ARCHITECTURAL SUPPORT FOR A VARIABLE GRANULARITY CACHE MEMORY SYSTEM

by

Snehasish Kumar
B.Tech, Biju Patnaik University of Technology, 2010

A Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

© Snehasish Kumar 2012
SIMON FRASER UNIVERSITY
Fall 2012

All rights reserved.
However, in accordance with the *Copyright Act of Canada*, this work may be reproduced without authorization under the conditions for "Fair Dealing." Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.

APPROVAL

Name:	Snehasish Kumar
Degree:	Master of Science
Title of Thesis:	Architectural support for a variable granularity cache memory system
Examining Committee:	Dr. Hu Kaers Chair
	Dr. Arrvindh Shriraman, Senior Supervisor
	Dr. Alexandra Federova, Supervisor

Date Approved: _____

Partial Copyright Licence

The author, whose copyright is declared on the title page of this work, has granted to Simon Fraser University the right to lend this thesis, project or extended essay to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users.

The author has further granted permission to Simon Fraser University to keep or make a digital copy for use in its circulating collection (currently available to the public at the "Institutional Repository" link of the SFU Library website (www.lib.sfu.ca) at http://summit/sfu.ca and, without changing the content, to translate the thesis/project or extended essays, if technically possible, to any medium or format for the purpose of preservation of the digital work.

The author has further agreed that permission for multiple copying of this work for scholarly purposes may be granted by either the author or the Dean of Graduate Studies.

It is understood that copying or publication of this work for financial gain shall not be allowed without the author's written permission.

Permission for public performance, or limited permission for private scholarly use, of any multimedia materials forming part of this work, may have been granted by the author. This information may be found on the separately catalogued multimedia material and in the signed Partial Copyright Licence.

While licensing SFU to permit the above uses, the author retains copyright in the thesis, project or extended essays, including the right to change the work for subsequent purposes, including editing and publishing the work in whole or in part, and licensing other parties, as the author may desire.

The original Partial Copyright Licence attesting to these terms, and signed by this author, may be found in the original bound copy of this work, retained in the Simon Fraser University Archive.

Simon Fraser University Library
Burnaby, British Columbia, Canada

revised Fall 2011

Abstract

Memory in modern computing systems are hierarchial in nature. Maintaining a memory hierarchy enables the system to service frequently requested data from a small low latency store located close to the processor. The design paradigms of the memory hierachy have been mostly unchanged since their inception in the late 1960's. However in the meantime there have been significant changes in the tasks computers perform and the way they are programmed. Modern computing systems perform more data centric tasks and are programmed in higher level languages which introduce many layers of abstraction between the programmer and the system.

Waste in the memory hierarchy refers to the under utilised space in the memory system and consequently wasted energy and time. The data access patterns of modern workloads are increasingly less uniform which makes it hard to design a memory hierarchy with rigid design principles that performs optimally for a wide range of workloads. The problem is exacerbated by the implications of the growing fraction of dark silicon on a processor chip.

This dissertation proposes and evaluates the benefits of a novel architecture for the on chip memory hierarchy which would allow it to dynamically adapt to the requirements of the application. We propose a design that can support a variable number of cache blocks, each of a different granularity. It employs a novel organization that completely eliminates the tag array, treating the storage array as uniform and morphable between tags and data. This enables the cache to harvest space from unused words in blocks for additional tag storage, thereby supporting a variable number of tags (and correspondingly, blocks). The design adjusts individual cache line granularities according to the spatial locality in the application. It adapts to the appropriate granularity both for different data objects in an

application as well as for different phases of access to the same data.

Compared to a fixed granularity cache, improves cache utilization to 90% - 99% for most applications, saves miss rate by up to 73% at the L1 level and up to 88% at the LLC level, and reduces miss bandwidth by up to 84% at the L1 and 92% at the LLC. Correspondingly reduces on-chip memory hierarchy energy by as much as 36% and improves performance by as much as 50%.

To whomever whoever reads this!

"Don't worry, Gromit. Everything's under control!"

— *The Wrong Trousers*, Aardman Animations, *1993*

Acknowledgments

Here go all the people you want to thank.

Contents

Approval ii

Partial Copyright License iii

Abstract iv

Dedication vi

Quotation vii

Acknowledgments viii

Contents ix

List of Tables xii

List of Figures xiii

List of Programs xiv

Preface xv

1 Introduction 1

1.1 Cache Memory Systems 1

1.2 Motivation for Change 1

1.3 Dissertation Outline 1

2 Background **2**
2.1 Cache Memory Architecture . . . 2
2.1.1 Fundamental Building Blocks . . . 2
2.1.2 Standard Cache Operations . . . 2
2.1.3 Replacement Policies . . . 2
2.1.4 Hierarchial Cache Memory Systems . . . 2
2.1.5 On-chip Interconnection Network . . . 2
2.1.6 Multi Core and Coherence . . . 2
2.2 Related Work . . . 2
2.2.1 Sector Caches . . . 2
2.2.2 Cache Filtering . . . 2
2.2.3 Spatial Access Pattern Prediction . . . 2
2.2.4 Software based approaches . . . 2

3 Amoeba Cache Architecture **3**
3.1 Fundamental Building Blocks . . . 3
3.2 Cache Management . . . 3
3.2.1 Standard Operations . . . 3
3.2.2 Hardware Aids . . . 3
3.2.3 Replacement Policy . . . 3
3.3 Spatial Pattern Predictor . . . 3
3.3.1 Oracle . . . 3
3.3.2 Region based Approach . . . 3
3.3.3 Program Counter based Approach . . . 3
3.4 Hierarchial Amoeba Cache Architecture . . . 3
3.5 On-chip Interconnection Network . . . 3
3.6 Multi Core and Coherence . . . 3

4 Implementation **4**
4.1 Application Traces . . . 4
4.1.1 Intel Pin . . . 4
4.1.2 Generating a memory access trace . . . 4
4.1.3 Workload selection . . . 4
4.2 GEMS Infrastructure . . . 4

4.2.1 Introduction 4
4.2.2 Components 4
4.2.3 SLICC 4
4.2.4 Amoeba-Single Protocol 4

5 Evaluation **5**
5.1 Best Effort - Oracle 5
5.1.1 Miss Rate - Performance 5
5.1.2 Bandwidth - Energy 5
5.2 Amoeba Cache vs Other Approaches 5
5.2.1 Sector Caches 5
5.2.2 Sector Caches with Prefetching 5
5.2.3 Line Distillation 5
5.2.4 Multi Cache 5
5.3 A feasible online approach 5
5.4 Multi Core Shared Cache 5

6 Conclusion **6**
6.1 Summary 6
6.2 Future Work 6

Bibliography **6**

Index **8**

List of Tables

List of Figures

List of Programs

Preface

Here go all the interesting reasons why you decided to write this thesis.

Chapter 1

Introduction

1.1 Cache Memory Systems

1.2 Motivation for Change

1.3 Dissertation Outline

Chapter 2

Background

2.1 Cache Memory Architecture

2.1.1 Fundamental Building Blocks

2.1.2 Standard Cache Operations

2.1.3 Replacement Policies

2.1.4 Hierarchial Cache Memory Systems

2.1.5 On-chip Interconnection Network

2.1.6 Multi Core and Coherence

2.2 Related Work

2.2.1 Sector Caches

2.2.2 Cache Filtering

2.2.3 Spatial Access Pattern Prediction

2.2.4 Software based approaches

Chapter 3

Amoeba Cache Architecture

3.1 Fundamental Building Blocks

3.2 Cache Management

3.2.1 Standard Operations

3.2.2 Hardware Aids

3.2.3 Replacement Policy

3.3 Spatial Pattern Predictor

3.3.1 Oracle

3.3.2 Region based Approach

3.3.3 Program Counter based Approach

3.4 Hierarchial Amoeba Cache Architecture

3.5 On-chip Interconnection Network

3.6 Multi Core and Coherence

Chapter 4

Implementation

4.1 Application Traces

4.1.1 Intel Pin

4.1.2 Generating a memory access trace

4.1.3 Workload selection

4.2 GEMS Infrastructure

4.2.1 Introduction

4.2.2 Components

4.2.3 SLICC

4.2.4 Amoeba-Single Protocol

Chapter 5

Evaluation

5.1 Best Effort - Oracle

5.1.1 Miss Rate - Performance

5.1.2 Bandwidth - Energy

5.2 Amoeba Cache vs Other Approaches

5.2.1 Sector Caches

5.2.2 Sector Caches with Prefetching

5.2.3 Line Distillation

5.2.4 Multi Cache

5.3 A feasible online approach

5.4 Multi Core Shared Cache

Chapter 6

Conclusion

6.1 Summary

6.2 Future Work

Bibliography

[1] Hiyan Alshawi. *Memory and Context for Language Interpretation*. Studies in Natural Language Processing. Cambridge University Press: Cambridge, New York, 1987.

[2] Nicholas Asher. From discourse macro-structure to micro-structure and back again: Discourse semantics and the focus/background distinction. In Hans Kamp and Barbara H. Partee, editors, *Proceedings of the workshops on Context Dependence in the Analysis of Linguistic Meaning*, volume 1: Papers, pages 21–51. IMS, University of Stuttgart, 1995.

[3] Gennaro Chierchia. *Dynamics of Meaning: Anaphora, Presupposition and the Theory of Grammar*. The University of Chicago Press: Chicago, London, 1995.

[4] Herbert H. Clark and Susan E. Haviland. Comprehension and the given–new contract. In Roy O. Freedle, editor, *Discourse Production and Comprehension*, number 1 in Discourse Processes: Advances in Research and Theory, pages 1–40. Ablex Publishing Corporation: Norwood, NJ, 1977.

[5] Östen Dahl. Topic-comment structure revisited. In Östen Dahl, editor, *Topic and Comment, Contextual Boundness and Focus*, number 6 in Papers in Textlinguistics, pages 1–24. Helmut Buske Verlag: Hamburg, 1974.

[6] Paul Dekker and Herman Hendriks. Files in focus. In Elisabet Engdahl, editor, *Integrating Information Structure into Constraint-based and Categorial Approaches*, ESPRIT Basic Research Project 6852, Dynamic Interpretation of Natural Language, DYANA-2 Deliverable R1.3.B, pages 29–37. ILLC, University of Amsterdam, 1994.

[7] Elisabet Engdahl and Enric Vallduví. Information packaging and grammar architecture: A constraint-based approach. In Elisabet Engdahl, editor, *Integrating Information Structure into Constraint-based and Categorial Approaches*, ESPRIT Basic Research Project 6852, Dynamic Interpretation of Natural Language, DYANA-2 Deliverable R1.3.B, pages 41–79. ILLC, University of Amsterdam, 1994.

[8] Nomi Erteschik-Shir. *The Dynamics of Focus Structure*. Number 84 in Cambridge Studies in Linguistics. Cambridge University Press: Cambridge, New York, Melbourne, 1997.

[9] L. T. F. Gamut. *Logic, Language, and Meaning*, volume 1: Introduction to Logic. The University of Chicago Press: Chicago, London, 1991.

[10] L. T. F. Gamut. *Logic, Language, and Meaning*, volume 2: Intensional Logic and Logical Grammar. The University of Chicago Press: Chicago, London, 1991.

[11] Barbara J. Grosz and Candace L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12:175–204, 1986.

[12] Jeanette K. Gundel. *The Role of Topic and Comment in Linguistic Theory.* PhD thesis, University of Texas, 1974. Published by the Indiana University Linguistics Club, Bloomington, 1977.

[13] Jeanette K. Gundel. Universals of topic–comment structure. In Michael Hammond, Edith A. Moravcsik, and Jessica R. Wirth, editors, *Studies in Syntactic Typology*, volume 17 of *Typological Studies in Language*, pages 209–239. John Benjamins: Amsterdam, Philadelphia, 1988.

[14] Herman Hendriks and Paul Dekker. Links without location. In Paul Dekker and Martin Stokhof, editors, *Proceedings of the Tenth Amsterdam Colloquium*, pages 339–358. ILLC—Dept. of Philosophy, University of Amsterdam, 1996.

[15] Kai von Fintel. A minimal theory of adverbial quantification. In Hans Kamp and Barbara H. Partee, editors, *Proceedings of the workshops on Context Dependence in the Analysis of Linguistic Meaning*, volume 1: Papers, pages 153–193. IMS, University of Stuttgart, 1995.