ADVANCED COMPUTER ARCHITECTURE

Kai Hwang

PARALLELISM

SCALABILITY

PROGRAMMABILITY

For sale in
India,
Nepal, Bangladesh,
Sri Lanka and Bhutan
Only

TATA McGRAW-HILL
EDITION

Scanned by TapScanner

Contents

Foreword xvii

Preface xix

PART I THEORY OF PARALLELISM 1

Chapter 1 Parallel Computer Models 3

1.1 The State of Computing 3
1.1.1 Computer Development Milestones 3
1.1.2 Elements of Modern Computers 6
1.1.3 Evolution of Computer Architecture 9
1.1.4 System Attributes to Performance 14

1.2 Multiprocessors and Multicomputers 19
1.2.1 Shared-Memory Multiprocessors 19
1.2.2 Distributed-Memory Multicomputers 24
1.2.3 A Taxonomy of MIMD Computers 27

1.3 Multivector and SIMD Computers 27
1.3.1 Vector Supercomputers 27
1.3.2 SIMD Supercomputers 30

1.4 PRAM and VLSI Models 32
1.4.1 Parallel Random-Access Machines 33
1.4.2 VLSI Complexity Model 38

1.5 Architectural Development Tracks 41
1.5.1 Multiple-Processor Tracks 41
1.5.2 Multivector and SIMD Tracks 43
1.5.3 Multithreaded and Dataflow Tracks 44

1.6 Bibliographic Notes and Exercises 45

Scanned by TapScanner

Chapter 2 Program and Network Properties 51
2.1 Conditions of Parallelism 51
2.1.1 Data and Resource Dependences 51
2.1.2 Hardware and Software Parallelism 57
2.1.3 The Role of Compilers 60
2.2 Program Partitioning and Scheduling 61
2.2.1 Grain Sizes and Latency 61
2.2.2 Grain Packing and Scheduling 64
2.2.3 Static Multiprocessor Scheduling 67
2.3 Program Flow Mechanisms 70
2.3.1 Control Flow Versus Data Flow 71
2.3.2 Demand-Driven Mechanisms 74
2.3.3 Comparison of Flow Mechanisms 75
2.4 System Interconnect Architectures 76
2.4.1 Network Properties and Routing 77
2.4.2 Static Connection Networks 80
2.4.3 Dynamic Connection Networks 89
2.5 Bibliographic Notes and Exercises 96

Chapter 3 Principles of Scalable Performance 105
3.1 Performance Metrics and Measures 105
3.1.1 Parallelism Profile in Programs 105
3.1.2 Harmonic Mean Performance 108
3.1.3 Efficiency, Utilization, and Quality 112
3.1.4 Standard Performance Measures 115
3.2 Parallel Processing Applications 118
3.2.1 Massive Parallelism for Grand Challenges 118
3.2.2 Application Models of Parallel Computers 122
3.2.3 Scalability of Parallel Algorithms 125
3.3 Speedup Performance Laws 129
3.3.1 Amdahl's Law for a Fixed Workload 129
3.3.2 Gustafson's Law for Scaled Problems 131
3.3.3 Memory-Bounded Speedup Model 134
3.4 Scalability Analysis and Approaches 138
3.4.1 Scalability Metrics and Goals 138
3.4.2 Evolution of Scalable Computers 143
3.4.3 Research Issues and Solutions 147
3.5 Bibliographic Notes and Exercises 149

PART II HARDWARE TECHNOLOGIES 155

Scanned by TapScanner

Chapter 4 Processors and Memory Hierarchy **157**
4.1 Advanced Processor Technology 157
4.1.1 Design Space of Processors 157
4.1.2 Instruction-Set Architectures 162
4.1.3 CISC Scalar Processors 165
4.1.4 RISC Scalar Processors 169
4.2 Superscalar and Vector Processors 177
4.2.1 Superscalar Processors 178
4.2.2 The VLIW Architecture 182
4.2.3 Vector and Symbolic Processors 184
4.3 Memory Hierarchy Technology 188
4.3.1 Hierarchical Memory Technology 188
4.3.2 Inclusion, Coherence, and Locality 190
4.3.3 Memory Capacity Planning 194
4.4 Virtual Memory Technology 196
4.4.1 Virtual Memory Models 196
4.4.2 TLB, Paging, and Segmentation 198
4.4.3 Memory Replacement Policies 205
4.5 Bibliographic Notes and Exercises 208

Chapter 5 Bus, Cache, and Shared Memory **213**
5.1 Backplane Bus Systems 213
5.1.1 Backplane Bus Specification 213
5.1.2 Addressing and Timing Protocols 216
5.1.3 Arbitration, Transaction, and Interrupt 218
5.1.4 The IEEE Futurebus+ Standards 221
5.2 Cache Memory Organizations 224
5.2.1 Cache Addressing Models 225
5.2.2 Direct Mapping and Associative Caches 228
5.2.3 Set-Associative and Sector Caches 232
5.2.4 Cache Performance Issues 236
5.3 Shared-Memory Organizations 238
5.3.1 Interleaved Memory Organization 239
5.3.2 Bandwidth and Fault Tolerance 242
5.3.3 Memory Allocation Schemes 244
5.4 Sequential and Weak Consistency Models 248
5.4.1 Atomicity and Event Ordering 248
5.4.2 Sequential Consistency Model 252
5.4.3 Weak Consistency Models 253
5.5 Bibliographic Notes and Exercises 256

Chapter 6 Pipelining and Superscalar Techniques **265**

Scanned by TapScanner

6.1 Linear Pipeline Processors 265
6.1.1 Asynchronous and Synchronous Models 265
6.1.2 Clocking and Timing Control 267
6.1.3 Speedup, Efficiency, and Throughput 268
6.2 Nonlinear Pipeline Processors 270
6.2.1 Reservation and Latency Analysis 270
6.2.2 Collision-Free Scheduling 274
6.2.3 Pipeline Schedule Optimization 276
6.3 Instruction Pipeline Design 280
6.3.1 Instruction Execution Phases 280
6.3.2 Mechanisms for Instruction Pipelining 283
6.3.3 Dynamic Instruction Scheduling 288
6.3.4 Branch Handling Techniques 291
6.4 Arithmetic Pipeline Design 297
6.4.1 Computer Arithmetic Principles 297
6.4.2 Static Arithmetic Pipelines 299
6.4.3 Multifunctional Arithmetic Pipelines 307
6.5 Superscalar and Superpipeline Design 308
6.5.1 Superscalar Pipeline Design 310
6.5.2 Superpipelined Design 316
6.5.3 Supersymmetry and Design Tradeoffs 320
6.6 Bibliographic Notes and Exercises 322

PART III PARALLEL AND SCALABLE ARCHITECTURES 329

Chapter 7 Multiprocessors and Multicomputers 331
7.1 Multiprocessor System Interconnects 331
7.1.1 Hierarchical Bus Systems 333
7.1.2 Crossbar Switch and Multiport Memory 336
7.1.3 Multistage and Combining Networks 341
7.2 Cache Coherence and Synchronization Mechanisms 348
7.2.1 The Cache Coherence Problem 348
7.2.2 Snoopy Bus Protocols 351
7.2.3 Directory-Based Protocols 358
7.2.4 Hardware Synchronization Mechanisms 364
7.3 Three Generations of Multicomputers 368
7.3.1 Design Choices in the Past 368
7.3.2 Present and Future Development 370
7.3.3 The Intel Paragon System 372
7.4 Message-Passing Mechanisms 375
7.4.1 Message-Routing Schemes 375

Scanned by TapScanner

7.4.2 Deadlock and Virtual Channels 379
7.4.3 Flow Control Strategies 383
7.4.4 Multicast Routing Algorithms 387
7.5 Bibliographic Notes and Exercises 393

Chapter 8 Multivector and SIMD Computers 403
8.1 Vector Processing Principles 403
8.1.1 Vector Instruction Types 403
8.1.2 Vector-Access Memory Schemes 408
8.1.3 Past and Present Supercomputers 410
8.2 Multivector Multiprocessors 415
8.2.1 Performance-Directed Design Rules 415
8.2.2 Cray Y-MP, C-90, and MPP 419
8.2.3 Fujitsu VP2000 and VPP500 425
8.2.4 Mainframes and Minisupercomputers 429
8.3 Compound Vector Processing 435
8.3.1 Compound Vector Operations 436
8.3.2 Vector Loops and Chaining 437
8.3.3 Multipipeline Networking 442
8.4 SIMD Computer Organizations 447
8.4.1 Implementation Models 447
8.4.2 The CM-2 Architecture 449
8.4.3 The MasPar MP-1 Architecture 453
8.5 The Connection Machine CM-5 457
8.5.1 A Synchronized MIMD Machine 457
8.5.2 The CM-5 Network Architecture 460
8.5.3 Control Processors and Processing Nodes 462
8.5.4 Interprocessor Communications 465
8.6 Bibliographic Notes and Exercises 468

Chapter 9 Scalable, Multithreaded, and Dataflow Architectures 475
9.1 Latency-Hiding Techniques 475
9.1.1 Shared Virtual Memory 476
9.1.2 Prefetching Techniques 480
9.1.3 Distributed Coherent Caches 482
9.1.4 Scalable Coherence Interface 483
9.1.5 Relaxed Memory Consistency 486
9.2 Principles of Multithreading 490
9.2.1 Multithreading Issues and Solutions 490
9.2.2 Multiple-Context Processors 495
9.2.3 Multidimensional Architectures 499
9.3 Fine-Grain Multicomputers 504

Scanned by TapScanner

9.3.1 Fine-Grain Parallelism 505
9.3.2 The MIT J-Machine 506
9.3.3 The Caltech Mosaic C 514
9.4 Scalable and Multithreaded Architectures 516
9.4.1 The Stanford Dash Multiprocessor 516
9.4.2 The Kendall Square Research KSR-1 521
9.4.3 The Tera Multiprocessor System 524
9.5 Dataflow and Hybrid Architectures 531
9.5.1 The Evolution of Dataflow Computers 531
9.5.2 The ETL/EM-4 in Japan 534
9.5.3 The MIT/Motorola *T Prototype 536
9.6 Bibliographic Notes and Exercises 539

PART IV SOFTWARE FOR PARALLEL PROGRAMMING 545

Chapter 10 Parallel Models, Languages, and Compilers 547
10.1 Parallel Programming Models 547
10.1.1 Shared-Variable Model 547
10.1.2 Message-Passing Model 551
10.1.3 Data-Parallel Model 554
10.1.4 Object-Oriented Model 556
10.1.5 Functional and Logic Models 559
10.2 Parallel Languages and Compilers 560
10.2.1 Language Features for Parallelism 560
10.2.2 Parallel Language Constructs 562
10.2.3 Optimizing Compilers for Parallelism 564
10.3 Dependence Analysis of Data Arrays 567
10.3.1 Iteration Space and Dependence Analysis 567
10.3.2 Subscript Separability and Partitioning 570
10.3.3 Categorized Dependence Tests 573
10.4 Code Optimization and Scheduling 578
10.4.1 Scalar Optimization with Basic Blocks 578
10.4.2 Local and Global Optimizations 581
10.4.3 Vectorization and Parallelization Methods 585
10.4.4 Code Generation and Scheduling 592
10.4.5 Trace Scheduling Compilation 596
10.5 Loop Parallelization and Pipelining 599
10.5.1 Loop Transformation Theory 599
10.5.2 Parallelization and Wavefronting 602
10.5.3 Tiling and Localization 605
10.5.4 Software Pipelining 610

Scanned by TapScanner

10.6 Bibliographic Notes and Exercises 612

Chapter 11 Parallel Program Development and Environments 617

11.1 Parallel Programming Environments 617
11.1.1 Software Tools and Environments 617
11.1.2 Y-MP, Paragon, and CM-5 Environments 621
11.1.3 Visualization and Performance Tuning 623

11.2 Synchronization and Multiprocessing Modes 625
11.2.1 Principles of Synchronization 625
11.2.2 Multiprocessor Execution Modes 628
11.2.3 Multitasking on Cray Multiprocessors 629

11.3 Shared-Variable Program Structures 634
11.3.1 Locks for Protected Access 634
11.3.2 Semaphores and Applications 637
11.3.3 Monitors and Applications 640

11.4 Message-Passing Program Development 644
11.4.1 Distributing the Computation 644
11.4.2 Synchronous Message Passing 645
11.4.3 Asynchronous Message Passing 647

11.5 Mapping Programs onto Multicomputers 648
11.5.1 Domain Decomposition Techniques 648
11.5.2 Control Decomposition Techniques 652
11.5.3 Heterogeneous Processing 656

11.6 Bibliographic Notes and Exercises 661

Chapter 12 UNIX, Mach, and OSF/1 for Parallel Computers 667

12.1 Multiprocessor UNIX Design Goals 667
12.1.1 Conventional UNIX Limitations 668
12.1.2 Compatibility and Portability 670
12.1.3 Address Space and Load Balancing 671
12.1.4 Parallel I/O and Network Services 671

12.2 Master-Slave and Multithreaded UNIX 672
12.2.1 Master-Slave Kernels 672
12.2.2 Floating-Executive Kernels 674
12.2.3 Multithreaded UNIX Kernel 678

12.3 Multicomputer UNIX Extensions 683
12.3.1 Message-Passing OS Models 683
12.3.2 Cosmic Environment and Reactive Kernel 683
12.3.3 Intel NX/2 Kernel and Extensions 685

12.4 Mach/OS Kernel Architecture 686
12.4.1 Mach/OS Kernel Functions 687
12.4.2 Multithreaded Multitasking 688

Scanned by TapScanner

12.4.3 Message-Based Communications 694
12.4.4 Virtual Memory Management 697
12.5 OSF/1 Architecture and Applications 701
12.5.1 The OSF/1 Architecture 702
12.5.2 The OSF/1 Programming Environment 707
12.5.3 Improving Performance with Threads 709
12.6 Bibliographic Notes and Exercises 712

Bibliography 717

Index 739

Answers to Selected Problems 765

Scanned by TapScanner