1. Record Nr. UNINA9910828339503321

Titolo	Heterogeneous computing with OpenCL / / Benedict R. Gaster ... [et al.]
Pubbl/distr/stampa	Amsterdam ; ; Boston, : Elsevier/MK, c2013
ISBN	1-283-71652-6 0-12-405520-6
Edizione	[Rev. OpenCL 1.2 ed.]
Descrizione fisica	1 online resource (309 p.)
Altri autori (Persone)	GasterBenedict R
Disciplina	005.2752
Soggetti	OpenCL (Computer program language) Parallel programming (Computer science)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Front Cover; Heterogeneous Computing with OpenCL; Copyright; Contents; Foreword to the Revised OpenCL 1.2 Edition; Foreword to the First Edition; Preface; Our Heterogeneous World; OpenCL; This Text; Acknowledgments; About the Authors; Chapter 1: Introduction to Parallel Programming; Introduction; OpenCL; The Goals of This Book; Thinking Parallel; Concurrency and Parallel Programming Models; Threads and Shared Memory; Message-Passing Communication; Different Grains of Parallelism; Data Sharing and Synchronization; Structure; Reference; Further Reading and Relevant Websites Chapter 2: Introduction to OpenCL Introduction; The OpenCL Standard; The OpenCL Specification; Kernels and the OpenCL Execution Model; Platform and Devices; Host-Device Interaction; The Execution Environment; Contexts; Command Queues; Events; Memory Objects; Buffers; Images; Flush and Finish; Creating an OpenCL Program Object; The OpenCL Kernel; Memory Model; Writing Kernels; Full Source Code Example for Vector Addition; Vector Addition with C++ Wrapper; Summary; Reference; Chapter 3: OpenCL Device Architectures; Introduction; Hardware trade-offs Performance Increase by Frequency, and Its Limitations Superscalar Execution; VLIW; SIMD and Vector Processing; Hardware Multithreading; Multi-Core Architectures; Integration: Systems-on-Chip and the APU; Cache Hierarchies and Memory Systems; The architectural design space; CPU Designs; Low-Power CPUs; Mainstream Desktop CPUs; Intel

Itanium 2; Niagara; GPU Architectures; Handheld GPUs; At the High End: AMD Radeon HD7970 and NVIDIA GTX580; APU and APU-Like Designs; Summary; References; Chapter 4: Basic OpenCL Examples; Introduction; Example Applications; Simple Matrix Multiplication Example
Step 1: Set Up Environment Step 2: Declare Buffers and Move Data; Step 3: Run time Kernel Compilation; Step 4: Run the Program; Step 5: Return Results to Host; Image Rotation Example; Step 1: Set Up Environment; Step 2: Declare Buffers and Move Data; Step 3: Run time Kernel Compilation; Step 4: Run the Program; Step 5: Read Result Back to Host; Image Convolution Example; Step 1: Create Image and Buffer Objects; Step 2: Write the Input Data; Step 3: Create Sampler Object; Step 4: Compile and Execute the Kernel; Step 5: Read the Result; The Convolution Kernel; Compiling OpenCL Host Applications
Summary Chapter 5: Understanding OpenCL's Concurrency and Execution Model; Introduction; Kernels, Work-Items, Workgroups, and the Execution Domain; OpenCL Synchronization: Kernels, Fences, and Barriers; Queuing and Global Synchronization; Memory Consistency in OpenCL; Events; Command Queues to Multiple Devices; Event Uses beyond Synchronization; User Events; Event Callbacks; Native Kernels; Command Barriers and Markers; The Host-Side Memory Model; Buffers; Manipulating Buffer Objects; Images; The Device-Side Memory Model; Device-Side Relaxed Consistency; Global Memory; Local Memory Constant Memory

Sommario/riassunto

Heterogeneous Computing with OpenCL teaches OpenCL and parallel programming for complex systems that may include a variety of device architectures: multi-core CPUs, GPUs, and fully-integrated Accelerated Processing Units (APUs) such as AMD Fusion technology. Designed to work on multiple platforms and with wide industry support, OpenCL will help you more effectively program for a heterogeneous future. Written by leaders in the parallel computing and OpenCL communities, this book will give you hands-on OpenCL experience to address a range of fundamental parallel algorithms. The