Contents

Preface 5

Chapter 1 Introduction 29

1.1 What Is Inside a Box? 30
1.2 Levels of Abstraction in a Computer System 30
1.3 The Role of the Operating System 33
1.4 What Is Happening Inside the Box? 36
1.4.1 Launching an Application on the Computer 38
1.5 Evolution of Computer Hardware 39
1.6 Evolution of Operating Systems 41
1.7 Roadmap of the Rest of the Book 42
Exercises 42
Bibliographic Notes and Further Reading 43

Chapter 2 Processor Architecture 46

2.1 What Is Involved in Processor Design? 47
2.2 How Do We Design an Instruction Set? 48
2.3 A Common High-Level Language Feature Set 49
2.4 Expressions and Assignment Statements 49
2.4.1 Where To Keep the Operands? 50
2.4.2 How Do We Specify a Memory Address in an Instruction? 54
2.4.3 How Wide Should Each Operand Be? 55
2.4.4 Endianness 58
2.4.5 Packing of Operands and Alignment of Word Operands 60
Chapter 3 Processor Implementation  

3.1 Architecture versus Implementation  104
3.2 What Is Involved in Processor Implementation?  105
3.3 Key Hardware Concepts  106
3.3.1 Circuits  106
Chapter 7  Memory Management Techniques  305

7.1  Functionalities Provided by a Memory Manager  306
7.2  Simple Schemes for Memory Management  308
7.3  Memory Allocation Schemes  313
  7.3.1  Fixed-Size Partitions  314
  7.3.2  Variable-Size Partitions  315
  7.3.3  Compaction  317
7.4  Paged Virtual Memory  318
  7.4.1  Page Table  321
  7.4.2  Hardware for Paging  323
  7.4.3  Page Table Setup  324
  7.4.4  Relative Sizes of Virtual and Physical Memories  324
7.5  Segmented Virtual Memory  325
  7.5.1  Hardware for Segmentation  331
7.6  Paging versus Segmentation  331
  7.6.1  Interpreting the CPU-Generated Address  334
Summary  336
Historical Perspective  337
MULTICS  339
Intel's Memory Architecture  340
Exercises  342
Bibliographic Notes and Further Reading  343

Chapter 8  Details of Page-Based Memory Management  344

8.1  Demand Paging  344
  8.1.1  Hardware for Demand Paging  345
  8.1.2  Page Fault Handler  346
  8.1.3  Data Structures for Demand-Paged Memory Management  346
  8.1.4  Anatomy of a Page Fault  348
Chapter 9 Memory Hierarchy 381

9.1 The Concept of a Cache 382
9.2 Principle of Locality 383
9.3 Basic Terminologies 383
9.4 Multilevel Memory Hierarchy 385
9.5 Cache Organization 388
9.6 Direct-Mapped Cache Organization 388
9.6.1 Cache Lookup 391
9.6.2 Fields of a Cache Entry 393
9.6.3 Hardware for a Direct-Mapped Cache 394
9.7 Repercussion on Pipelined Processor Design 397
Chapter 9 Cache Read/Write Algorithms

9.8 Cache Read/Write Algorithms 398
9.8.1 Read Access to the Cache from the CPU 398
9.8.2 Write Access to the Cache from the CPU 398
9.9 Dealing with Cache Misses in the Processor Pipeline 403
9.9.1 Effect of Memory Stalls Due to Cache Misses on Pipeline Performance 404
9.10 Exploiting Spatial Locality to Improve Cache Performance 405
9.10.1 Performance Implications of Increased Block Size 411
9.11 Flexible Placement 412
9.11.1 Fully Associative Cache 413
9.11.2 Set Associative Cache 415
9.11.3 Extremes of Set Associativity 415
9.12 Instruction and Data Caches 420
9.13 Reducing Miss Penalty 421
9.14 Cache Replacement Policy 422
9.15 Recapping Types of Misses 424
9.16 Integrating TLB and Caches 427
9.17 Cache Controller 428
9.18 Virtually Indexed Physically Tagged Cache 429
9.19 Recap of Cache Design Considerations 432
9.20 Main Memory Design Considerations 433
9.20.1 Simple Main Memory 433
9.20.2 Main Memory and Bus to Match Cache Block Size 434
9.20.3 Interleaved Memory 435
9.21 Elements of Modern Main Memory Systems 436
9.21.1 Page Mode DRAM 441
9.22 Performance Implications of Memory Hierarchy 443
Summary 444

Chapter 10 Input/Output and Stable Storage 451

10.1 Communication Between the CPU and the I/O Devices 451
10.1.1 Device Controller 452
10.1.2 Memory Mapped I/O 453
Chapter 11 File System 497

11.1 Attributes 497
11.2 Design Choices in Implementing a File System on a Disk Subsystem 504
11.2.1 Contiguous Allocation 505
11.2.2 Contiguous Allocation with Overflow Area 508
11.2.3 Linked Allocation 508
11.2.4 File Allocation Table (FAT) 509
11.2.5 Indexed Allocation 511
11.2.6 Multilevel Indexed Allocation 513
Chapter 12 Multithreaded Programming and Multiprocessors 549

12.1 Why Multithreading? 550
12.2 Programming Support for Threads 551
12.2.1 Thread Creation and Termination 551
12.2.2 Communication Among Threads 554
12.2.3 Read-Write Conflict, Race Condition, and Nondeterminism 556
12.2.4 Synchronization Among Threads 561
12.2.5 Internal Representation of Data Types Provided by the Threads Library 568
12.2.6 Simple Programming Examples 569
12.2.7 Deadlocks and Livelocks 574
12.2.8 Condition Variables 576
12.2.9 A Complete Solution for the Video Processing Example 581
12.2.10 Discussion of the Solution 582
12.2.11 Rechecking the Predicate 584
Chapter 12 \textit{Fundamentals of Networking and Network Protocols} 648

12.3 Summary of Thread Function Calls and Threaded Programming Concepts 587
12.4 Points to Remember in Programming with Threads 589
12.5 Using Threads as Software Structuring Abstraction 589
12.6 POSIX pthreads Library Calls Summary 590
12.7 OS Support for Threads 593
12.7.1 User Level Threads 595
12.7.2 Kernel-Level Threads 598
12.7.3 Solaris Threads: An Example of Kernel-Level Threads 600
12.7.4 Threads and Libraries 601
12.8 Hardware Support for Multithreading in a Uniprocessor 602
12.8.1 Thread Creation, Termination, and Communication Among Threads 602
12.8.2 Inter-Thread Synchronization 603
12.8.3 An Atomic Test-and-Set Instruction 603
12.8.4 Lock Algorithm with Test-and-Set Instruction 605
12.9 Multiprocessors 606
12.9.1 Page Tables 607
12.9.2 Memory Hierarchy 608
12.9.3 Ensuring Atomicity 610
12.10 Advanced Topics 611
12.10.1 OS Topics 611
12.10.2 Architecture Topics 624
12.10.3 The Road Ahead: Multicore and Many-Core Architectures 638

Summary 640
Historical Perspective 640
Exercises 642
Bibliographic Notes and Further Reading 645

Chapter 13 \textit{Fundamentals of Networking and Network Protocols} 648

13.1 Preliminaries 648
13.2 Basic Terminologies 649
13.3 Networking Software 654
Chapter 14 Epilogue: A Look Back at the Journey 748

14.1 Processor Design 748
14.2 Process 748
14.3 Virtual Memory System and Memory Management 749
14.4 Memory Hierarchy 750
14.5 Parallel System 750
14.6 Input/Output Systems 750
14.7 Persistent Storage 751
14.8 Network 751

Concluding Remarks 751

Appendix: Network Programming with UNIX Sockets 752

Bibliography 764

Index 770