## MICROPROCESSOR ARCHITECTURE

This book gives a comprehensive description of the architecture of microprocessors from simple in-order short pipeline designs to out-of-order superscalars. It discusses topics such as

- The policies and mechanisms needed for out-of-order processing, such as register renaming, reservation stations, and reorder buffers
- Optimizations for high performance, such as branch predictors, instruction scheduling, and load–store speculations
- Design choices and enhancements to tolerate latency in the cache hierarchy of single and multiple processors
- State-of-the-art multithreading and multiprocessing, emphasizing single-chip implementations

Topics are presented as conceptual ideas, with metrics to assess the effects on performance, if appropriate, and examples of realization. The emphasis is on how things work at a black box and algorithmic level. The author also provides sufficient detail at the register transfer level so that readers can appreciate how design features enhance performance as well as complexity.

Jean-Loup Baer is Professor Emeritus of Computer Science and Engineering at the University of Washington, where he has been since 1969. Professor Baer is the author of *Computer Systems Architecture* and of more than 100 refereed papers. He is a Guggenheim Fellow, an ACM Fellow, and an IEEE Fellow. Baer has held several editorial positions, including editor-in-chief of the *Journal of VLSI and Computer Systems* and editor of the *IEEE Transactions on Computers*, the *IEEE Transactions on Parallel and Distributed Systems*, and the *Journal of Parallel and Distributed Computing*. He has served as General Chair and Program Chair of several conferences, including ISCA and HPCA.

# Microprocessor Architecture

## FROM SIMPLE PIPELINES TO CHIP MULTIPROCESSORS

**Jean-Loup Baer**
University of Washington, Seattle

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore,
São Paulo, Delhi, Dubai, Tokyo

Cambridge University Press
32 Avenue of the Americas, New York, NY 10013-2473, USA

www.cambridge.org
Information on this title: www.cambridge.org/9780521769921



First published 2010

Printed in the United States of America

*A catalog record for this publication is available from the British Library.*

*Library of Congress Cataloging in Publication data*

Baer, Jean-Loup.
Microprocessor architecture: from simple pipelines to chip multiprocessors /
Jean-Loup Baer.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-521-76992-1 (hardback)
1. Microprocessors. 2. Computer architecture. I. Title.
QA76.5.B227 2009
004.2′2 – dc22 2009025686

ISBN 978-0-521-76992-1 Hardback

*To Diane, Marc, Shawn, and Danielle*

# Contents

# Preface

Computer architecture is at a turning point. Radical changes occurred in the 1980s when the Reduced Instruction Set Computer (RISC) philosophy, spurred in good part by academic research, permeated the industry as a reaction to the Complex Instruction Set Computer (CISC) complexities. Today, three decades later, we have reached a point where physical limitations such as power dissipation and design complexity limit the speed and the performance of single-processor systems. The era of chip multiprocessors (CMP), or multicores, has arrived.

Besides the uncertainty on the structure of CMPs, it is not known yet whether the future CMPs will be composed of simple or complex processors or a combination of both and how the on-chip and off-chip memory hierarchy will be managed. It is important for computer scientists and engineers to look at the possible options for the modules that will compose the new generations of multicores. In this book, we describe the architecture of microprocessors from simple in-order short pipe designs to out-of-order superscalars with many optimizations. We also present choices and enhancements in the cache hierarchy of single processors. The last part of this book introduces readers to the state-of-the-art in multithreading and multiprocessing, emphasizing single-chip implementations and their cache hierarchy.

The emphasis in this book is on "how things work" at a black box and algorithmic level. It is not as close to the hardware as books that explain features at the register transfer level. However, it provides sufficient detail, say at the "bit level" and through pseudocode, so that the reader can appreciate design features that have or will enhance performance as well as their complexity.

As much as possible we present topics as conceptual ideas, define metrics to assess the performance impact if appropriate, show alternate (if any) ways of implementing them, and provide examples of realization.

## Synopsis

This book has three main parts. Chapter 1 and Chapter 2 review material that should have been taught in a prerequisite class. Chapters 3 through 6 describe single-processor systems and their memory hierarchy. Chapter 7 and Chapter 8 are

devoted to parallelism. The last (short) chapter, Chapter 9, introduces limitations due to technology and presents challenges for future CMPs.

More specifically:

- Chapter 1 reviews Moore's law and its influence as well as current limitations to ever faster processors. Performance metrics such as *cycles per instruction* (CPI), *instruction per cycle* (IPC), and speedup are defined. A section on performance evaluation introduces benchmarks and simulators. Chapter 2 summarizes the main concepts behind pipelining, including data and control hazards. Two versions of the "classical" 5-stage pipeline are contrasted. The basics of cache organization and performance are reviewed. Virtual memory (paging) and *Translation Look-Aside Buffers* (TLBs) are presented from the computer architecture (not the operating system) viewpoint.
- Chapter 3 describes two landmark "families" in the history of modern multiple-issue microprocessors: in-order processors represented by the DEC Alpha 21164 and out-of-order processors represented by the Intel Pentium P6 microarchitecure. In order to ease the description of the latter, a first look at register renaming is provided, with a sidebar showing the classical Tomasulo algorithm. This chapter ends with an introduction to the Very Long Instruction Word (VLIW) / Explicitly Parallel Instruction Computing (EPIC) philosophy and a brief overview of its most well-known industrial realization, the Intel Itanium. Chapter 4 and Chapter 5 are devoted to detailed explanations of, respectively, the front end and the back end of the pipeline. Because many advances in superscalar performance have been attained through speculation, we start Chapter 4 with a "model" of a branch predictor. This model will be referred to in the book when other speculative schemes are explored. We then describe actual branch predictors and branch target predictors. A sidebar describes the sophisticated Alpha 21264 tournament predictor. The remainder of the chapter deals with instruction fetching, including trace caches, and presents another view of register renaming that has become more popular than Tomasulo's. Chapter 5 looks at another potential bottleneck in the pipeline, namely the wakeup-select cycle. Load speculation and other back-end optimizations are introduced. Chapter 6 deals with the cache hierarchy: how to improve access times for first-level caches (another critical potential source of slowdown); methods to hide latency from higher-level caches and main memory such as prefetching and lock-up free caches; and, in a more research-oriented way, design and performance issues for large caches. We conclude this chapter with a look at main memory.
- Chapter 7 and Chapter 8 present multiprocessors and multithreading. In Chapter 7, we first introduce taxonomies that will set the stage for the two chapters. Cache coherence is then treated with both snoopy protocols and directory protocols, because it is likely that both types of protocols will be used in future parallel processing systems, whether on-chip or between chips. The next topic is synchronization, in which, in addition to the existing lock mechanisms, we mention approaches that might replace or complement them, such as transactional

memory. We also briefly introduce relaxed memory models, a topic that may become more important in the near future. The chapter concludes with a description of multimedia instruction set extensions because this is a (limited) form of parallelism. In Chapter 8, we start by looking at various flavors of multithreading: fine-grained, coarse-grained, and SMT. This leads us to the description of some current CMPs. We start with two examples of general-purpose CMPs: the Sun Niagara, a CMP using fine-grained multithreaded simple processors, and the Intel Core Duo, a representative of Intel multicores using complex superscalar processors. We conclude with two special-purpose CMPs: the IBM Cell, intended for games and scientific computation, and the Intel IXP, a network processor using coarse-grained multithreaded microengines.

- Finally, Chapter 9 gives a brief introduction to the factors that have capped the exponential performance curve of single processors, namely power issues, wire lengths, and pipeline depths. We end the chapter with an (incomplete) list of challenges for future CMPs.

## Use of the book

This book grew out of courses given over the last ten years at the senior undergraduate and first-year graduate level for computer science and computer engineering students at the University of Washington. It is intended as a book for a second course in computer architecture. In an undergraduate class, we typically spend 10% of the time on review and presentation of simulators. We spend 60% of the time on single processors. About 25% is spent on multiprocessors and multithreading, and the remainder on some research presentations by colleagues and advanced graduate students. At the graduate level, the choice of topics depends more on the instructor. For example, one possible approach would be to assign the first two chapters as required reading and start directly with topics of the instructor's choice. A rule of thumb that we have followed is to have a 50–50 split in classroom time between single-processor (Chapters 3 through 6) and CMPs (Chapters 7 through 9).

## Acknowledgments

I am indebted to students who took my classes and to many colleagues, including my doctoral students, for their help and comments. I want especially to recognize the leaders and participants of the weekly CSE "Computer Architecture Lunch," which Wen-Hann Wang and I created more than two decades ago. Their choice of the topics in this seminar and their constructive criticism of the importance of specific areas have influenced the contents of this book. Their patient explanations of what were for me obscure points has hopefully led to clearer exposition. In particular, I thank my faculty colleagues Luis Ceze, Carl Ebeling, Susan Eggers, Mark Oskin, and Larry Snyder and my research collaborators Craig Anderson, Tien-Fu Chen, Patrick Crowley, Dennis Lee, Douglas Low, Sang-Lyul Min, Xiaohan Qin, Taylor VanVleet, Wen-Hann Wang, Wayne Wong, and Rick Zucker.

I thank Lauren Cowles and David Jou at Cambridge University Press for providing me with excellent anonymous reviews, and Shana Meyer and colleagues at Aptara, Inc. – in particular, copy editor Joseph C. Fineman – for a smooth and timely production process.

Jean-Loup Baer
Seattle, WA
August 2009