

# **DESIGN OF CLUSTERED SUPERSCALAR MICROARCHITECTURES**

---

**Joan-Manuel Parcerisa**

Departament d'Arquitectura de Computadors  
Universitat Politècnica de Catalunya  
Barcelona (Spain), April 2004

A THESIS SUBMITTED IN FULFILLMENT  
OF THE REQUIREMENTS FOR THE DEGREE OF  
Doctor en Informàtica



# DESIGN OF CLUSTERED SUPERSCALAR MICROARCHITECTURES

---

**Joan-Manuel Parcerisa**

Departament d'Arquitectura de Computadors  
Universitat Politècnica de Catalunya  
Barcelona (Spain), April 2004

Advisor:  
**Antonio González**

A THESIS SUBMITTED IN FULFILLMENT  
OF THE REQUIREMENTS FOR THE DEGREE OF  
Doctor en Informàtica



# ABSTRACT

Over the past decade superscalar microprocessors have achieved enormous improvements in computing power by exploiting higher levels of parallelism in many different ways. High-performance superscalar processors have experienced remarkable increases in processor width, pipeline depth and speculative execution. All of these trends have come at an extremely high increase in hardware complexity and chip resource consumption. Until recently, their main limitation has been the availability of such resources in the chip, but with current technology shrinks and increases in transistor budgets, other limiting factors have become preeminent, such as power consumption, temperature and wire delays. These new problems greatly compromise the scalability of conventional superscalar designs.

Many previous works have demonstrated the effectiveness of partitioning the layout of several critical hardware components as a means to keep most of the parallelism while improving the scalability. Some of these components are the register file, the issue queue and the bypass network. Their partitioning is the basis for the so called clustered architectures. A clustered processor core, made up of several low complex blocks or clusters, can efficiently execute chains of dependent instructions without paying the overheads of a long issue, register read or bypass latencies. Of course, when two dependent instructions execute in different clusters, an inter-cluster communication penalty is incurred. Moreover, distributed structures usually imply lower dynamic power requirements, and they simplify power management via techniques such a selective clock/power gating and voltage scaling.

The purpose of this thesis is to study several key aspects of a clustered architecture with fully distributed components. The first target of this research is the distribution of instructions among clusters, since it plays a major role on performance. The main goals of a cluster assignment algorithm are to keep the workload balanced and to reduce critical communications among clusters. Several techniques are proposed to achieve these goals from two different perspectives: slice-based algorithms, that assign groups of dynamic instructions, and algorithms that operate on a per-instruction basis. The second contribution of this thesis proposes value prediction as a means to mitigate the penalties of inter-cluster communications while also improving workload balance. The third aspect considered is the cluster interconnect, which mainly determines communication latency. Several techniques are proposed to design it seeking for the best trade-off between cost and performance. Finally, the last contribution proposes techniques for clustering the main components of the processor front-end, i.e. those involved in branch prediction, instruction fetch, decoding, cluster assignment and renaming.



# AGRAÏMENTS

En primer lloc, gràcies a Antonio González, de qui he après tantes coses sobre l'ofici de la recerca, i sobre l'arquitectura de computadors al llarg d'aquests anys. Gràcies per la seva confiança, els seus savis consells i la seva tenaç iniciativa.

A en Ramon Canal, en Julio Sahuquillo, en José Duato i en Jim Smith, entusiastes col.laboradors amb qui he tingut fèrtils discussions tècniques i que han proporcionat significatives contribucions a aquesta tesi.

A Jordi Cortadella i Mateo Valero per confiar en mi per al primer projecte amb què vaig debutar al DAC, i que va suposar el meu primer contacte amb la recerca científica.

A tots els companys del departament, de qui he rebut sempre mostres de suport i amb els quals he tingut sempre converses estimulants. La llista és llarga, i segur que em deixo algú, però vull esmentar especialment a Pedro, Ramon, Josep-Llorenç i Fernando, amb qui he compartit les millors gresques i hem passat les millors estones. També a en Pepe, Suso, Josep-Ramon, Daniel, Enric, Cristina, Miguel, Agustín, Fermín, Josep, Luïma, Jordi, David, Llorenç, Chema i els antics companys de despatx Susana i Fernando. Gràcies també al personal tècnic de sistemes i de secretaria, que m'han ajudat sempre amb promptitud i eficàcia, i als organismes públics que han finançat aquest treball<sup>1</sup>.

A la Roser, per la seva paciència, suport i afecte, amb qui he compartit els moments dolços i els difícils d'aquesta tesi durant tots aquests anys, i amb qui espero compartir nous projectes durant molts anys més.

I finalment, vull agrair a tota la meva família, pares Josep i Manuela i germans Josep, Jordi i Ramon, que sempre m'han guiat i encoratjat, i junt amb els quals he après a conèixer el valor de l'esforç, el plaer de l'aprenentatge, i tantes altres virtuts sense les quals no hauria reeixit. Gràcies en especial al meu pare, que em va brindar l'oportunitat d'iniciar, ja tardanament, els estudis d'informàtica.

---

1. This work was supported by the Spanish Ministry of Education under contracts TIC-95-0429, TIC-98-0511 and TIC-2001-0995, and by the CEPBA (European Center for Parallelism of Barcelona)





# TABLE OF CONTENTS

<b>ABSTRACT</b>	<b>I</b>
<b>AGRAÏMENTS</b>	<b>III</b>
<b>TABLE OF CONTENTS</b>	<b>V</b>
<b>CHAPTER 1. INTRODUCTION</b>	<b>1</b>
1.1 Background and Motivation .....	1
1.1.1 The Performance of Superscalar Processors.....	1
1.1.2 Technology Scaling of Wire Delays .....	2
1.1.3 Power Consumption.....	3
1.1.4 Clock Speed .....	3
1.1.5 Complexity.....	4
1.1.6 The Clustered Approach .....	5
1.2 Related Work .....	6
1.2.1 The Two-Cluster Alpha 21264 .....	7
1.2.2 The Integer-Decoupled Architecture .....	7
1.2.3 The Dependence-Based and Architectures with Replicated Register Files.....	8
1.2.4 Cluster Assignment Schemes Based on a Trace Cache .....	11
1.2.5 The Multicluster and Architectures with Distributed Register Files .....	11
1.2.6 Other Related Works.....	13
1.3 Thesis Overview and Contributions.....	15
1.3.1 Cluster Assignment Algorithms.....	15
1.3.2 Reducing Wire Delay Penalties through Value Prediction.....	17
1.3.3 Efficient Interconnects for Clustered Microarchitectures.....	17
1.3.4 A Clustered Front-End for Superscalar Processors .....	18
1.3.5 Thesis Contributions .....	18
1.4 Document Organization .....	19
<b>CHAPTER 2. EXPERIMENTAL ENVIRONMENT</b>	<b>21</b>
2.1 Reference Clustered Microarchitecture .....	21
2.1.1 Superscalar Model .....	22

2.1.2	Clustered Model .....	22
2.1.3	Implementation Issues for Copy Instructions .....	25
2.2	Main Architectural Parameters .....	26
2.3	Simulation Methodology .....	26

### **CHAPTER 3. SLICE-BASED DYNAMIC CLUSTER ASSIGNMENT MECHANISMS 29**

3.1	Main goals of a Cluster Assignment Mechanism .....	30
3.1.1	Communication .....	30
3.1.2	Workload Balance .....	31
3.2	The Cost-Effective Clustered Microarchitecture .....	32
3.3	Cluster Assignment Schemes .....	33
3.3.1	Terminology .....	33
3.3.2	Experimental Framework .....	35
3.3.3	Static versus Dynamic Cluster Assignment with LdSt Slice Steering .....	36
3.3.4	LdSt Slice Steering versus Br Slice Steering .....	37
3.3.5	Non-Slice Balance Steering .....	38
3.3.6	Slice Balance Steering .....	41
3.3.7	Priority Slice Balance Steering .....	42
3.3.8	General Balance Steering .....	44
3.4	Evaluation .....	45
3.4.1	Overall Performance Comparison .....	45
3.4.2	Comparison to a Static Slice-Based Scheme .....	46
3.4.3	Comparison to Another Dynamic Scheme .....	46
3.4.4	Register Replication .....	47
3.4.5	Running FP Programs on a Cost-Effective Clustered Architecture .....	48
3.5	Conclusions .....	48

### **CHAPTER 4. INSTRUCTION-BASED DYNAMIC CLUSTER ASSIGNMENT MECHANISMS 51**

4.1	Communication and Workload Balance .....	52
4.1.1	Communication .....	52
4.1.2	Workload Balance .....	52
4.2	Instruction-Based Cluster Assignment Schemes .....	54
4.2.1	Experimental Framework .....	54
4.2.2	Modulo Steering .....	55
4.2.3	Simple RMB Steering .....	55
4.2.4	Balanced RMB Steering .....	55
4.2.5	Improving the Workload Balance with the Advanced RMB Steering .....	57
4.2.6	Optimizing the Critical Path with the Priority RMB Steering Scheme .....	59
4.2.7	Reducing Communications with Accurate-Rebalancing .....	60
4.3	Evaluation .....	61
4.3.1	RMB versus Slice-Based Steering Schemes .....	62
4.3.2	The AR-Priority RMB versus Other Instruction-Based Schemes .....	62

4.3.3	Sensitivity to the Communication Latency.....	64
4.3.4	Sensitivity to the Interconnect Bandwidth.....	65
4.3.5	Overall Evaluation of the RMB Steering Schemes.....	66
4.4	Conclusions.....	67

## **CHAPTER 5. REDUCING WIRE DELAY PENALTY THROUGH VALUE PREDICTION** **69**

5.1	Introduction.....	69
5.2	Microarchitecture.....	70
5.2.1	Value Prediction.....	70
5.2.2	Speculation on Remote Operands.....	71
5.2.3	The Baseline Steering Algorithm.....	71
5.2.4	Performance Evaluation.....	72
5.3	A Steering Scheme for Value Prediction.....	72
5.3.1	Enhancing the Partitioning through Value Prediction.....	72
5.3.2	The VPB Steering Scheme.....	74
5.4	Sensitivity Analysis.....	75
5.4.1	Communication Latency.....	76
5.4.2	Register Read Latency and Misprediction Penalty.....	77
5.4.3	Value Predictor Table Size.....	77
5.4.4	Experiments with the SpecInt95.....	78
5.5	Conclusions.....	78

## **CHAPTER 6. EFFICIENT INTERCONNECTS FOR CLUSTERED MICROARCHITECTURES** **81**

6.1	Introduction.....	81
6.2	Improved Steering Schemes.....	82
6.2.1	A Topology-Aware Steering.....	83
6.3	The Interconnection Network.....	83
6.3.1	Routing Algorithms.....	83
6.3.2	Register File Write Ports.....	83
6.3.3	Communication Timing.....	84
6.3.4	Transmission Time.....	84
6.3.5	Router Structures.....	85
6.3.6	Bus versus Point-to-Point Interconnects.....	86
6.4	Four-Cluster Network Topologies.....	87
6.4.1	Bus2.....	87
6.4.2	Synchronous Ring.....	88
6.4.3	Partially Asynchronous Ring.....	88
6.4.4	Ideal Ring.....	89
6.4.5	Ideal Crossbar.....	89
6.5	Eight-Cluster Network Topologies.....	89

6.5.1	Bus2 and Bus4 .....	89
6.5.2	Synchronous and Partially Asynchronous Rings .....	89
6.5.3	Mesh .....	90
6.5.4	Torus .....	90
6.5.5	Ideal Torus .....	91
6.5.6	Ideal Crossbar .....	91
6.6	Experimental Evaluation .....	92
6.6.1	Network Latency Analysis .....	92
6.6.2	Performance of Four-Cluster Interconnects .....	94
6.6.3	Queue Length .....	95
6.6.4	Performance of Eight-Cluster Interconnects .....	96
6.6.5	Effectiveness of the Accurate-Rebalancing and Topology-Aware Steering .....	97
6.6.6	Experiments with the SpecInt95 .....	98
6.7	Summary and Conclusions of this Chapter .....	99

## **CHAPTER 7. A CLUSTERED FRONT-END FOR SUPERSCALAR PROCESSORS 101**

7.1	Introduction .....	101
7.2	Clustering Front-End Subsystems .....	103
7.2.1	Clustering the Branch Predictor (Stage 1) .....	104
7.2.2	Clustering the I-Cache (stage 2) .....	106
7.2.3	Decode (stage 3) .....	107
7.2.4	Clustering the Steering Logic (stage 4) and Broadcast of Cluster Assignments (stage 5) .....	107
7.2.5	Clustering the Rename Logic (stage 6) .....	109
7.2.6	Dispatch (stage 7) .....	110
7.2.7	Back-End Timing and Commit .....	110
7.3	Microarchitecture Evaluation .....	111
7.3.1	Performance .....	111
7.3.2	Impact of Partitioning the Branch Predictor .....	112
7.3.3	Impact of Steering with Outdated Renaming Information .....	113
7.4	Conclusions .....	114

## **CHAPTER 8. CONCLUSIONS 117**

8.1	Conclusions .....	117
8.2	Open-Research Areas .....	120

## **BIBLIOGRAPHY 123**

## **LIST OF FIGURES 133**

## **LIST OF TABLES 137**