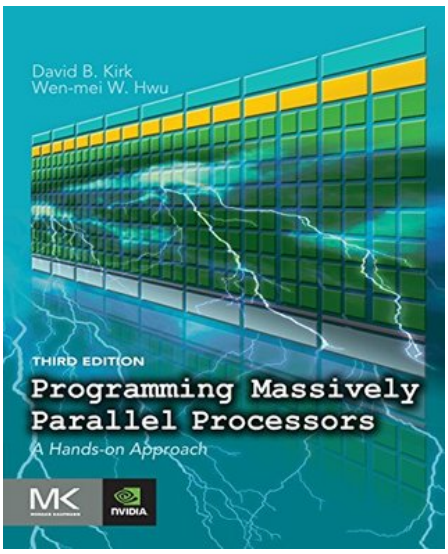

Programming Massively Parallel Processors: A Hands-On Approach

David B. Kirk , Wen-Mei W Hwu

Programming Massively Parallel Processors: A Hands-On Approach

David B. Kirk , Wen-Mei W Hwu

Programming Massively Parallel Processors: A Hands-On Approach David B. Kirk , Wen-Mei W Hwu *Programming Massively Parallel Processors: A Hands-on Approach, Third Edition* shows both student and professional alike the basic concepts of parallel programming and GPU architecture, exploring, in detail, various techniques for constructing parallel programs.

Case studies demonstrate the development process, detailing computational thinking and ending with effective and efficient parallel programs. Topics of performance, floating-point format, parallel patterns, and dynamic parallelism are covered in-depth.

For this new edition, the authors have updated their coverage of CUDA, including coverage of newer libraries, such as CuDNN, moved content that has become less important to appendices, added two new chapters on parallel patterns, and updated case studies to reflect current industry practices.

Teaches computational thinking and problem-solving techniques that facilitate high-performance parallel computing
Utilizes CUDA version 7.5, NVIDIA's software development tool created specifically for massively parallel environments
Contains new and updated case studies
Includes coverage of newer libraries, such as CuDNN for Deep Learning

Programming Massively Parallel Processors: A Hands-On Approach Details

Date	:	Published December 21st 2016 by Morgan Kaufmann Publishers (first published January 1st 2010)
ISBN	:	9780128119860
Author	:	David B. Kirk , Wen-Mei W Hwu
Format	:	Paperback 576 pages
Genre	:	Computer Science, Programming, Technical, Science, Textbooks

From Reader Review Programming Massively Parallel Processors: A Hands-On Approach for online ebook

Peter Aronson says

Not a bad introductory book, (if perhaps excessively NVIDIA-centric), but the first edition (which is what I read) is showing its age in 2018.

Steve says

A somewhat light overview of nVidia's GPU architecture and CUDA programming model.

The explanation of GPU architecture was clear if somewhat vague. Given that the whole book was focused on the GT200 series, and they referenced the GTX 295 specifically, I wish they had given exact latency numbers for the memory hierarchy and other hardware features as motivation for why you would do all the backflips necessary to fit your problem effectively into the cuda model. There are a few log-log graphs in the book to show you just how much speed a naive algorithm will give up but they could have done better.

The chapter on floating point was unnecessary, I felt. There was a brief discussion at the end of it about how sorting can affect accuracy due to fixed precision but hardly gave enough detail for the programmer to use effectively.

It has an appendix mapping the CUDA primitives and organization you've learned to their OpenCL equivalents.

Sibongiseni says

commetnt

Andrewcharles420 says

Useful explanations of how CUDA programming works. Nice to see more examples and depth than the software documentation. Would like to have seen mention of using the texture memory and associated functions, which were entirely absent.

Alex Ott says

This book provides enough information to start develop programs using CUDA technology. Besides CUDA specific information, it explains how to design algorithms for massively parallel hardware, how to optimize

performance, and provides detailed description of two tasks, implemented on GPU. There is also a chapter, that provides short overview of OpenCL technology

---

Phillip Nordwall says

Taught be some about the different memory model issues that can arrise with CUDA. This book asn't put together well. Most of the chapters seemed to spend a couple pages setting the stage as if they all originally stood alone.

Medhat Rabie says

Reading the book with the offered course on Coursera will be a very good start. It is written in a very simple way and good for beginners. little bit general but good enough.
