Multi-core RISC Processor Design and Implementation (Rev. 1.00)

ELEC5881M - Interim Report

Ben David Lancaster
Student ID: 201280376

Submitted in accordance with the requirements for the degree of
Master of Science (MSc)
in Embedded Systems Engineering

Supervisor: Dr. David Cowell
Assessor: Mr David Moore

University of Leeds
School of Electrical and Electronic Engineering

April 11, 2019

Revision History

Date	Version	Changes
20/05/2018	3.14	Add background research to appendix.
19/05/2018	3.13	Update abstract to align with guidelines.
19/05/2018	3.12	Fix ISA pseudo-codes.
11/03/2018	1.00	Initial section outline.

Table 1: Document revisions.

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Acknowledgements

I would like to thank my project supervisors Nigel Barlow and Serge Thill for their support and guidance throughout this project.

I would also like to thank James Spalding (Spirent Communications) and firmware team for their encouragement, ideas, and industrial sponsorship supporting this final project.

Declaration of Academic Integrity

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated in the report. The candidate confirms that appropriate credit has been given within the report where reference has been made to the work of others.

This copy has been supplied on the understanding that no quotation from the report may be published without proper acknowledgement. The candidate, however, confirms his/her consent to the University of Leeds copying and distributing all or part of this work in any forms and using third parties, who might be outside the University, to monitor breaches of regulations, to verify whether this work contains plagiarised material, and for quality assurance purposes.

The candidate confirms that the details of any mitigating circumstances have been submitted to the Student Support Office at the School of Electronic and Electrical Engineering, at the University of Leeds.

Name: Ben David Lancaster
Date: April 11, 2019

Table of Contents

[1] [2] [3]

References

[1] J. Balkind, M. McKeown, Y. Fu, T. Nguyen, Y. Zhou, A. Lavrov, M. Shahrad, A. Fuchs, S. Payne, X. Liang *et al.*, "Openpiton: An open source manycore research framework," in *ACM SIGARCH Computer Architecture News*, vol. 44, no. 2. ACM, 2016, pp. 217–232.

[2] N. Satish, M. Harris, and M. Garland, "Designing efficient sorting algorithms for manycore gpus," in *2009 IEEE International Symposium on Parallel & Distributed Processing*. IEEE, 2009, pp. 1–10.

[3] S. Binet, P. Calafiura, S. Snyder, W. Wiedenmann, and F. Winklmeier, "Harnessing multi-cores: Strategies and implementations in atlas," in *Journal of Physics: Conference Series*, vol. 219, no. 4. IOP Publishing, 2010, p. 042002.

Chapter 1

Abstract (not in toc)

Chapter 2

Declaration of Academic Integrity (not in toc)

Chapter 3

Acknowledgements (not in toc)

Chapter 4

Introduction

4.1 Why Multi-core?

4.2 Why RISC?

4.3 Why FPGA?

Chapter 5

Background

5.1 Single core vs. Multi-core vs. Many-core

5.2 Network-on-chip

5.2.1 OpenPiton

5.3 Summary

Chapter 6

Project Management

6.1 Project Goals

6.1.1 Core Deliverables (CD)

6.1.2 Extended Deliverables (ED)

6.2 Project Timeline

6.2.1 Project Stages

6.2.2 Timeline

6.3 Resources

6.3.1 Terasic DE1-SoC Development Board

6.3.2 Minispartan 6+ FPGA Development Board

Chapter 7

RISC Core Design

7.1 Design Goals

7.2 Instruction Set Architecture

7.2.1 Data Sizes

7.2.2 Registers

7.2.3 Endianness

7.3 High Level Design

7.4 Memory-mapped peripherals

7.4.1 Wishbone Master Bus

7.5 Optimisations

7.5.1 Pipelining

7.5.2 Stall Avoidance

7.6 Core Verification

7.7 Conclusion

Chapter 8

Multi-core Communication Design

8.1 Interconnect Goals

8.2 Interconnect Layout

8.2.1 Bus Design

8.2.2 Core Pin Assignments

8.3 Core-to-core Communication

8.4 Shared-Resource Control

8.4.1 Resource Scheduling

8.5 Verification

8.6 Conclusion

Chapter 9

Core Analysis

9.1 FPGA Implementation Analysis

9.1.1 Space/Resource Usage

9.1.2 Static Timing Analysis

9.2 Speed Analysis

9.2.1 Parallel Reduction Algorithms

9.2.2 Parallel DFT Algorithm

Chapter 10

Conclusion

10.1 RISC Core Review

10.2 Multi-core Review

10.3 Verification Review

10.4 Goal Review