	Macro Definition, Call, and Expansion	518
7.2.2	Macros with Parameters	520
7.2.3	Advanced Features	521
7.2.4	Implementation of a Macro Facility in an Assembler	521
7.3	THE ASSEMBLY PROCESS	522
7.3.1	Two-Pass Assemblers	522
7.3.2	Pass One	523
7.3.3	Pass Two	527
7.3.4	The Symbol Table	529
7.4	LINKING AND LOADING	530
7.4.1	Tasks Performed by the Linker	532
7.4.2	Structure of an Object Module	535
7.4.3	Binding Time and Dynamic Relocation	536
7.4.4	Dynamic Linking	539
7.5	SUMMARY	543

- 8.1 ON-CHIP PARALELLISM 548
  - 8.1.1 Instruction-Level Parallelism 549
  - 8.1.2 On-Chip Multithreading 556
  - 8.1.3 Single-Chip Multiprocessors 562
  
- 8.2 COPROCESSORS 567
  - 8.2.1 Network Processors 568
  - 8.2.2 Media Processors 576
  - 8.2.3 Cryptoprocessors 581
  
- 8.3 SHARED-MEMORY MULTIPROCESSORS 582
  - 8.3.1 Multiprocessors vs. Multicomputers 582
  - 8.3.2 Memory Semantics 590
  - 8.3.3 UMA Symmetric Multiprocessor Architectures 594
  - 8.3.4 NUMA Multiprocessors 602
  - 8.3.5 COMA Multiprocessors 611
  
- 8.4 MESSAGE-PASSING MULTICOMPUTERS 612
  - 8.4.1 Interconnection Networks 614
  - 8.4.2 MPPs—Massively Parallel Processors 617
  - 8.4.3 Cluster Computing 627
  - 8.4.4 Communication Software for Multicomputers 632
  - 8.4.5 Scheduling 635
  - 8.4.6 Application-Level Shared Memory 636
  - 8.4.7 Performance 643
  
- 8.5 GRID COMPUTING 649
  
- 8.6 SUMMARY 651

**9      READING LIST AND BIBLIOGRAPHY      655**

- 9.1 SUGGESTIONS FOR FURTHER READING 655
  - 9.1.1 Introduction and General Works 655
  - 9.1.2 Computer Systems Organization 657
  - 9.1.3 The Digital Logic Level 658

- 9.1.4 The Microarchitecture Level 659
  - 9.1.5 The Instruction Set Architecture Level 659
  - 9.1.6 The Operating System Machine Level 660
  - 9.1.7 The Assembly Language Level 661
  - 9.1.8 Parallel Computer Architectures 661
  - 9.1.9 Binary and Floating-Point Numbers 663
  - 9.1.10 Assembly Language Programming 664
- 9.2 ALPHABETICAL BIBLIOGRAPHY 664

## **A BINARY NUMBERS 679**

- A.1 FINITE-PRECISION NUMBERS 679
- A.2 RADIX NUMBER SYSTEMS 681
- A.3 CONVERSION FROM ONE RADIX TO ANOTHER 683
- A.4 NEGATIVE BINARY NUMBERS 685
- A.5 BINARY ARITHMETIC 688

## **B FLOATING-POINT NUMBERS 691**

- B.1 PRINCIPLES OF FLOATING POINT 692
- B.2 IEEE FLOATING-POINT STANDARD 754 694

## **C ASSEMBLY LANGUAGE PROGRAMMING 701**

- C.1 OVERVIEW 702
  - C.1.1 Assembly Language 702
  - C.1.2 A Small Assembly Language Program 703

- C.2 THE 8088 PROCESSOR 704
  - C.2.1 The Processor Cycle 705
  - C.2.2 The General Registers 705
  - C.2.3 Pointer Registers 708
  
- C.3 MEMORY AND ADDRESSING 709
  - C.3.1 Memory Organization and Segments 709
  - C.3.2 Addressing 711
  
- C.4 THE 8088 INSTRUCTION SET 715
  - C.4.1 Move, Copy and Arithmetic 715
  - C.4.2 Logical, Bit and Shift Operations 718
  - C.4.3 Loop and Repetitive String Operations 718
  - C.4.4 Jump and Call Instructions 719
  - C.4.5 Subroutine Calls 721
  - C.4.6 System Calls and System Subroutines 723
  - C.4.7 Final Remarks on the Instruction Set 725
  
- C.5 THE ASSEMBLER 725
  - C.5.1 Introduction 726
  - C.5.2 The ACK-Based Tutorial Assembler as88 727
  - C.5.3 Some Differences with Other 8088 Assemblers 730
  
- C.6 THE TRACER 732
  - C.6.1 Tracer Commands 734
  
- C.7 GETTING STARTED 735
  
- C.8 EXAMPLES 736
  - C.8.1 Hello World Example 736
  - C.8.2 General Registers Example 740
  - C.8.3 Call Command and Pointer Registers 742
  - C.8.4 Debugging an Array Print Program 744
  - C.8.5 String Manipulation and String Instructions 748
  - C.8.6 Dispatch Tables 750
  - C.8.7 Buffered and Random File Access 752