

# CONTENTS

Preface xvii

## Chapter 1

### BASIC STRUCTURE OF COMPUTERS 1

- 1.1 Computer Types 2
- 1.2 Functional Units 3
  - 1.2.1 Input Unit 4
  - 1.2.2 Memory Unit 4
  - 1.2.3 Arithmetic and Logic Unit 5
  - 1.2.4 Output Unit 6
  - 1.2.5 Control Unit 6
- 1.3 Basic Operational Concepts 7
- 1.4 Bus Structures 9
- 1.5 Software 10
- 1.6 Performance 13
  - 1.6.1 Processor Clock 14
  - 1.6.2 Basic Performance Equation 14
  - 1.6.3 Pipelining and Superscalar Operation 15
  - 1.6.4 Clock Rate 16
  - 1.6.5 Instruction Set: CISC and RISC 16
  - 1.6.6 Compiler 17
  - 1.6.7 Performance Measurement 17
- 1.7 Multiprocessors and Multicomputers 18
- 1.8 Historical Perspective 19
  - 1.8.1 The First Generation 19
  - 1.8.2 The Second Generation 20
  - 1.8.3 The Third Generation 20
  - 1.8.4 The Fourth Generation 20
  - 1.8.5 Beyond the Fourth Generation 21
  - 1.8.6 Evolution of Performance 21
- 1.9 Concluding Remarks 21
  - Problems 22
  - References 23

## Chapter 2

### MACHINE INSTRUCTIONS AND PROGRAMS 25

- 2.1 Numbers, Arithmetic Operations, and Characters 27

- 2.1.1 Number Representation 27
- 2.1.2 Addition of Positive Numbers 28
- 2.1.3 Addition and Subtraction of Signed Numbers 29
- 2.1.4 Overflow in Integer Arithmetic 32
- 2.1.5 Characters 33
- 2.2 Memory Locations and Addresses 33
  - 2.2.1 Byte Addressability 33
  - 2.2.2 Big-endian and Little-endian Assignments 35
  - 2.2.3 Word Alignment 36
  - 2.2.4 Accessing Numbers, Characters, and Character Strings 36
- 2.3 Memory Operations 36
- 2.4 Instructions and Instruction Sequencing 37
  - 2.4.1 Register Transfer Notation 37
  - 2.4.2 Assembly Language Notation 38
  - 2.4.3 Basic Instruction Types 38
  - 2.4.4 Instruction Execution and Straight-Line Sequencing 42
  - 2.4.5 Branching 44
  - 2.4.6 Condition Codes 46
  - 2.4.7 Generating Memory Addresses 47
- 2.5 Addressing Modes 48
  - 2.5.1 Implementation of Variables and Constants 49
  - 2.5.2 Indirection and Pointers 50
  - 2.5.3 Indexing and Arrays 52
  - 2.5.4 Relative Addressing 56
  - 2.5.5 Additional Modes 56
- 2.6 Assembly Language 58
  - 2.6.1 Assembler Directives 59
  - 2.6.2 Assembly and Execution of Programs 62
  - 2.6.3 Number Notation 64
- 2.7 Basic Input/Output Operations 64
- 2.8 Stacks and Queues 68
- 2.9 Subroutines 72
  - 2.9.1 Subroutine Nesting and the Processor Stack 73
  - 2.9.2 Parameter Passing 74
  - 2.9.3 The Stack Frame 75
- 2.10 Additional Instructions 81

- 2.10.1 Logic Instructions 81
- 2.10.2 Shift and Rotate Instructions 82
- 2.10.3 Multiplication and Division 86
- 2.11 Example Programs 86
  - 2.11.1 Vector Dot Product Program 86
  - 2.11.2 Byte-Sorting Program 87
  - 2.11.3 Linked Lists 89
- 2.12 Encoding of Machine Instructions 94
- 2.13 Concluding Remarks 98
  - Problems 98

### Chapter 3

## ARM, MOTOROLA, AND INTEL INSTRUCTION SETS 103

### Part I The ARM Example 104

- 3.1 Registers, Memory Access, and Data Transfer 104
  - 3.1.1 Register Structure 105
  - 3.1.2 Memory Access Instructions and Addressing Modes 106
  - 3.1.3 Register Move Instructions 113
- 3.2 Arithmetic and Logic Instructions 113
  - 3.2.1 Arithmetic Instructions 113
  - 3.2.2 Logic Instructions 115
- 3.3 Branch Instructions 116
  - 3.3.1 Setting Condition Codes 117
  - 3.3.2 A Loop Program for Adding Numbers 118
- 3.4 Assembly Language 118
  - 3.4.1 Pseudo-Instructions 120
- 3.5 I/O Operations 121
- 3.6 Subroutines 122
- 3.7 Program Examples 126
  - 3.7.1 Vector Dot Product Program 126
  - 3.7.2 Byte-Sorting Program 127
  - 3.7.3 Linked-List Insertion and Deletion Subroutines 127

### Part II The 68000 Example 130

- 3.8 Registers and Addressing 131
  - 3.8.1 The 68000 Register Structure 131
  - 3.8.2 Addressing 131
- 3.9 Instructions 136
- 3.10 Assembly Language 140
- 3.11 Program Flow Control 141

- 3.11.1 Condition Code Flags 141
- 3.11.2 Branch Instructions 141
- 3.12 I/O Operations 145
- 3.13 Stacks and Subroutines 146
- 3.14 Logic Instructions 151
- 3.15 Program Examples 152
  - 3.15.1 Vector Dot Product Program 152
  - 3.15.2 Byte-Sorting Program 153
  - 3.15.3 Linked-List Insertion and Deletion Subroutines 154

### Part III The IA-32 Pentium Example 155

- 3.16 Registers and Addressing 156
  - 3.16.1 IA-32 Register Structure 156
  - 3.16.2 IA-32 Addressing Modes 159
- 3.17 IA-32 Instructions 164
  - 3.17.1 Machine Instruction Format 168
- 3.18 IA-32 Assembly Language 170
- 3.19 Program Flow Control 171
  - 3.19.1 Conditional Jumps and Condition Code Flags 171
  - 3.19.2 Unconditional Jump 173
- 3.20 Logic and Shift/Rotate Instructions 173
  - 3.20.1 Logic Operations 173
  - 3.20.2 Shift and Rotate Operations 173
- 3.21 I/O Operations 174
  - 3.21.1 Memory-Mapped I/O 174
  - 3.21.2 Isolated I/O 175
  - 3.21.3 Block Transfers 176
- 3.22 Subroutines 177
- 3.23 Other Instructions 182
  - 3.23.1 Multiply and Divide Instructions 182
  - 3.23.2 Multimedia Extension (MMX) Instructions 183
  - 3.23.3 Vector (SIMD) Instructions 184
- 3.24 Program Examples 184
  - 3.24.1 Vector Dot Product Program 184
  - 3.24.2 Byte-Sorting Program 185
  - 3.24.3 Linked-List Insertion and Deletion Subroutines 185
- 3.25 Concluding Remarks 188
  - Problems 188
  - References 201

### Chapter 4

## INPUT/OUTPUT ORGANIZATION 203

- 4.1 Accessing I/O Devices 204
- 4.2 Interrupts 208

- 4.2.1 Interrupt Hardware 210
- 4.2.2 Enabling and Disabling Interrupts 211
- 4.2.3 Handling Multiple Devices 213
- 4.2.4 Controlling Device Requests 217
- 4.2.5 Exceptions 218
- 4.2.6 Use of Interrupts in Operating Systems 220
- 4.3 Processor Examples 224
  - 4.3.1 ARM Interrupt Structure 224
  - 4.3.2 68000 Interrupt Structure 229
  - 4.3.3 Pentium Interrupt Structure 231
- 4.4 Direct Memory Access 234
  - 4.4.1 Bus Arbitration 237
- 4.5 Buses 240
  - 4.5.1 Synchronous Bus 241
  - 4.5.2 Asynchronous Bus 244
  - 4.5.3 Discussion 247
- 4.6 Interface Circuits 248
  - 4.6.1 Parallel Port 248
  - 4.6.2 Serial Port 257
- 4.7 Standard I/O Interfaces 259
  - 4.7.1 Peripheral Component Interconnect (PCI) Bus 261
  - 4.7.2 SCSI Bus 266
  - 4.7.3 Universal Serial Bus (USB) 272
- 4.8 Concluding Remarks 283
  - Problems 283
  - References 289

## Chapter 5 THE MEMORY SYSTEM 291

- 5.1 Some Basic Concepts 292
- 5.2 Semiconductor RAM Memories 295
  - 5.2.1 Internal Organization of Memory Chips 295
  - 5.2.2 Static Memories 297
  - 5.2.3 Asynchronous Drams 299
  - 5.2.4 Synchronous DRAMs 302
  - 5.2.5 Structure of Larger Memories 305
  - 5.2.6 Memory System Considerations 307
  - 5.2.7 Rambus Memory 308
- 5.3 Read-Only Memories 309
  - 5.3.1 ROM 310
  - 5.3.2 PROM 311
  - 5.3.3 EPROM 311
  - 5.3.4 EEPROM 311
  - 5.3.5 Flash Memory 312
- 5.4 Speed, Size, and Cost 313
- 5.5 Cache Memories 314
  - 5.5.1 Mapping Functions 316
  - 5.5.2 Replacement Algorithms 321
  - 5.5.3 Example of Mapping Techniques 322
  - 5.5.4 Examples of Caches in Commercial Processors 325
- 5.6 Performance Considerations 329
  - 5.6.1 Interleaving 330
  - 5.6.2 Hit Rate and Miss Penalty 332
  - 5.6.3 Caches on the Processor Chip 334
  - 5.6.4 Other Enhancements 335
- 5.7 Virtual Memories 337
  - 5.7.1 Address Translation 339
- 5.8 Memory Management Requirements 343
- 5.9 Secondary Storage 344
  - 5.9.1 Magnetic Hard Disks 344
  - 5.9.2 Optical Disks 352
  - 5.9.3 Magnetic Tape Systems 358
- 5.10 Concluding Remarks 359
  - Problems 360
  - References 366

## Chapter 6 ARITHMETIC 367

- 6.1 Addition and Subtraction of Signed Numbers 368
  - 6.1.1 Addition/Subtraction Logic Unit 369
- 6.2 Design of Fast Adders 371
  - 6.2.1 Carry-Lookahead Addition 372
- 6.3 Multiplication of Positive Numbers 376
- 6.4 Signed-Operand Multiplication 380
  - 6.4.1 Booth Algorithm 380
- 6.5 Fast Multiplication 383
  - 6.5.1 Bit-Pair Recoding of Multipliers 384
  - 6.5.2 Carry-Save Addition of Summands 385
- 6.6 Integer Division 390
- 6.7 Floating-Point Numbers and Operations 393
  - 6.7.1 IEEE Standard for Floating-Point Numbers 394
  - 6.7.2 Arithmetic Operations on Floating-Point Numbers 398
  - 6.7.3 Guard Bits and Truncation 399

- 6.7.4 Implementing Floating-Point Operations 400
- 6.8 Concluding Remarks 403
  - Problems 403
  - References 410

## Chapter 7

### BASIC PROCESSING UNIT 411

- 7.1 Some Fundamental Concepts 412
  - 7.1.1 Register Transfers 415
  - 7.1.2 Performing an Arithmetic or Logic Operation 415
  - 7.1.3 Fetching a Word from Memory 418
  - 7.1.4 Storing a Word in Memory 420
- 7.2 Execution of a Complete Instruction 421
  - 7.2.1 Branch Instructions 422
- 7.3 Multiple-Bus Organization 423
- 7.4 Hardwired Control 425
  - 7.4.1 A Complete Processor 428
- 7.5 Microprogrammed Control 429
  - 7.5.1 Microinstructions 432
  - 7.5.2 Microprogram Sequencing 435
  - 7.5.3 Wide-Branch Addressing 437
  - 7.5.4 Microinstructions with Next-Address Field 440
  - 7.5.5 Prefetching Microinstructions 443
  - 7.5.6 Emulation 443
- 7.6 Concluding Remarks 445
  - Problems 446

## Chapter 8

### PIPELINING 453

- 8.1 Basic Concepts 454
  - 8.1.1 Role of Cache Memory 456
  - 8.1.2 Pipeline Performance 458
- 8.2 Data Hazards 461
  - 8.2.1 Operand Forwarding 462
  - 8.2.2 Handling Data Hazards in Software 464
  - 8.2.3 Side Effects 464
- 8.3 Instruction Hazards 465
  - 8.3.1 Unconditional Branches 466
  - 8.3.2 Conditional Branches and Branch Prediction 470
- 8.4 Influence on Instruction Sets 476
  - 8.4.1 Addressing Modes 476
  - 8.4.2 Condition Codes 478

- 8.5 Datapath and Control Considerations 479
- 8.6 Superscalar Operation 481
  - 8.6.1 Out-of-Order Execution 483
  - 8.6.2 Execution Completion 485
  - 8.6.3 Dispatch Operation 486
- 8.7 UltraSPARC II EXAMPLE 486
  - 8.7.1 SPARC Architecture 487
  - 8.7.2 UltraSPARC II 493
  - 8.7.3 Pipeline Structure 493
- 8.8 Performance Considerations 503
  - 8.8.1 Effect of Instruction Hazards 504
  - 8.8.2 Number of Pipeline Stages 505
- 8.9 Concluding Remarks 506
  - Problems 506
  - Reference 509

## Chapter 9

### EMBEDDED SYSTEMS 511

- 9.1 Examples of Embedded Systems 512
  - 9.1.1 Microwave Oven 512
  - 9.1.2 Digital Camera 514
  - 9.1.3 Home Telemetry 516
- 9.2 Processor Chips for Embedded Applications 517
- 9.3 A Simple Microcontroller 518
  - 9.3.1 Parallel I/O Ports 518
  - 9.3.2 Serial I/O Interface 521
  - 9.3.3 Counter/Timer 523
  - 9.3.4 Interrupt Control Mechanism 525
- 9.4 Programming Considerations 525
  - 9.4.1 Polling Approach 526
  - 9.4.2 Interrupt Approach 529
- 9.5 I/O Device Timing Constraints 531
  - 9.5.1 C Program for Transfer via a Circular Buffer 533
  - 9.5.2 Assembly Language Program for Transfer via a Circular Buffer 534
- 9.6 Reaction Timer — An Example 535
  - 9.6.1 C Program for the Reaction Timer 537
  - 9.6.2 Assembly Language Program for the Reaction Timer 537
  - 9.6.3 Final Comments 541
- 9.7 Embedded Processor Families 541
  - 9.7.1 Microcontrollers Based on the Intel 8051 542
  - 9.7.2 Motorola Microcontrollers 542
  - 9.7.3 ARM Microcontrollers 543

- 9.8 Design Issues 544
- 9.9 System-on-a-Chip 546
  - 9.9.1 FPGA Implementation 547
- 9.10 Concluding Remarks 549
  - Problems 550
  - References 552

## Chapter 10

### COMPUTER PERIPHERALS 553

- 10.1 Input Devices 554
  - 10.1.1 Keyboard 554
  - 10.1.2 Mouse 555
  - 10.1.3 Trackball, Joystick, and Touchpad 556
  - 10.1.4 Scanners 557
- 10.2 Output Devices 558
  - 10.2.1 Video Displays 558
  - 10.2.2 Flat-Panel Displays 559
  - 10.2.3 Printers 560
  - 10.2.4 Graphics Accelerators 561
- 10.3 Serial Communication Links 563
  - 10.3.1 Asynchronous Transmission 566
  - 10.3.2 Synchronous Transmission 568
  - 10.3.3 Standard Communications Interfaces 571
- 10.4 Concluding Remarks 574
  - Problems 575

## Chapter 11

### PROCESSOR FAMILIES 577

- 11.1 The ARM Family 579
  - 11.1.1 The Thumb Instruction Set 579
  - 11.1.2 Processor and CPU Cores 580
- 11.2 The Motorola 680X0 and ColdFire Families 582
  - 11.2.1 68020 Processor 582
  - 11.2.2 Enhancements in 68030 and 68040 Processors 584
  - 11.2.3 68060 Processor 585
  - 11.2.4 The ColdFire Family 585
- 11.3 The Intel IA-32 Family 585
  - 11.3.1 IA-32 Memory Segmentation 586
  - 11.3.2 Sixteen-Bit Mode 588
  - 11.3.3 80386 and 80486 Processors 588
  - 11.3.4 Pentium Processor 589
  - 11.3.5 Pentium Pro Processor 589
  - 11.3.6 Pentium II and III Processors 590

- 11.3.7 Pentium 4 Processor 590
- 11.3.8 Advanced Micro Devices IA-32 Processors 591
- 11.4 The PowerPC Family 591
  - 11.4.1 Register Set 591
  - 11.4.2 Memory Addressing Modes 592
  - 11.4.3 Instructions 592
  - 11.4.4 PowerPC Processors 592
- 11.5 The Sun Microsystems SPARC Family 594
- 11.6 The Compaq Alpha Family 596
  - 11.6.1 Instruction and Addressing Mode Formats 596
  - 11.6.2 Alpha 21064 Processor 597
  - 11.6.3 Alpha 21164 Processor 597
  - 11.6.4 Alpha 21264 Processor 597
- 11.7 The Intel IA-64 Family 598
  - 11.7.1 Instruction Bundles 598
  - 11.7.2 Conditional Execution 598
  - 11.7.3 Speculative Loads 600
  - 11.7.4 Registers and the Register Stack 600
  - 11.7.5 Itanium Processor 602
- 11.8 A Stack Processor 603
  - 11.8.1 Stack Structure 604
  - 11.8.2 Stack Instructions 606
  - 11.8.3 Hardware Registers in the Stack 610
- 11.9 Concluding Remarks 612
  - Problems 612
  - References 614

## Chapter 12

### LARGE COMPUTER SYSTEMS 617

- 12.1 Forms of Parallel Processing 619
  - 12.1.1 Classification of Parallel Structures 619
- 12.2 Array Processors 620
- 12.3 The Structure of General-Purpose Multiprocessors 622
- 12.4 Interconnection Networks 624
  - 12.4.1 Single Bus 624
  - 12.4.2 Crossbar Networks 625
  - 12.4.3 Multistage Networks 626
  - 12.4.4 Hypercube Networks 628
  - 12.4.5 Mesh Networks 630
  - 12.4.6 Tree Networks 630
  - 12.4.7 Ring Networks 631
  - 12.4.8 Practical Considerations 632
  - 12.4.9 Mixed Topology Networks 636
  - 12.4.10 Symmetric Multiprocessors 636

- 12.5 Memory Organization in Multiprocessors 637
- 12.6 Program Parallelism and Shared Variables 638
  - 12.6.1 Accessing Shared Variables 640
  - 12.6.2 Cache Coherence 641
  - 12.6.3 Need for Locking and Cache Coherence 645
- 12.7 Multicomputers 645
  - 12.7.1 Local Area Networks 646
  - 12.7.2 Ethernet (CSMA/CD) Bus 646
  - 12.7.3 Token Ring 647
  - 12.7.4 Network of Workstations 647
- 12.8 Programmer's View of Shared Memory and Message Passing 648
  - 12.8.1 Shared Memory Case 648
  - 12.8.2 Message-Passing Case 651
- 12.9 Performance Considerations 653
  - 12.9.1 Amdahl's Law 654
  - 12.9.2 Performance Indicators 656
- 12.10 Concluding Remarks 656
  - Problems 657
  - References 660

## APPENDIX A: LOGIC CIRCUITS 661

- A.1 Basic Logic Functions 662
  - A.1.1 Electronic Logic Gates 665
- A.2 Synthesis of Logic Functions 666
- A.3 Minimization of Logic Expressions 668
  - A.3.1 Minimization Using Karnaugh Maps 671
  - A.3.2 Don't-Care Conditions 674
- A.4 Synthesis with NAND and NOR Gates 674
- A.5 Practical Implementation of Logic Gates 678
  - A.5.1 CMOS Circuits 681
  - A.5.2 Propagation Delay 686
  - A.5.3 Fan-In and Fan-Out Constraints 687
  - A.5.4 Tri-state Buffers 687
  - A.5.5 Integrated Circuit Packages 688
- A.6 Flip-Flops 690
  - A.6.1 Gated Latches 690
  - A.6.2 Master-Slave Flip-Flop 694
  - A.6.3 Edge Triggering 694
  - A.6.4 T Flip-Flop 697

- A.6.5 JK Flip-Flop 697
- A.6.6 Flip-Flops with Preset and Clear 698
- A.7 Registers and Shift Registers 699
- A.8 Counters 702
- A.9 Decoders 703
- A.10 Multiplexers 705
- A.11 Programmable Logic Devices (PLDs) 705
  - A.11.1 Programmable Logic Array (PLA) 707
  - A.11.2 Programmable Array Logic (PAL) 710
  - A.11.3 Complex Programmable Logic Devices (CPLDs) 711
- A.12 Field-Programmable Gate Arrays 712
- A.13 Sequential Circuits 714
  - A.13.1 An Example of an Up/Down Counter 714
  - A.13.2 Timing Diagrams 718
  - A.13.3 The Finite State Machine Model 719
  - A.13.4 Synthesis of Finite State Machines 720
- A.14 Concluding Remarks 724
  - Problems 724
  - References 731

## APPENDIX B: ARM INSTRUCTION SET 733

- B.1 Instruction Encoding 734
  - B.1.1 Arithmetic and Logic Instructions 734
  - B.1.2 Memory Load and Store Instructions 741
  - B.1.3 Block Load and Store Instructions 744
  - B.1.4 Branch and Branch with Link Instructions 747
  - B.1.5 Machine Control Instructions 747
- B.2 Other ARM Instructions 750
  - B.2.1 Coprocessor Instructions 750
  - B.2.2 Versions v4 and v5 Instructions 750
- B.3 Programming Experiments 750

## APPENDIX C: MOTOROLA 68000 INSTRUCTION SET 751

**APPENDIX D: INTEL IA-32  
INSTRUCTION SET 769**

- D.1 Instruction Encoding 770
  - D.1.1 Addressing Modes 772
- D.2 Basic Instructions 773
  - D.2.1 Conditional Jump Instructions 782
  - D.2.2 Unconditional Jump Instructions 782
- D.3 Prefix Bytes 782
- D.4 Other Instructions 783
  - D.4.1 String Instructions 783

- D.4.2 Floating-Point, MMX, and SSE  
Instructions 784

- D.5 Sixteen-Bit Operation 785
- D.6 Programming Experiments 785

**APPENDIX E: CHARACTER CODES AND  
NUMBER CONVERSION 789**

- E.1 Character Codes 790
- E.2 Decimal-to-Binary Conversion 793

**INDEX 795**