

Contents

Acknowledgments	xiii
Foreword	xvii
Preface	xxiii

SECTION I KNIGHTS LANDING

CHAPTER 1 Introduction	3
Introduction to Many-Core Programming	4
Trend: More Parallelism.....	4
Why Intel® Xeon Phi™ Processors Are Needed.....	6
Processors Versus Coprocessor.....	8
Measuring Readiness for Highly Parallel Execution.....	9
What About GPUs?	10
Enjoy the Lack of Porting Needed but Still Tune!.....	10
Transformation for Performance	11
Hyper-Threading Versus Multithreading	11
Programming Models	12
Why We Could Skip To Section II Now.....	12
For More Information.....	13
CHAPTER 2 Knights Landing Overview	15
Overview.....	15
Instruction Set.....	16
Architecture Overview	17
Motivation: Our Vision and Purpose	21
Summary	23
For More Information.....	24
CHAPTER 3 Programming MCDRAM and Cluster Modes	25
Programming for Cluster Modes.....	26
Programming for Memory Modes.....	27
Query Memory Mode and MCDRAM Available.....	45
SNC Performance Implications of Allocation and Threading	45
How to Not Hard Code the NUMA Node Numbers	47
Approaches to Determining What to Put in MCDRAM	48
Why Rebooting Is Required to Change Modes	56

	BIOS	56
	Summary	60
	For More Information.....	60
CHAPTER 4	Knights Landing Architecture	63
	Tile Architecture.....	63
	Cluster Modes.....	71
	Memory Interleaving	76
	Memory Modes.....	78
	Interactions of Cluster and Memory Modes	82
	Summary	84
	For More Information.....	84
CHAPTER 5	Intel Omni-Path Fabric	85
	Overview.....	85
	Performance and Scalability.....	88
	Transport Layer APIs	90
	Quality of Service.....	92
	Virtual Fabrics	95
	Unicast Address Resolution	101
	Multicast Address Resolution	103
	Summary	104
	For More Information.....	105
CHAPTER 6	μarch Optimization Advice	107
	Best Performance From 1, 2, or 4 Threads Per Core, Rarely 3	107
	Memory Subsystem	109
	μ arch Nuances (Tile).....	110
	Direct Mapped MCDRAM Cache.....	119
	Advice: Use AVX-512	120
	Summary	144
	For More Information.....	145
 SECTION II PARALLEL PROGRAMMING		
CHAPTER 7	Programming Overview for Knights Landing	149
	To Refactor, or Not to Refactor, That Is the Question.....	150
	Evolutionary Optimization of Applications.....	151
	Revolutionary Optimization of Applications.....	152

	Know When to Hold'em and When to Fold'em	153
	For More Information.....	154
CHAPTER 8	Tasks and Threads.....	155
	OpenMP	157
	Fortran 2008.....	162
	Intel TBB	165
	hStreams.....	170
	Summary	171
	For More Information.....	172
CHAPTER 9	Vectorization.....	173
	Why Vectorize?	174
	How to Vectorize.....	174
	Three Approaches to Achieving Vectorization.....	174
	Six-Step Vectorization Methodology.....	176
	Streaming Through Caches: Data Layout, Alignment, Prefetching, and so on.....	178
	Compiler Tips	187
	Compiler Options	190
	Compiler Directives.....	192
	Use Array Sections to Encourage Vectorization	206
	Look at What the Compiler Created: Assembly Code Inspection.....	209
	Numerical Result Variations with Vectorization.....	211
	Summary.....	211
	For More Information.....	211
CHAPTER 10	Vectorization Advisor	213
	Getting Started with Intel Advisor for Knights Landing.....	214
	Enabling and Improving AVX-512 Code with the Survey Report.....	216
	Memory Access Pattern Report	232
	AVX-512 Gather/Scatter Profiler.....	233
	Mask Utilization and FLOPs Profiler	236
	Advisor Roofline Report	238
	Explore AVX-512 Code Characteristics Without AVX-512 Hardware.....	240
	Example — Analysis of a Computational Chemistry Code	242

Summary	250
For More Information.....	250
CHAPTER 11 Vectorization with SDLT	251
What Is SDLT?.....	251
Getting Started.....	252
SDLT Basics.....	254
Example Normalizing 3d Points with SIMD.....	256
What Is Wrong with AOS Memory Layout and SIMD?	258
SIMD Prefers Unit-Stride Memory Accesses.....	259
Alpha-Blended Overlay Reference	260
Alpha-Blended Overlay With SDLT	263
Additional Features.....	266
Summary	266
For More Information.....	267
CHAPTER 12 Vectorization with AVX-512 Intrinsics.....	269
What Are Intrinsics?.....	269
AVX-512 Overview.....	274
Migrating From Knights Corner	277
AVX-512 Detection.....	278
Learning AVX-512 Instructions.....	281
Learning AVX-512 Intrinsics.....	281
Step-by-Step Example Using AVX-512 Intrinsics	283
Results Using Our Intrinsics Code.....	294
For More Information.....	295
CHAPTER 13 Performance Libraries	297
Intel Performance Library Overview	297
Intel Math Kernel Library Overview	299
Intel Data Analytics Library Overview.....	300
Together: MKL and DAAL.....	302
Intel Integrated Performance Primitives Library Overview.....	303
Intel Performance Libraries and Intel Compilers	305
Native (Direct) Library Usage	306
Offloading to Knights Landing While Using a Library	308
Precision Choices and Variations.....	312
Performance Tip for Faster Dynamic Libraries.....	313
For More Information.....	314

CHAPTER 14 Profiling and Timing	315
Introduction to Knight Landing Tuning.....	315
Event-Monitoring Registers	316
Efficiency Metrics	317
Potential Performance Issues.....	323
Intel VTune Amplifier XE Product.....	333
Performance Application Programming Interface	334
MPI Analysis: ITAC.....	334
HPCToolkit.....	335
Tuning and Analysis Utilities.....	335
Timing.....	335
Summary	337
For More Information.....	337
CHAPTER 15 MPI	339
Internode Parallelism.....	339
MPI on Knights Landing.....	339
MPI Overview	340
How to Run MPI Applications.....	341
Analyzing MPI Application Runs	347
Tuning of MPI Applications	352
Heterogeneous Clusters	355
Recent Trends in MPI Coding	357
Putting it all Together.....	362
Summary	365
For More Information.....	365
CHAPTER 16 PGAS Programming Models	369
To Share or not to Share	369
Why Use PGAS on Knights Landing?.....	372
Programming with PGAS.....	373
Performance Evaluation	378
Beyond PGAS.....	381
Summary	381
For More Information.....	382
CHAPTER 17 Software-Defined Visualization	383
Motivation for Software-Defined Visualization	384
Software-Defined Visualization Architecture	387
OpenSWR: OpenGL Raster-Graphics Software Rendering	388

Embree: High-Performance Ray Tracing Kernel Library	390
OSPRay: Scalable Ray Tracing Framework.....	392
Summary.....	399
Image Attributions.....	400
For More Information.....	400
CHAPTER 18 Offload to Knights Landing	403
Offload Programming Model—Using with Knights Landing.....	403
Processors Versus Coprocessor	404
Offload Model Considerations	405
OpenMP Target Directives.....	406
Concurrent Host and Target Execution.....	408
Offload Over Fabric	410
Summary.....	411
For More Information.....	411
CHAPTER 19 Power Analysis	413
Power Demand Gates Exascale.....	413
Power 101	415
Hardware-Based Power Analysis Techniques	416
Software-Based Knights Landing Power Analyzer	419
ManyCore Platform Software Package Power Tools	429
Running Average Power Limit	430
Performance Profiling on Knights Landing	434
Intel Remote Management Module.....	436
Summary.....	438
For More Information.....	439
 SECTION III PEARLS	
<hr/>	
CHAPTER 20 Optimizing Classical Molecular Dynamics in LAMMPS.....	443
Molecular Dynamics.....	443
LAMMPS.....	446
Knights Landing Processors	447
LAMMPS Optimizations.....	449
Data Alignment.....	449
Data Types and Layout	450
Vectorization.....	452
Neighbor List	459
Long-Range Electrostatics.....	462
MPI and OpenMP Parallelization	462

Performance Results	465
System, Build, and Run Configurations.....	465
Workloads	466
Organic Photovoltaic Molecules	467
Hydrocarbon Mixtures.....	467
Rhodopsin Protein in Solvated Lipid Bilayer.....	468
Coarse Grain Liquid Crystal Simulation.....	468
Coarse-Grain Water Simulation	468
Summary	469
Acknowledgment	470
For More Information.....	470
CHAPTER 21 High Performance Seismic Simulations	471
High-Order Seismic Simulations.....	472
Numerical Background.....	472
Application Characteristics	476
Intel Architecture as Compute Engine.....	484
Highly-Efficient Small Matrix Kernels.....	484
Sparse Matrix Kernel Generation and Sparse/Dense Kernel Selection	485
Dense Matrix Kernel Generation: AVX2	486
Dense Matrix Kernel Generation: AVX-512.....	487
Kernel Performance Benchmarking	489
Incorporating Knights Landing's Different Memory Subsystems.....	490
Performance Evaluation	493
Mount Merapi	493
1992 Landers	495
Summary and Take-Aways	497
For More Information.....	498
CHAPTER 22 Weather Research and Forecasting (WRF)	499
WRF Overview	499
WRF Execution Profile: Relatively Flat.....	500
History of WRF on Intel Many-Core (Intel Xeon Phi Product Line)	500
Our Early Experiences with WRF on Knights Landing.....	501
Compiling WRF for Intel Xeon and Intel Xeon Phi Systems...	503
WRF CONUS12km Benchmark Performance.....	504
MCDRAM Bandwidth.....	504
Vectorization: Boost of AVX-512 Over AVX2	507
Core Scaling	508

Summary	509
For More Information.....	509
CHAPTER 23 <i>N</i>-Body simulation	511
Parallel Programming for Noncomputer Scientists	511
Step-by-Step Improvements	512
<i>N</i> -Body Simulation	513
Optimization	515
Initial Implementation (Optimization Step 0).....	515
Thread Parallelism (Optimization Step 1)	516
Scalar Performance Tuning (Optimization Step 2)	518
Vectorization with SOA (Optimization Step 3).....	519
Memory Traffic (Optimization Step 4).....	521
Impact of MCDRAM on Performance.....	523
Summary	524
For More Information.....	525
CHAPTER 24 Machine Learning	527
Convolutional Neural Networks.....	528
OverFeat-FAST Results.....	538
For More Information.....	548
CHAPTER 25 Trinity Workloads	549
Out of the Box Performance	549
Optimizing MiniGhost OpenMP Performance	571
Summary	578
For More Information.....	579
CHAPTER 26 Quantum Chromodynamics	581
LQCD.....	581
The QPhiX Library and Code Generator.....	582
Wilson-Dslash Operator	583
Configuring the QPhiX Code Generator.....	586
The Experimental Setup	589
Results.....	590
Conclusion	597
For More Information.....	597
Contributors	599
Glossary	613
Index	623