

Contents

Foreword xv

Preface xix

Chapter 1 Network Processor Overview 1

What Is a Network Processor? 2

Why Do You Need a Network Processor? 3

 Data Rates That Keep Going Up 3

 Nature of Network Traffic and Impact to Memory Latency 4

 Protocols That Keep Evolving 5

 Reuse Across Product Lines 6

Design Techniques Used by Network Processors 6

 Multiple Processing Units in Parallel 7

 Multiple Processing Units in a Pipeline 7

 Multi-threading Support 7

 Coproductors 8

 Special Instructions for Packet Processing 8

 Efficient Communication Mechanisms and System Services 8

 Memory Hierarchy 9

 Sufficient Bandwidth for Moving Packets 9

 Fast-path- Slow-path Decomposition 9

Where Can You Use a Network Processor?	10
Enterprise Networks	12
SOHO Networks	13
Access Networks	13
Wide Area Networks	14
MAN Networks	15
Conclusion	16

Chapter 2 Intel® IXP2XXX Network Processor Hardware Overview 19

IXP2XXX Product Line Description	20
Hardware Architecture	21
Microengine Features	24
Multi-threading	25
Instruction Set	25
Signals	27
Registers	28
Local Memory	30
Control Store	32
Content Addressable Memory (CAM)	33
Local CSRs	35
Memory Hierarchy	36
Interfacing with Media Devices	38
Packet Segmentation and Reassembly	38
Support for Standard PHY Protocols	39
Interfacing to a Fabric	42
Communication and Synchronization	45
Integrated Intel XScale® Core	52
What's New on IXP23XX Network Processors	53
Conclusion	57

Chapter 3 Intel® IXP2XXX Network Processor Software Overview 59

Software in a Networking Device	60
Software Tools for the IXP2XXX Product Line	66
Architecture Tool	67
Intel Developer Workbench	67
Intel XScale® Core Development Tools	69

The Intel® IXA Portability Framework	69
Goals and Benefits	70
Components of the Intel® IXA Portability Framework	72
Conclusion	77

Chapter 4 Packet Buffer Architecture 79

Packet Format	80
Packet Buffer	81
Packet Descriptor or Packet Meta-data	82
Packet Handle	86
Allocation and Freeing of Packet Buffers	88
Freeing a Chain of Buffers	88
Optimizing Memory Bandwidth by Recycling Packet Buffers	90
Using Bitmaps: An Alternate Approach to Buffer Allocation	91
Support for Multiple Freelists	93
Utilizing DRAM Bandwidth Efficiently	94
Bank Interleaving on the Intel® IXP2400 Network Processor	94
Channel and Bank Interleaving on the IXP2800 Network Processor	95
Chaining Buffers to Support Large Packets	96
Supporting Multi-cast	100
APIs from the Microblock Infrastructure Library	101
Summary of Features and Optimizations	102
Conclusion	104

Chapter 5 Packet Movement Model 105

Fast-path Design for a Typical Application	106
Buffering Packets in DRAM Versus Processing from RBUFs	109
Run-to-completion Model	113
Separation of Driver Code from Packet-processing Code	115
A Closer Look at a Packet-processing Microengine	117
Packet Header Caching	118
Packet Descriptor Caching	118
Separation of Application-specific and Reusable Code	119
Controlling the Data Flow	120
Data Flow Description	120
Conclusion	123

Chapter 6 Critical Sections and Packet Ordering 125

- Locks and Semaphores 126
 - The Single Flow Problem 126
 - Ordering Accesses to Data Structures 127
- Design Requirements for Critical Sections 127
- Synchronizing Accesses on a Single Microengine 128
 - Folding 129
 - Deli Ticket Server 133
 - Ordering Accesses with the Deli Ticket Server Algorithm 136
- Synchronizing Accesses Across Microengines 137
 - Using CAP Versus Next-neighbor Signal 141
 - Impact of Critical Sections on Performance 142
 - Does the Cache Actually Help? 144
 - Locks in Memory—A Second Look 144
 - Locking Using `test_and_set` and `clr` Instructions 145
 - Deli Ticket Server Using `test_and_incr` Instructions 146
- Packet Ordering 147
 - Ordered Thread Execution 147
 - Unordered Thread Execution 152
 - Using Local Memory to Cache AISR Entries 154
 - Using Multiple AISR Arrays to Avoid Head-of-Line Blocking 155
- Conclusion 156

Chapter 7 Writing a Microblock 157

- Granularity of a Microblock 158
- Microblock Basics 159
 - Directory Structure 161
- Controlling the Data Flow 161
 - Naming Microblock Outputs 163
 - Binding Microblock Outputs 163
- Modifying the Packet Descriptor 169
- Modifying the Packet Header 172
 - XBUF Library 176
 - Single Globally Visible Cache 176
 - Distributed Cache of Headers 179
 - Distributed Aligned Cache of Headers in Local Memory 182
 - Application Example for Header Caching 182
- Sending Packets to the Intel XScale® Core 189

Receiving Packets from the Intel XScale® Core	192
Sharing Data Structures with the Intel XScale® Core	194
Design Considerations for Intel® IXP23XXX Network Processors	195
Assigning Multiple Dispatch Loops to a Microengine	195
Sharing the CAM	200
Relocating Data Structures	203
Summary	205

Chapter 8 Performance Analysis 207

Goals	208
Performance Metrics	209
Application Performance Metrics	209
Intel® IXP2XXX Network Processor Performance Metrics	211
Capacity Metrics	223
Performance Analysis Steps	226
Application Mapping	226
Characterizing Critical Paths	229
Choosing Analysis Scenarios	236
Back-of-the-Envelope Performance Analysis	237
Memory Bandwidth Utilization	237
Internal Bandwidth Utilization	241
Microengine Utilization	245
Conclusion	251

Chapter 9 Using the Intel® IXP2XXX Product Line Architecture Tool 253

The IXP2XXX Product Line Architecture Tool	253
Creating an AT Project	255
Limitations of the Back-of-the-Envelope Analysis Technique	265
AT Analysis Technique	271
Tracking Performance During Development	272
Conclusion	275

Chapter 10 Performance Tuning 277

Microengine Compute Bottlenecks	278
Reducing the Instruction Cycle Count	278
Increasing the Microengine Cycle Budget	281
Memory Bandwidth Bottlenecks	282
Using Optimal Alignment for Data Structures	283
Using DRAM Bank Interleaving Effectively	285

- Splitting Data Structures Across SRAM Channels 286
- Trading Latency for Bandwidth 286
- Trading Memory Bandwidth for Internal Bandwidth 287
- Internal Bus Bandwidth Bottlenecks 288
 - Balancing Utilization Across Microengine Clusters 288
 - Optimizing for Scratchpad Bandwidth Utilization 289
 - Optimizing Command Bus Utilization 290
 - Optimizing Hash Unit Utilization 291
- I/O Latency Bottlenecks 291
 - I/O Overlapping 291
 - Code Motion 292
 - Trading Bandwidth for Latency 293
 - Using Interleaved Execution 294
- Critical Section Bottlenecks 296
 - Moving I/O Latency Outside the Critical Section 297
 - Using Folding 299
- Performance Tuning Checklist 302
- Conclusion 303

Chapter 11 Receiving and Transmitting Packets 305

- Media Switch Fabric Receive Hardware 306
 - RBUF Elements 306
 - Receive-thread Freelists 307
 - MSF Receive Control Logic 308
- Basics of Packet Reassembly 309
- Challenges in Packet Reassembly 314
 - Large Number of Reassembly Contexts 314
 - Alignment of Packet Headers 317
 - CRC and Checksum Computation 319
 - Performance Constraints 320
 - Computing Cell Count 321
- MSF Transmit Hardware 323
 - TBUF Elements 323
 - Flow Control 326
- Basics of Packet Segmentation 331
 - Reading the Transmit Request 331
 - Reading the Packet Descriptor 331
 - Queuing the Packet Handle in Local Memory 331
 - Allocating a TBUF Element 332

Checking for Flow Control	332
Picking and Updating a TXC	333
Writing into a TBUF Element	333
Validating a TBUF Element	333
Freeing the Packet Buffer	334
Optimizing Packet Transmit	335
Tuning the Critical Paths	335
Scaling to Higher Data Rates—Parallel or Pipeline?	336
Latency Hiding	339
Handling a Large Number of Transmit Contexts	343
Using Pipelining to Improve Flow-control Handling	344
Adding Layer-2 Headers	345
Handling Bus Width Restrictions	346
Conclusion	348

Chapter 12 Protocol Processing 349

Packet Classification	350
Exact-match Classification	350
Longest-prefix Match Classification	358
Range-match Classification	369
Connection Setup and Teardown	374
Distributed Versus Centralized Processing	375
Connection Rate Limiting	377
Maintaining Packet Order	377
Connection-state Allocation/Deletion	378
Exact-match Classifier Locking	379
Connection Aging	380
Statistics	382
Design Considerations	382
Maintaining 32-bit Statistics	385
Maintaining 64-bit Statistics	386
Summary of Statistics	388
Conclusion	388

Chapter 13 Traffic Management 389

Elements of Traffic Management	390
Packet Classifiers	390
Meters and Markers	390
Buffer and Queue Managers	391
Schedulers and Shapers	391
Useful Hardware Features	392

- Timestamp CSR 392
- Multiply Instruction 392
- Pseudo-random Number Generator 392
- FFS 392
- Q-Array 393
- CAM and Local Memory 393
- Hash Unit and CRC instruction 393
- Metering and Marking 393
 - Two-rate Three-color Meter 395
 - Pseudo-code for trTCM 400
 - Marking 402
- Buffer Management 403
 - Tail Drop 403
 - Random Early Detection (RED) 404
 - Implementation of WRED on Intel® IXP2XXX Network Processors 407
 - Optimizing WRED for High-speed Links 411
- Scheduling and Shaping 414
 - Functions of a Scheduler 414
 - Challenges with Implementing High-speed Schedulers 415
 - Design Techniques and Trade-offs 416
 - Design of a WRR Fabric Scheduler 425
 - Design of an Enqueue-time DRR Scheduler 428
 - Design of a Hierarchical Scheduler 432
 - Implementing Shaping 436
- Queue Manager 441
- Conclusion 445

Chapter 14 Core/Metro Router 447

- System Architecture 448
- Application Features and Requirements 449
 - Supported Media Interfaces 449
 - Fabric Support 449
 - Ingress Packet Classification and Filtering 450
 - Ingress Header Verification and Packet Forwarding 451
 - Ingress Policing 452
 - Ingress Scheduling and Congestion Avoidance 452
 - Egress Packet Processing 452
 - Egress Traffic Management 452

Performance Requirements	453
Application Design	453
Data Flow	453
Packet-processing Stages	460
Microblocks	461
Header Caching	467
Optimizing Memory Bandwidth Usage	467
Microengine Mapping	468
Performance Analysis	470
Critical Paths	471
Microengine Cycle and Memory Bandwidth Budget	471
Headroom Analysis	473
Conclusion	482

Chapter 15 IP DSLAM 483

System Architecture	484
Application Requirements	485
Functional Requirements	485
Performance Requirements	486
Application Design	487
Data Flow	487
Packet-processing Stages	491
Microblocks	493
Header Caching	497
Packet Queuing and Buffering	498
Microengine Mapping	500
Performance Analysis	501
Critical Paths	501
Budget Analysis	502
Headroom Analysis	504
Conclusion	510

References 511

Index 513

