

Robotic Computing on FPGAs



Synthesis Lectures on Computer Architecture

Editor

Natalie Enright Jerger, *University of Toronto*

Editor Emerita

Margaret Martonosi, *Princeton University*

Founding Editor Emeritus

Mark D. Hill, *University of Wisconsin, Madison*

Synthesis Lectures on Computer Architecture publishes 50- to 100-page books on topics pertaining to the science and art of designing, analyzing, selecting, and interconnecting hardware components to create computers that meet functional, performance, and cost goals. The scope will largely follow the purview of premier computer architecture conferences, such as ISCA, HPCA, MICRO, and ASPLOS.

Robotic Computing on FPGAs

Shaoshan Liu, Zishen Wan, Bo Yu, and Yu Wang
2021

AI for Computer Architecture: Principles, Practice, and Prospects

Lizhong Chen, Drew Penney, and Daniel Jiménez
2020

Deep Learning Systems: Algorithms, Compilers, and Processors for Large-Scale Production

Andres Rodriguez
2020

Parallel Processing, 1980 to 2020

Robert Kuhn and David Padua
2020

Data Orchestration in Deep Learning Accelerators

Tushar Krishna, Hyoukjun Kwon, Angshuman Parashar, Michael Pellauer, and Ananda Samajdar
2020

Efficient Processing of Deep Neural Networks

Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer
2020

Quantum Computer System: Research for Noisy Intermediate-Scale Quantum Computers

Yongshan Ding and Frederic T. Chong
2020

A Primer on Memory Consistency and Cache Coherence, Second Edition

Vijay Nagarajan, Daniel J. Sorin, Mark D. Hill, and David Wood
2020

Innovations in the Memory System

Rajeev Balasubramonian
2019

Cache Replacement Policies

Akanksha Jain and Calvin Lin
2019

The Datacenter as a Computer: Designing Warehouse-Scale Machines, Third Edition

Luiz André Barroso, Urs Hölzle, and Parthasarathy Ranganathan
2018

Principles of Secure Processor Architecture Design

Jakub Szefer
2018

General-Purpose Graphics Processor Architectures

Tor M. Aamodt, Wilson Wai Lun Fung, and Timothy G. Rogers
2018

Compiling Algorithms for Heterogenous Systems

Steven Bell, Jing Pu, James Hegarty, and Mark Horowitz
2018

Architectural and Operating System Support for Virtual Memory

Abhishek Bhattacharjee and Daniel Lustig
2017

Deep Learning for Computer Architects

Brandon Reagen, Robert Adolf, Paul Whatmough, Gu-Yeon Wei, and David Brooks
2017

On-Chip Networks, Second Edition

Natalie Enright Jerger, Tushar Krishna, and Li-Shiuan Peh
2017

Space-Time Computing with Temporal Neural Networks

James E. Smith

2017

Hardware and Software Support for Virtualization

Edouard Bugnion, Jason Nieh, and Dan Tsafir

2017

Datacenter Design and Management: A Computer Architect's Perspective

Benjamin C. Lee

2016

A Primer on Compression in the Memory Hierarchy

Somayeh Sardashti, Angelos Arelakis, Per Stenström, and David A. Wood

2015

Research Infrastructures for Hardware Accelerators

Yakun Sophia Shao and David Brooks

2015

Analyzing Analytics

Rajesh Bordawekar, Bob Blainey, and Ruchir Puri

2015

Customizable Computing

Yu-Ting Chen, Jason Cong, Michael Gill, Glenn Reinman, and Bingjun Xiao

2015

Die-stacking Architecture

Yuan Xie and Jishen Zhao

2015

Single-Instruction Multiple-Data Execution

Christopher J. Hughes

2015

Power-Efficient Computer Architectures: Recent Advances

Magnus Sjalander, Margaret Martonosi, and Stefanos Kaxiras

2014

FPGA-Accelerated Simulation of Computer Systems

Hari Angepat, Derek Chiou, Eric S. Chung, and James C. Hoe

2014

A Primer on Hardware Prefetching

Babak Falsafi and Thomas F. Wenisch

2014

On-Chip Photonic Interconnects: A Computer Architect's Perspective

Christopher J. Nitta, Matthew K. Farrens, and Venkatesh Akella

2013

Optimization and Mathematical Modeling in Computer Architecture

Tony Nowatzki, Michael Ferris, Karthikeyan Sankaralingam, Cristian Estan, Nilay Vaish, and David Wood

2013

Security Basics for Computer Architects

Ruby B. Lee

2013

The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, Second Edition

Luiz André Barroso, Jimmy Clidaras, and Urs Hölzle

2013

Shared-Memory Synchronization

Michael L. Scott

2013

Resilient Architecture Design for Voltage Variation

Vijay Janapa Reddi and Meeta Sharma Gupta

2013

Multithreading Architecture

Mario Nemirovsky and Dean M. Tullsen

2013

Performance Analysis and Tuning for General Purpose Graphics Processing Units (GPGPU)

Hyesoon Kim, Richard Vuduc, Sara Baghsorkhi, Jee Choi, and Wen-mei Hwu

2012

Automatic Parallelization: An Overview of Fundamental Compiler Techniques

Samuel P. Midkiff

2012

Phase Change Memory: From Devices to Systems

Moinuddin K. Qureshi, Sudhanva Gurumurthi, and Bipin Rajendran

2011

Multi-Core Cache Hierarchies

Rajeev Balasubramonian, Norman P. Jouppi, and Naveen Muralimanohar

2011

A Primer on Memory Consistency and Cache Coherence

Daniel J. Sorin, Mark D. Hill, and David A. Wood

2011

Dynamic Binary Modification: Tools, Techniques, and Applications

Kim Hazelwood

2011

Quantum Computing for Computer Architects, Second Edition

Tzvetan S. Metodi, Arvin I. Faruque, and Frederic T. Chong

2011

High Performance Datacenter Networks: Architectures, Algorithms, and Opportunities

Dennis Abts and John Kim

2011

Processor Microarchitecture: An Implementation Perspective

Antonio González, Fernando Latorre, and Grigorios Magklis

2010

Transactional Memory, Second Edition

Tim Harris, James Larus, and Ravi Rajwar

2010

Computer Architecture Performance Evaluation Methods

Lieven Eeckhout

2010

Introduction to Reconfigurable Supercomputing

Marco Lanzagorta, Stephen Bique, and Robert Rosenberg

2009

On-Chip Networks

Natalie Enright Jerger and Li-Shiuan Peh

2009

The Memory System: You Can't Avoid It, You Can't Ignore It, You Can't Fake It

Bruce Jacob

2009

Fault Tolerant Computer Architecture

Daniel J. Sorin

2009

The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines

Luiz André Barroso and Urs Hölzle

2009

Computer Architecture Techniques for Power-Efficiency

Stefanos Kaxiras and Margaret Martonosi

2008

Chip Multiprocessor Architecture: Techniques to Improve Throughput and Latency

Kunle Olukotun, Lance Hammond, and James Laudon

2007

Transactional Memory

James R. Larus and Ravi Rajwar

2006

Quantum Computing for Computer Architects

Tzvetan S. Metodiev and Frederic T. Chong

2006

© Springer Nature Switzerland AG 2022
Reprint of original edition ©Morgan & Claypool 2021

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopy, recording, or any other except for brief quotations in printed reviews, without the prior permission of the publisher.

Robotic Computing on FPGAs
Shaoshan Liu, Zishen Wan, Bo Yu, and Yu Wang

ISBN: 978-3-031-00643-2 paperback
ISBN: 978-3-031-01771-1 ebook
ISBN: 978-3-031-00068-3 hardcover

DOI 10.1007/978-3-031-01771-1

A Publication in the Springer series
SYNTHESIS LECTURES ON COMPUTER ARCHITECTURE

Lecture #56
Series Editor: Natalie Enright Jerger, *University of Toronto*
Editor Emerita: Margaret Martonosi, *Princeton University*
Founding Editor Emeritus: Mark D. Hill, *University of Wisconsin, Madison*
Series ISSN
Print 1935-3235 Electronic 1935-3243

Robotic Computing on FPGAs

Shaoshan Liu
PerceptIn

Zishen Wan
Georgia Institute of Technology

Bo Yu
PerceptIn

Yu Wang
Tsinghua University

SYNTHESIS LECTURES ON COMPUTER ARCHITECTURE #56

ABSTRACT

This book provides a thorough overview of the state-of-the-art field-programmable gate array (FPGA)-based robotic computing accelerator designs and summarizes their adopted optimized techniques. This book consists of ten chapters, delving into the details of how FPGAs have been utilized in robotic perception, localization, planning, and multi-robot collaboration tasks. In addition to individual robotic tasks, this book provides detailed descriptions of how FPGAs have been used in robotic products, including commercial autonomous vehicles and space exploration robots.

KEYWORDS

robotics, FPGAs, autonomous machines, perception, localization, planning, control, space exploration, deep learning

Contents

	Preface	xv
1	Introduction and Overview	1
1.1	Sensing	2
1.2	Perception	4
1.3	Localization	5
1.4	Planning and Control	6
1.5	FPGAs in Robotic Applications	7
1.6	The Deep Processing Pipeline	7
1.7	Summary	9
2	FPGA Technologies	11
2.1	An Introduction to FPGA Technologies	11
2.1.1	Types of FPGAs	12
2.1.2	FPGA Architecture	12
2.1.3	Commercial Applications of FPGAs	14
2.2	Partial Reconfiguration	15
2.2.1	What is Partial Reconfiguration?	16
2.2.2	How to Use Partial Reconfiguration?	16
2.2.3	Achieving High Performance	19
2.2.4	Real-World Case Study	23
2.3	Robot Operating System (ROS) on FPGAs	23
2.3.1	Robot Operating System (ROS)	24
2.3.2	ROS-Compliant FPGAs	26
2.3.3	Optimizing Communication Latency for the ROS-Compliant FPGAs	27
2.4	Summary	29
3	Perception on FPGAs – Deep Learning	31
3.1	Why Choose FPGAs for Deep Learning?	32
3.2	Preliminary: Deep Neural Network	33

3.3	Design Methodology and Criteria	34
3.4	Hardware-Oriented Model Compression	37
3.4.1	Data Quantization	37
3.4.2	Weight Reduction	39
3.5	Hardware Design: Efficient Architecture	40
3.5.1	Computation Unit Designs	40
3.5.2	Loop Unrolling Strategies	44
3.5.3	System Design	46
3.6	Evaluation	49
3.7	Summary	53
4	Perception on FPGAs – Stereo Vision	55
4.1	Perception in Robotics	55
4.2	Stereo Vision in Robotics	58
4.3	Local Stereo Matching on FPGAs	59
4.3.1	Algorithm Framework	59
4.3.2	FPGA Designs	60
4.4	Global Stereo Matching on FPGAs	61
4.4.1	Algorithm Framework	61
4.4.2	FPGA Designs	61
4.5	Semi-Global Matching on FPGAs	62
4.5.1	Algorithm Framework	62
4.5.2	FPGA Designs	62
4.6	Efficient Large-Scale Stereo Matching on FPGAs	63
4.6.1	ELAS Algorithm Framework	63
4.6.2	FPGA Designs	66
4.7	Evaluation and Discussion	68
4.7.1	Dataset and Accuracy	68
4.7.2	Power and Performance	69
4.8	Summary	69
5	Localization on FPGAs	73
5.1	Preliminary	73
5.1.1	Context	73
5.1.2	Algorithm Overview	75
5.2	Algorithm Framework	78

5.3	Frontend FPGA Design	81
5.3.1	Overview	81
5.3.2	Exploiting Task-Level Parallelisms	82
5.4	Backend FPGA Design	83
5.5	Evaluation	85
5.5.1	Experimental Setup	85
5.5.2	Resource Consumption	86
5.5.3	Performance	87
5.6	Summary	88
6	Planning on FPGAs	91
6.1	Motion Planning Context Overview	91
6.1.1	Probabilistic Roadmap	92
6.1.2	Rapidly Exploring Random Tree	93
6.2	Collision Detection on FPGAs	94
6.2.1	Motion Planning Compute Time Profiling	94
6.2.2	General Purpose Processor-Based Solutions	95
6.2.3	Specialized Hardware Accelerator-Based Solutions	97
6.2.4	Evaluation and Discussion	103
6.3	Graph Search on FPGAs	106
6.4	Summary	107
7	Multi-Robot Collaboration on FPGAs	109
7.1	Multi-Robot Exploration	109
7.2	INCAME Framework for Multi-Task on FPGAs	113
7.2.1	Hardware Resource Conflicts in ROS	113
7.2.2	Interruptible Accelerator with ROS (INCAME)	115
7.3	Virtual Instruction-Based Accelerator Interrupt	117
7.3.1	Instruction Driven Accelerator	117
7.3.2	How to Interrupt: Virtual Instruction	119
7.3.3	Where to Interrupt: After SAVE/CALC_F	121
7.3.4	Latency Analysis	122
7.3.5	Virtual Instruction ISA (VI-ISA)	124
7.3.6	Instruction Arrangement Unit (IAU)	125
7.3.7	Example of Virtual Instruction	125
7.4	Evaluation and Results	127
7.4.1	Experiment Setup	127

	7.4.2	Virtual Instruction-Based Interrupts	128
	7.4.3	ROS-Based MR-Exploration	131
	7.5	Summary	131
8		Autonomous Vehicles Powered by FPGAs	133
	8.1	The PerceptIn Case Study	133
	8.2	Design Constraints	134
	8.2.1	Overview of the Vehicle	134
	8.2.2	Performance Requirements	135
	8.2.3	Energy and Cost Considerations	136
	8.3	Software Pipeline	138
	8.4	On Vehicle Processing System	140
	8.4.1	Hardware Design Space Exploration	140
	8.4.2	Hardware Architecture	142
	8.4.3	Sensor Synchronization	144
	8.4.4	Performance Characterizations	146
	8.5	Summary	147
9		Space Robots Powered by FPGAs	149
	9.1	Radiation Tolerance for Space Computing	149
	9.2	Space Robotic Algorithm Acceleration on FPGAs	151
	9.2.1	Feature Detection and Matching	152
	9.2.2	Stereo Vision	153
	9.2.3	Deep Learning	153
	9.3	Utilization of FPGAs in Space Robotic Missions	154
	9.3.1	Mars Exploration Rover Missions	155
	9.3.2	Mars Science Laboratory Mission	156
	9.3.3	Mars 2020 Mission	157
	9.4	Summary	158
10		Conclusion	159
	10.1	What we Have Covered in This Book	159
	10.2	Looking Forward	160
		Bibliography	163
		Authors' Biographies	201

Preface

In this book, we provide a thorough overview of the state-of-the-art FPGA-based robotic computing accelerator designs and summarize their adopted optimized techniques. The authors combined have over 40 years of research experiences of utilizing FPGAs in robotic applications, both in academic research and commercial deployments. For instance, the authors have demonstrated that, by co-designing both the software and hardware, FPGAs can achieve more than 10× better performance and energy efficiency compared to the CPU and GPU implementations. The authors have also pioneered the utilization of the partial reconfiguration methodology in FPGA implementations to further improve the design flexibility and reduce the overhead. In addition, the authors have successfully developed and shipped commercial robotic products powered by FPGAs and the authors demonstrate that FPGAs have excellent potential and are promising candidates for robotic computing acceleration due to its high reliability, adaptability, and power efficiency.

The authors believe that FPGAs are the best compute substrate for robotic applications for several reasons. First, robotic algorithms are still evolving rapidly, and thus any ASIC-based accelerators will be months or even years behind the state-of-the-art algorithms. On the other hand, FPGAs can be dynamically updated as needed. Second, robotic workloads are highly diverse, thus it is difficult for any ASIC-based robotic computing accelerator to reach economies of scale in the near future. On the other hand, FPGAs are a cost effective and energy-effective alternative before one type of accelerator reaches economies of scale. Third, compared to systems on a chip (SoCs) that have reached economies of scale, e.g., mobile SoCs, FPGAs deliver a significant performance advantage. Fourth, partial reconfiguration allows multiple robotic workloads to time-share an FPGA, thus allowing one chip to serve multiple applications, leading to overall cost and energy reduction.

Specifically, FPGAs require little power and are often built into small systems with less memory. They have the ability of massively parallel computations and to make use of the properties of perception (e.g., stereo matching), localization (e.g., simultaneous localization and mapping (SLAM)), and planning (e.g., graph search) kernels to remove additional logic so as to simplify the end-to-end system implementation. Taking into account hardware characteristics, several algorithms are proposed which can be run in a hardware-friendly way and achieve similar software performance. Therefore, FPGAs are possible to meet real-time requirements while achieving high energy efficiency compared to central processing units (CPUs) and graphics processing units (GPUs). In addition, unlike the application-specific integrated circuit (ASIC) counterparts, FPGA technologies provide the flexibility of on-site programming and re-programming without going through re-fabrication with a modified design. Partial Recon-

figuration (PR) takes this flexibility one step further, allowing the modification of an operating FPGA design by loading a partial configuration file. Using PR, part of the FPGA can be reconfigured at runtime without compromising the integrity of the applications running on those parts of the device that are not being reconfigured. As a result, PR can allow different robotic applications to time-share part of an FPGA, leading to energy and performance efficiency, and making FPGA a suitable computing platform for dynamic and complex robotic workloads. Due to the advantages over other compute substrates, FPGAs have been successfully utilized in commercial autonomous vehicles as well as in space robotic applications, for FPGAs offer unprecedented flexibility and significantly reduced the design cycle and development cost.

This book consists of ten chapters, providing a thorough overview of how FPGAs have been utilized in robotic perception, localization, planning, and multi-robot collaboration tasks. In addition to individual robotic tasks, we provide detailed descriptions of how FPGAs have been used in robotic products, including commercial autonomous vehicles and space exploration robots.

Shaoshan Liu
June 2021