

CONTENTS

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 Number Representation and Arithmetic Operations 9
 - 1.4.1 Integers 10
 - 1.4.2 Floating-Point Numbers 16
- 1.5 Character Representation 17
- 1.6 Performance 17
 - 1.6.1 Technology 17
 - 1.6.2 Parallelism 19
- 1.7 Historical Perspective 19
 - 1.7.1 The First Generation 20
 - 1.7.2 The Second Generation 20
 - 1.7.3 The Third Generation 21
 - 1.7.4 The Fourth Generation 21
- 1.8 Concluding Remarks 22
- 1.9 Solved Problems 22
 - Problems 24
 - References 25

Chapter 2

INSTRUCTION SET ARCHITECTURE 27

- 2.1 Memory Locations and Addresses 28
 - 2.1.1 Byte Addressability 30
 - 2.1.2 Big-Endian and Little-Endian Assignments 30
 - 2.1.3 Word Alignment 31
 - 2.1.4 Accessing Numbers and Characters 32
- 2.2 Memory Operations 32

- 2.3 Instructions and Instruction Sequencing 32
 - 2.3.1 Register Transfer Notation 33
 - 2.3.2 Assembly-Language Notation 33
 - 2.3.3 RISC and CISC Instruction Sets 34
 - 2.3.4 Introduction to RISC Instruction Sets 34
 - 2.3.5 Instruction Execution and Straight-Line Sequencing 36
 - 2.3.6 Branching 37
 - 2.3.7 Generating Memory Addresses 40
- 2.4 Addressing Modes 40
 - 2.4.1 Implementation of Variables and Constants 41
 - 2.4.2 Indirection and Pointers 42
 - 2.4.3 Indexing and Arrays 45
- 2.5 Assembly Language 48
 - 2.5.1 Assembler Directives 50
 - 2.5.2 Assembly and Execution of Programs 53
 - 2.5.3 Number Notation 54
- 2.6 Stacks 55
- 2.7 Subroutines 56
 - 2.7.1 Subroutine Nesting and the Processor Stack 58
 - 2.7.2 Parameter Passing 59
 - 2.7.3 The Stack Frame 63
- 2.8 Additional Instructions 65
 - 2.8.1 Logic Instructions 67
 - 2.8.2 Shift and Rotate Instructions 68
 - 2.8.3 Multiplication and Division 71
- 2.9 Dealing with 32-Bit Immediate Values 73
- 2.10 CISC Instruction Sets 74
 - 2.10.1 Additional Addressing Modes 75
 - 2.10.2 Condition Codes 77
- 2.11 RISC and CISC Styles 78
- 2.12 Example Programs 79
 - 2.12.1 Vector Dot Product Program 79
 - 2.12.2 String Search Program 81
- 2.13 Encoding of Machine Instructions 82
- 2.14 Concluding Remarks 85
- 2.15 Solved Problems 85
 - Problems 90

Chapter 3 BASIC INPUT/OUTPUT 95

- 3.1 Accessing I/O Devices 96
 - 3.1.1 I/O Device Interface 97
 - 3.1.2 Program-Controlled I/O 97
 - 3.1.3 An Example of a RISC-Style I/O Program 101
 - 3.1.4 An Example of a CISC-Style I/O Program 101
- 3.2 Interrupts 103
 - 3.2.1 Enabling and Disabling Interrupts 106
 - 3.2.2 Handling Multiple Devices 107
 - 3.2.3 Controlling I/O Device Behavior 109
 - 3.2.4 Processor Control Registers 110
 - 3.2.5 Examples of Interrupt Programs 111
 - 3.2.6 Exceptions 116
- 3.3 Concluding Remarks 119
- 3.4 Solved Problems 119
 - Problems 126

Chapter 4 SOFTWARE 129

- 4.1 The Assembly Process 130
 - 4.1.1 Two-pass Assembler 131
- 4.2 Loading and Executing Object Programs 131
- 4.3 The Linker 132
- 4.4 Libraries 133
- 4.5 The Compiler 133
 - 4.5.1 Compiler Optimizations 134
 - 4.5.2 Combining Programs Written in Different Languages 134
- 4.6 The Debugger 134
- 4.7 Using a High-level Language for I/O Tasks 137
- 4.8 Interaction between Assembly Language and C Language 139
- 4.9 The Operating System 143
 - 4.9.1 The Boot-strapping Process 144
 - 4.9.2 Managing the Execution of Application Programs 144
 - 4.9.3 Use of Interrupts in Operating Systems 146
- 4.10 Concluding Remarks 149
 - Problems 149
 - References 150

Chapter 5 BASIC PROCESSING UNIT 151

- 5.1 Some Fundamental Concepts 152
- 5.2 Instruction Execution 155
 - 5.2.1 Load Instructions 155
 - 5.2.2 Arithmetic and Logic Instructions 156
 - 5.2.3 Store Instructions 157
- 5.3 Hardware Components 158
 - 5.3.1 Register File 158
 - 5.3.2 ALU 160
 - 5.3.3 Datapath 161
 - 5.3.4 Instruction Fetch Section 164
- 5.4 Instruction Fetch and Execution Steps 165
 - 5.4.1 Branching 168
 - 5.4.2 Waiting for Memory 171
- 5.5 Control Signals 172
- 5.6 Hardwired Control 175
 - 5.6.1 Datapath Control Signals 177
 - 5.6.2 Dealing with Memory Delay 177
- 5.7 CISC-Style Processors 178
 - 5.7.1 An Interconnect using Buses 180
 - 5.7.2 Microprogrammed Control 183
- 5.8 Concluding Remarks 185
- 5.9 Solved Problems 185
 - Problems 188

Chapter 6 PIPELINING 193

- 6.1 Basic Concept—The Ideal Case 194
- 6.2 Pipeline Organization 195
- 6.3 Pipelining Issues 196
- 6.4 Data Dependencies 197
 - 6.4.1 Operand Forwarding 198
 - 6.4.2 Handling Data Dependencies in Software 199
- 6.5 Memory Delays 201
- 6.6 Branch Delays 202
 - 6.6.1 Unconditional Branches 202
 - 6.6.2 Conditional Branches 204
 - 6.6.3 The Branch Delay Slot 204
 - 6.6.4 Branch Prediction 205
- 6.7 Resource Limitations 209
- 6.8 Performance Evaluation 209
 - 6.8.1 Effects of Stalls and Penalties 210
 - 6.8.2 Number of Pipeline Stages 212

- 6.9 Superscalar Operation 212
 - 6.9.1 Branches and Data Dependencies 214
 - 6.9.2 Out-of-Order Execution 215
 - 6.9.3 Execution Completion 216
 - 6.9.4 Dispatch Operation 217
- 6.10 Pipelining in CISC Processors 218
 - 6.10.1 Pipelining in ColdFire Processors 219
 - 6.10.2 Pipelining in Intel Processors 219
- 6.11 Concluding Remarks 220
- 6.12 Examples of Solved Problems 220
 - Problems 222
 - References 226

Chapter 7

INPUT/OUTPUT ORGANIZATION 227

- 7.1 Bus Structure 228
- 7.2 Bus Operation 229
 - 7.2.1 Synchronous Bus 230
 - 7.2.2 Asynchronous Bus 233
 - 7.2.3 Electrical Considerations 236
- 7.3 Arbitration 237
- 7.4 Interface Circuits 238
 - 7.4.1 Parallel Interface 239
 - 7.4.2 Serial Interface 243
- 7.5 Interconnection Standards 247
 - 7.5.1 Universal Serial Bus (USB) 247
 - 7.5.2 FireWire 251
 - 7.5.3 PCI Bus 252
 - 7.5.4 SCSI Bus 256
 - 7.5.5 SATA 258
 - 7.5.6 SAS 258
 - 7.5.7 PCI Express 258
- 7.6 Concluding Remarks 260
- 7.7 Solved Problems 260
 - Problems 263
 - References 266

Chapter 8

THE MEMORY SYSTEM 267

- 8.1 Basic Concepts 268
- 8.2 Semiconductor RAM Memories 270
 - 8.2.1 Internal Organization of Memory Chips 270
 - 8.2.2 Static Memories 271
 - 8.2.3 Dynamic RAMs 274

- 8.2.4 Synchronous DRAMs 276
- 8.2.5 Structure of Larger Memories 279
- 8.3 Read-only Memories 282
 - 8.3.1 ROM 283
 - 8.3.2 PROM 283
 - 8.3.3 EPROM 284
 - 8.3.4 EEPROM 284
 - 8.3.5 Flash Memory 284
- 8.4 Direct Memory Access 285
- 8.5 Memory Hierarchy 288
- 8.6 Cache Memories 289
 - 8.6.1 Mapping Functions 291
 - 8.6.2 Replacement Algorithms 296
 - 8.6.3 Examples of Mapping Techniques 297
- 8.7 Performance Considerations 300
 - 8.7.1 Hit Rate and Miss Penalty 301
 - 8.7.2 Caches on the Processor Chip 302
 - 8.7.3 Other Enhancements 303
- 8.8 Virtual Memory 305
 - 8.8.1 Address Translation 306
- 8.9 Memory Management Requirements 310
- 8.10 Secondary Storage 311
 - 8.10.1 Magnetic Hard Disks 311
 - 8.10.2 Optical Disks 317
 - 8.10.3 Magnetic Tape Systems 322
- 8.11 Concluding Remarks 323
- 8.12 Solved Problems 324
 - Problems 328
 - References 332

Chapter 9

ARITHMETIC 335

- 9.1 Addition and Subtraction of Signed Numbers 336
 - 9.1.1 Addition/Subtraction Logic Unit 336
- 9.2 Design of Fast Adders 339
 - 9.2.1 Carry-Lookahead Addition 340
- 9.3 Multiplication of Unsigned Numbers 344
 - 9.3.1 Array Multiplier 344
 - 9.3.2 Sequential Circuit Multiplier 346
- 9.4 Multiplication of Signed Numbers 346
 - 9.4.1 The Booth Algorithm 348
- 9.5 Fast Multiplication 351
 - 9.5.1 Bit-Pair Recoding of Multipliers 352
 - 9.5.2 Carry-Save Addition of Summands 353

- 9.5.3 Summand Addition Tree using 3-2 Reducers 355
- 9.5.4 Summand Addition Tree using 4-2 Reducers 357
- 9.5.5 Summary of Fast Multiplication 359
- 9.6 Integer Division 360
- 9.7 Floating-Point Numbers and Operations 363
 - 9.7.1 Arithmetic Operations on Floating-Point Numbers 367
 - 9.7.2 Guard Bits and Truncation 368
 - 9.7.3 Implementing Floating-Point Operations 369
- 9.8 Decimal-to-Binary Conversion 372
- 9.9 Concluding Remarks 372
- 9.10 Solved Problems 374
 - Problems 377
 - References 383

Chapter 10

EMBEDDED SYSTEMS 385

- 10.1 Examples of Embedded Systems 386
 - 10.1.1 Microwave Oven 386
 - 10.1.2 Digital Camera 387
 - 10.1.3 Home Telemetry 390
- 10.2 Microcontroller Chips for Embedded Applications 390
- 10.3 A Simple Microcontroller 392
 - 10.3.1 Parallel I/O Interface 392
 - 10.3.2 Serial I/O Interface 395
 - 10.3.3 Counter/Timer 397
 - 10.3.4 Interrupt-Control Mechanism 399
 - 10.3.5 Programming Examples 399
- 10.4 Reaction Timer—A Complete Example 401
- 10.5 Sensors and Actuators 407
 - 10.5.1 Sensors 407
 - 10.5.2 Actuators 410
 - 10.5.3 Application Examples 411
- 10.6 Microcontroller Families 412
 - 10.6.1 Microcontrollers Based on the Intel 8051 413
 - 10.6.2 Freescale Microcontrollers 413
 - 10.6.3 ARM Microcontrollers 414
- 10.7 Design Issues 414
- 10.8 Concluding Remarks 417
 - Problems 418
 - References 420

Chapter 11

SYSTEM-ON-A-CHIP—A CASE STUDY 421

- 11.1 FPGA Implementation 422
 - 11.1.1 FPGA Devices 423
 - 11.1.2 Processor Choice 423
- 11.2 Computer-Aided Design Tools 424
 - 11.2.1 Altera CAD Tools 425
- 11.3 Alarm Clock Example 428
 - 11.3.1 User's View of the System 428
 - 11.3.2 System Definition and Generation 429
 - 11.3.3 Circuit Implementation 430
 - 11.3.4 Application Software 431
- 11.4 Concluding Remarks 440
 - Problems 440
 - References 441

Chapter 12

PARALLEL PROCESSING AND PERFORMANCE 443

- 12.1 Hardware Multithreading 444
- 12.2 Vector (SIMD) Processing 445
 - 12.2.1 Graphics Processing Units (GPUs) 448
- 12.3 Shared-Memory Multiprocessors 448
 - 12.3.1 Interconnection Networks 450
- 12.4 Cache Coherence 453
 - 12.4.1 Write-Through Protocol 453
 - 12.4.2 Write-Back protocol 454
 - 12.4.3 Snoopy Caches 454
 - 12.4.4 Directory-Based Cache Coherence 456
- 12.5 Message-Passing Multicomputers 456
- 12.6 Parallel Programming for Multiprocessors 456
- 12.7 Performance Modeling 460
- 12.8 Concluding Remarks 461
 - Problems 462
 - References 463

Appendix A

LOGIC CIRCUITS 465

- A.1 Basic Logic Functions
 - A.1.1 Electronic Logic Gates 469
- A.2 Synthesis of Logic Functions 470

- A.3 Minimization of Logic Expressions 472
 - A.3.1 Minimization using Karnaugh Maps 475
 - A.3.2 Don't-Care Conditions 477
 - A.4 Synthesis with NAND and NOR Gates 479
 - A.5 Practical Implementation of Logic Gates 482
 - A.5.1 CMOS Circuits 484
 - A.5.2 Propagation Delay 489
 - A.5.3 Fan-In and Fan-Out Constraints 490
 - A.5.4 Tri-State Buffers 491
 - A.6 Flip-Flops 492
 - A.6.1 Gated Latches 493
 - A.6.2 Master-Slave Flip-Flop 495
 - A.6.3 Edge Triggering 498
 - A.6.4 T Flip-Flop 498
 - A.6.5 JK Flip-Flop 499
 - A.6.6 Flip-Flops with Preset and Clear 501
 - A.7 Registers and Shift Registers 502
 - A.8 Counters 503
 - A.9 Decoders 505
 - A.10 Multiplexers 506
 - A.11 Programmable Logic Devices (PLDs) 509
 - A.11.1 Programmable Logic Array (PLA) 509
 - A.11.2 Programmable Array Logic (PAL) 511
 - A.11.3 Complex Programmable Logic Devices (CPLDs) 512
 - A.12 Field-Programmable Gate Arrays 514
 - A.13 Sequential Circuits 516
 - A.13.1 Design of an Up/Down Counter as a Sequential Circuit 516
 - A.13.2 Timing Diagrams 519
 - A.13.3 The Finite State Machine Model 520
 - A.13.4 Synthesis of Finite State Machines 521
 - A.14 Concluding Remarks 522
 - Problems 522
 - References 528
- Appendix B**
THE ALTERA NIOS II PROCESSOR 529
- B.1 Nios II Characteristics 530
 - B.2 General-Purpose Registers 531
 - B.3 Addressing Modes 532
 - B.4 Instructions 533
 - B.4.1 Notation 533
 - B.4.2 Load and Store Instructions 534
 - B.4.3 Arithmetic Instructions 536
 - B.4.4 Logic Instructions 537
 - B.4.5 Move Instructions 537
 - B.4.6 Branch and Jump Instructions 538
 - B.4.7 Subroutine Linkage Instructions 541
 - B.4.8 Comparison Instructions 545
 - B.4.9 Shift Instructions 546
 - B.4.10 Rotate Instructions 547
 - B.4.11 Control Instructions 548
 - B.5 Pseudoinstructions 548
 - B.6 Assembler Directives 549
 - B.7 Carry and Overflow Detection 551
 - B.8 Example Programs 553
 - B.9 Control Registers 553
 - B.10 Input/Output 555
 - B.10.1 Program-Controlled I/O 556
 - B.10.2 Interrupts and Exceptions 556
 - B.11 Advanced Configurations of Nios II Processor 562
 - B.11.1 External Interrupt Controller 562
 - B.11.2 Memory Management Unit 562
 - B.11.3 Floating-Point Hardware 562
 - B.12 Concluding Remarks 563
 - B.13 Solved Problems 563
 - Problems 568
- Appendix C**
THE COLDFIRE PROCESSOR 571
- C.1 Memory Organization 572
 - C.2 Registers 572
 - C.3 Instructions 573
 - C.3.1 Addressing Modes 575
 - C.3.2 Move Instruction 577
 - C.3.3 Arithmetic Instructions 578
 - C.3.4 Branch and Jump Instructions 582
 - C.3.5 Logic Instructions 585
 - C.3.6 Shift Instructions 586
 - C.3.7 Subroutine Linkage Instructions 587
 - C.4 Assembler Directives 593
 - C.5 Example Programs 594
 - C.5.1 Vector Dot Product Program 594
 - C.5.2 String Search Program 595
 - C.6 Mode of Operation and Other Control Features 596
 - C.7 Input/Output 597
 - C.8 Floating-Point Operations 599
 - C.8.1 FMOVE Instruction 599

- C.8.2 Floating-Point Arithmetic Instructions 600
- C.8.3 Comparison and Branch Instructions 601
- C.8.4 Additional Floating-Point Instructions 601
- C.8.5 Example Floating-Point Program 602
- C.9 Concluding Remarks 603
- C.10 Solved Problems 603
 - Problems 608
 - References 609

Appendix D

THE ARM PROCESSOR 611

- D.1 ARM Characteristics 612
 - D.1.1 Unusual Aspects of the ARM Architecture 612
- D.2 Register Structure 613
- D.3 Addressing Modes 614
 - D.3.1 Basic Indexed Addressing Mode 614
 - D.3.2 Relative Addressing Mode 615
 - D.3.3 Index Modes with Writeback 616
 - D.3.4 Offset Determination 616
 - D.3.5 Register, Immediate, and Absolute Addressing Modes 618
 - D.3.6 Addressing Mode Examples 618
- D.4 Instructions 621
 - D.4.1 Load and Store Instructions 621
 - D.4.2 Arithmetic Instructions 622
 - D.4.3 Move Instructions 625
 - D.4.4 Logic and Test Instructions 626
 - D.4.5 Compare Instructions 627
 - D.4.6 Setting Condition Code Flags 628
 - D.4.7 Branch Instructions 628
 - D.4.8 Subroutine Linkage Instructions 631
- D.5 Assembly Language 635
 - D.5.1 Pseudoinstructions 637
- D.6 Example Programs 638
 - D.6.1 Vector Dot Product 639
 - D.6.2 String Search 639
- D.7 Operating Modes and Exceptions 639
 - D.7.1 Banked Registers 641
 - D.7.2 Exception Types 642
 - D.7.3 System Mode 644
 - D.7.4 Handling Exceptions 644
- D.8 Input/Output 646
 - D.8.1 Program-Controlled I/O 646
 - D.8.2 Interrupt-Driven I/O 648

- D.9 Conditional Execution of Instructions 648
- D.10 Coprocessors 650
- D.11 Embedded Applications and the Thumb ISA 651
- D.12 Concluding Remarks 651
- D.13 Solved Problems 652
 - Problems 657
 - References 660

Appendix E

THE INTEL IA-32 ARCHITECTURE 661

- E.1 Memory Organization 662
- E.2 Register Structure 662
- E.3 Addressing Modes 665
- E.4 Instructions 668
 - E.4.1 Machine Instruction Format 670
 - E.4.2 Assembly-Language Notation 670
 - E.4.3 Move Instruction 671
 - E.4.4 Load-Effective-Address Instruction 671
 - E.4.5 Arithmetic Instructions 672
 - E.4.6 Jump and Loop Instructions 674
 - E.4.7 Logic Instructions 677
 - E.4.8 Shift and Rotate Instructions 678
 - E.4.9 Subroutine Linkage Instructions 679
 - E.4.10 Operations on Large Numbers 681
- E.5 Assembler Directives 685
- E.6 Example Programs 686
 - E.6.1 Vector Dot Product Program 686
 - E.6.2 String Search Program 686
- E.7 Interrupts and Exceptions 687
- E.8 Input/Output Examples 689
- E.9 Scalar Floating-Point Operations 690
 - E.9.1 Load and Store Instructions 692
 - E.9.2 Arithmetic Instructions 693
 - E.9.3 Comparison Instructions 694
 - E.9.4 Additional Instructions 694
 - E.9.5 Example Floating-Point Program 694
- E.10 Multimedia Extension (MMX) Operations 695
- E.11 Vector (SIMD) Floating-Point Operations 696
- E.12 Examples of Solved Problems 697
- E.13 Concluding Remarks 702
 - Problems 702
 - References 703