RECONFIGURABLE COMPUTING

THE THEORY AND PRACTICE OF FPGA-BASED COMPUTATION

Edited by

Scott Hauck and André DeHon

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW DELHI • NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann Publishers is an imprint of Elsevier

Reconfigurable Computing
Hauck and DeHon

MORGAN KAUFMANN PUBLISHERS
An imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803-4255

Original ISBN: 978-0-12-370522-8

First Printed in India 2011

Indian Reprint ISBN: 978-93-80931-86-9

This edition has been authorized by Elsevier for sale in the following countries: India, Pakistan, Nepal, Sri Lanka and Bangladesh. Sale and purchase of this book outside these countries is not authorized and is illegal.

Published by Elsevier, a division of Reed Elsevier India Private Limited.

Registered Office: 622, Indraprakash Building, 21 Barakhamba Road, New Delhi-110 001.
Corporate Office: 14th floor, Building No. 10B, DLF Cyber City Phase-II, Gurgaon-122 002, Haryana, India.

Printed and bound in India by Sanat Printers, Kundli-131 028

CHAPTER 21

IMPLEMENTING APPLICATIONS WITH FPGAS

Brad L. Hutchings, Brent E. Nelson
Department of Electrical and Computer Engineering
Brigham Young University

Developers can choose various devices when implementing electronic systems: field-programmable gate arrays (FPGAs), microprocessors, and other standard products such as ASSPs, and custom chips or application-specific integrated circuits (ASICs). This chapter discusses how FPGAs compare to other digital devices, outlines the considerations that will help designers to determine when FPGAs are appropriate for a specific application, and presents implementation strategies that exploit features specific to FPGAs.

The chapter is divided into four major sections. Section 21.1 discusses the strengths and weaknesses of FPGAs, relative to other available devices. Section 21.2 suggests when FPGA devices are suitable choices for specific applications/algorithms, based upon their I/O and computation requirements. Section 21.3 discusses general implementation strategies appropriate for FPGA devices. Then Section 21.4 discusses FPGA-specific arithmetic design techniques.

21.1 STRENGTHS AND WEAKNESSES OF FPGAS

Developers can choose from three general classes of devices when implementing an algorithm or application: microprocessor, FPGA, or ASIC (for simplicity, ASSPs are not considered here). This section provides a brief summary of the advantages and disadvantages of these devices in terms of time to market, cost, development time, power consumption, and debug and verification.

21.1.1 Time to Market

Time to market is often touted as one of the FPGA's biggest strengths, at least relative to ASICs. With an ASIC, from specification to product requires (at least): (1) design, (2) verification, (3) fabrication, (4) packaging, and (5) device test. In addition, software development requires access to the ASIC device (or an emulation of such) before it can be verified and completed. As immediately available standard devices, FPGAs have already been fabricated, packaged, and tested by the vendor, thereby eliminating at least four months from time to market.

More difficult to quantify but perhaps more important are: (1) refabrications (respins) caused by either errors in the design or late changes to the specification, due to a change in an evolving standard, for example, and (2) software development schedules that depend on access to the ASIC. Both of these items impact product production schedules; a respin can easily consume an additional four months, and early access to hardware can greatly accelerate software development and debug, particularly for the embedded software that communicates directly with the device.

In light of these considerations, a conservative estimate of the time-to-market advantage of FPGAs relative to ASICs is 6 to 12 months. Such a reduction is significant; in consumer electronics markets, many products have only a 24-month lifecycle.

21.1.2 Cost

Per device, FPGAs can be much less expensive than ASICs, especially in lower volumes, because the nonrecurring costs of FPGA fabrication are borne by many users. However, because of their reprogrammability, FPGAs require much more silicon area to implement equivalent functionality. Thus, at the highest volumes possible in consumer electronics, FPGA device cost will eventually exceed ASIC device cost.

21.1.3 Development Time

FPGA application development is most often approached as hardware design: applications are described in Verilog or VHDL, simulated to determine correctness, and synthesized using commercial logic synthesis tools. Commercial tools are available that synthesize behavioral programs written in sequential languages such as C to FPGAs. However, in most cases, much better performance and higher densities are achieved using HDLs, because they allow the user to directly describe and exploit the intrinsic parallelism available in an application. Exploiting application parallelism is the single best way to achieve high FPGA performance. However, designing highly parallel implementations of applications in HDLs requires significantly more development effort than software development with conventional sequential programming languages such as Java or C++.

21.1.4 Power Consumption

FPGAs consume more power than ASICs simply because programmability requires many more transistors, relative to a customized integrated circuit (IC). FPGAs may consume more or less power than a microprocessor or digital signal processor (DSP), depending on the application.

21.1.5 Debug and Verification

FPGAs are developed with standard hardware design techniques and tools. Coded in VHDL or Verilog and synthesized, FPGA designs can be debugged

in simulators just as typical ASIC designs are. However, many designers verify their designs directly, by downloading them into an FPGA and testing them in a system. With this approach the application can be tested at speed (a million times faster than simulation) in the actual operating environment, where it is exposed to real-world conditions. If thorough, this testing provides a stronger form of functional verification than simulation. However, debugging applications in an FPGA can be difficult because vendor tools provide much less observability and controllability than, for example, an hardware description language (HDL) simulator.

21.1.6 FPGAs and Microprocessors

As discussed previously, FPGAs are most often contrasted with custom ASICs. However, if a programmable solution is dictated because of changing application requirements or other factors, it is important to study the application carefully to determine if it is possible to meet performance requirements with a programmable processor—microprocessor or DSP. Code development for programmable processors requires much less effort than that required for FPGAs or ASICs, because developing software with sequential languages such as C or Java is much less taxing than writing parallel descriptions with Verilog or VHDL. Moreover, the coding and debugging environments for programmable processors are far richer than their HDL counterparts. Microprocessors are also generally much less expensive than FPGAs. If the microprocessor can meet application requirements (performance, power, etc.), it is almost always the best choice.

In general, FPGAs are well suited to applications that demand extremely high performance and reprogrammability, for interfacing components that communicate with many other devices (so-called glue-logic) and for implementing hardware systems at volumes that make their economies of scale feasible. They are less well suited to products that will be produced at the highest possible volumes or for systems that must run at the lowest possible power.

21.2 APPLICATION CHARACTERISTICS AND PERFORMANCE

Application performance is largely determined by the computational and I/O requirements of the system. Computational requirements dictate how much hardware parallelism can be used to increase performance. I/O system limitations and requirements determine how much performance can actually be exploited from the parallel hardware.

21.2.1 Computational Characteristics and Performance

FPGAs can outperform today's processors only by exploiting massive amounts of parallelism. Their technology has always suffered from a significant clock-rate disadvantage; FPGA clock rates have always been slower than CPU clock rates by about a factor of 10. This remains true today, with clock rates for FPGAs

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.