An Optimizing Compiler for Low-Level Floating Point Operations

by

Eduardo A. Lundgren Melo

Submitted to the Center for Informatics
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science

at the

FEDERAL UNIVERSITY OF PERNAMBUCO

February 2013

Author .....
Center for Informatics
Mar 20, 2013

Certified by .....
Silvio de Barros Melo
Associate Professor
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

An Optimizing Compiler for Low-Level Floating Point Operations

by

Eduardo A. Lundgren Melo

Submitted to the Center for Informatics
on Mar 20, 2013, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science

Abstract

In this thesis, I designed and implemented a compiler which performs optimizations that reduce the number of low-level floating point operations necessary for a specific task; this involves the optimization of chains of floating point operations as well as the implementation of a "fixed" point data type that allows some floating point operations to simulated with integer arithmetic. The source language of the compiler is a subset of C, and the destination language is assembly language for a micro-floating point CPU. An instruction-level simulator of the CPU was written to allow testing of the code. A series of test pieces of codes was compiled, both with and without optimization, to determine how effective these optimizations were.

Thesis Supervisor: Silvio de Barros Melo
Title: Associate Professor

Acknowledgments

This is the acknowledgements section. You should replace this with your own acknowledgements. [1] foo [2].

This master thesis has been examined by a Committee as follows:

Professor Veronica Teichrieb .....
Chairman, Thesis Committee
Professor of Computer Science

Professor Daniel Cunha .....
Thesis Supervisor
Associate Computer Science

Professor Alvo Dumbledore .....
Member, Thesis Committee
Hogwarts School of Witchcraft and Wizardry

Contents

List of Figures

List of Tables

List of Acronyms

AJAX	Asynchronous JavaScript and XML
BAST	Bug Report Analysis and Search Tool
BTT	Bug Report Tracker Tool
BRN	Bug Report Network
CCB	Change Control Board

Chapter 1

Introduction

Micro-optimization is a technique to reduce the overall operation count of floating point operations. In a standard floating point unit, floating point operations are fairly high level, such as "multiply" and "add"; in a micro floating point unit (μFPU), these have been broken down into their constituent low-level floating point operations on the mantissas and exponents of the floating point numbers.

Chapter two describes the architecture of the μFPU unit, and the motivations for the design decisions made.

Chapter three describes the design of the compiler, as well as how the optimizations discussed in section 1.1 were implemented.

Chapter four describes the purpose of test code that was compiled, and which statistics were gathered by running it through the simulator. The purpose is to measure what effect the micro-optimizations had, compared to unoptimized code. Possible future expansions to the project are also discussed.

1.1 Description of micro-optimization

$$\tau(\mathbf{p}; x, y) := \begin{cases} 1 & \text{if} \quad \mathbf{p(x)} < \mathbf{p(y)}, \\ 0 & \text{otherwise} \end{cases}$$

Appendix A

Tables

Table A.1: Armadillos

Armadillos	are
our	friends

Bibliography

[1] Jeffrey N. Johnson and Paul F. Dubois. Issue tracking. *Computing in Science and Engineering*, 5(6):71–77, November 2003.

[2] Jeffrey N. Johnson and Paul F. Dubois. Issue tracking. *Computing in Science and Engineering*, 5(6):71–77, November 2003.