

# Contents

---

## **Preface xv**

## **Chapter 1 Introduction 1**

- CISC, RISC, VLIW, and EPIC Architectures 1
  - CISC and RISC 1
  - Parallel Execution 2
  - VLIW 3
  - EPIC 4
- 64-Bit Computing 5
- Instruction-Level Parallelism and the Itanium® Architecture 7
  - Instruction-Level Parallelism 7
  - Itanium Architecture Solutions to Limitations on ILP 8
- Scientific Computing 13
- Organization of this Book 15

## **Chapter 2 Application Registers, Predication, Branching, and Register Rotation 19**

- Instruction Groups and Instruction Bundles 19
- Application Registers 22
  - General Registers 22
  - Floating-Point Registers 23
  - Predicate Registers 24
  - Branch Registers 24

- Instruction Pointer 24
- Current Frame Marker 24
- User Mask 24
- Application Registers 25
- Advanced Load Address Table 26
- Process Identifiers 26
- Performance Monitor Data Registers 26
- Register Stack 26
- Predication 28
  - Setting Predicate Registers 29
  - Exploiting Predication 33
- Branching 34
  - Branch Instructions 34
  - Branch Prediction Hints 38
  - Branch Prediction Instructions 40
- Register Rotation and Modulo-Scheduled Loop Support 42
  - Counted Loop-Type Branch Instructions 45
  - While Loop-Type Branch Instructions 50

### **Chapter 3 Floating-Point Architecture 55**

- Available Floating-Point Formats 55
- Register and Memory Encodings 58
- Status Fields and Exceptions 65
- The Floating-Point Multiply-Add 69
  - Exact Arithmetic 72
  - Accurate Remainders 72
  - Accurate Range Reduction 73
- Comparison and Classification 74
- Division and Square Root 76
- Additional Features 77
  - Transferring Between Floating-Point and Integer Registers 78
  - Floating-Point Merging 79
  - Floating-Point/Integer Conversion 79
  - Integer Multiplication and Division 80
  - Maximum and Minimum 81

**Chapter 4 Memory and Speculation 83**

- Memory 84
  - Memory Access Instructions 85
  - Memory Hierarchy Allocation Control 92
- Speculation 96
  - Control Speculation 97
  - Data Speculation 104

**Chapter 5 Introduction to Itanium Architecture  
Assembly Language Programming 111**

- Itanium Processor Assembly Language 111
  - Identifiers 112
  - Names 112
  - Constants 117
  - Expressions 118
  - Statements 119
  - Intel Itanium Processor Assembler Command Line Options 131
- Software Conventions 131
  - Register Usage 132
  - Procedure Linkage 138

**Chapter 6 Itanium Processor Microarchitecture 149**

- The Itanium Processor Core Pipeline 149
- Functional Units, Issue Ports, and Issue Rules 155
  - Functional Units 155
  - Issue Ports 156
  - Instruction and Bundle Types 157
  - Split Issue 159
  - Dispersal and Bundle Rotation 161
- Instruction Dispersal Examples 163
- Latencies and Bypasses 167
- Measuring Latencies and Execution Times 169
- Memory Hierarchy 172
  - Processor Specific Exceptions 174
  - Data Prefetch 176
- Branch Instructions and Branch Prediction 187

## **Chapter 7 Floating-Point Exception Handling in the Itanium Processor Family 189**

- Handling Floating-Point Exceptions in the Itanium Architecture 191
- Handling IEEE and Denormal Floating-Point Exceptions 192
  - The IEEE Floating-Point Exception Filter 194
  - Example of Invalid Operation Fault 195
- Determining the Exponent of the Result for Enabled Overflow or Underflow 196
  - Overflow 197
  - Underflow 198
- Example of User Handler for Itanium and Itanium 2 Processor Floating-Point Exceptions 200
- Floating-Point Software Assistance 207
  - Architecturally Mandated FPSWA Faults 207
  - Processor-Specific FPSWA Faults 208
  - Processor-Specific FPSWA Traps 209
- Handling FPSWA Faults and Traps 209
  - Example of Architecturally Mandated FPSWA Fault Followed by Underflow Trap 212
- Floating-Point State Used in Exception Handling in the Itanium Architecture 213
- Floating-Point Exceptions Raised by Itanium and Itanium 2 Processor Instructions 215
  - Floating-Point Exception Priority 216
- Safe Instruction Recognition 221

## **Chapter 8 Division, Remainder, and Square Root 223**

- Ulps and Relative Error 223
- Floating-Point Division 225
  - Refining Reciprocals 226
  - Refining Quotients 227
  - IEEE-754 Correctness 228
  - Native Precision 232
  - Implementation 235
  - Special Division Algorithms 237
- Floating-Point Square Root 238
  - Exclusion Zones 242

Critical Values	244
Implementation	247
Floating-Point Remainder	248
Integer Division and Remainder	249
Integer Division	250
Integer Remainder	254
Conversions Between Floating-Point and Integer Formats	255
Implementation of the Integer Division and Remainder Operations	258

## **Chapter 9 Basic Examples 261**

Polynomial Evaluation	261
Horner's Rule	262
Parallelism	263
Complex Arithmetic	269
Implementation in the Itanium Architecture	270
Quad Precision Arithmetic	277
Quad Precision Addition and Subtraction	278
Quad Precision Multiplication and Division	279
Special Instructions	288
The <code>frcpa</code> Trick	288
Maximum and Minimum Values	290
Load Pair	291
The <code>setf</code> and <code>getf</code> Instructions	292
Software Pipelining	295

## **Chapter 10 Run-Time Support and Floating-Point Compilation Model 303**

Run-Time Library	303
Functions Provided	304
Performance Characteristics	307
Accuracy	311
Implementation Details	313
Double-Precision Algorithms	313
Double-Extended Precision Algorithms	316
Floating-Point Compilation Model	317
Use of the Fused Multiply-Add Instruction	318

- Expression Evaluation 319
- Precision of Intermediates 319
- Casting and Storing Intermediates 320
- Order of Evaluation 320
- Floating-Point Formats 320

## **Chapter 11 Selected Kernel Computations 323**

- Interval Arithmetic 323
  - Rectangular Representation of Interval 325
  - Circular Representation of Interval 328
  - Computational Method for Rounding Error Bound 333
- High-Precision Arithmetic 335
  - Representation of High-Precision Numbers 336
  - Radix-2 Representation by Integer Array 336
  - High-Radix Representation by Floating-Point Array 337
  - High-Precision Multiplication 337
- Fast Fourier Transform 340
  - Twiddle Factors Generation 341
  - Butterfly Kernels 343
- Numerical Linear Algebra and BLAS 346
  - Evolution of BLAS 346
  - An Illustration of the Use of Level-3 BLAS 350
  - An Implementation of DGEMM on Itanium Architecture 352
  - Performance of BLAS in MKL on the Itanium Processor 357
  - Performance of Linear Algebra Kernels on Itanium 2 Processor 360
- Vector Math Library 361
  - Mathematical Algorithm and Numerical Techniques 363
  - Serial Code for the Main Computation 364
  - Software Pipeline Code for the Main Computation 365
- Cryptography 368
  - High-Precision Integer Arithmetic 369
  - Fast Multiplication Algorithms 373
  - Implementing Multiplication 375
  - 512 × 512-Bit Multiplication 376
  - Summary of Multiplication Efficiency 386

Integer Squaring 387  
AES Implementation 388

**References 393**

**Index 399**