In Praise of *Memory Systems: Cache, DRAM, Disk*

Memory Systems: Cache, DRAM, Disk *is the first book that takes on the whole hierarchy in a way that is consistent, covers the complete memory hierarchy, and treats each aspect in significant detail. This book will serve as a definitive reference manual for the expert designer, yet it is so complete that it can be read by a relative novice to the computer design space. While memory technologies improve in terms of density and performance, and new memory device technologies provide additional properties as design options, the principles and methodology presented in this amazingly complete treatise will remain useful for decades. I only wish that a book like this had been available when I started out more than three decades ago. It truly is a landmark publication. Kudos to the authors.*

—Al Davis, University of Utah

Memory Systems: Cache, DRAM, Disk *fills a huge void in the literature about modern computer architecture. The book starts by providing a high level overview and building a solid knowledge basis and then provides the details for a deep understanding of essentially all aspects of modern computer memory systems including architectural considerations that are put in perspective with cost, performance and power considerations. In addition, the historical background and politics leading to one or the other implementation are revealed. Overall, Jacob, Ng, and Wang have created one of the truly great technology books that turns reading about bits and bytes into an exciting journey towards understanding technology.*

—Michael Schuette, Ph.D., VP of Technology Development at OCZ Technology

This book is a critical resource for anyone wanting to know how DRAM, cache, and hard drives really work. It describes the implementation issues, timing constraints, and trade-offs involved in past, present, and future designs. The text is exceedingly well-written, beginning with high-level analysis and proceeding to incredible detail only for those who need it. It includes many graphs that give the reader both explanation and intuition. This will be an invaluable resource for graduate students wanting to study these areas, implementers, designers, and professors.

—Diana Franklin, California Polytechnic University, San Luis Obispo

Memory Systems: Cache, DRAM, Disk *fills an important gap in exploring modern disk technology with accuracy, lucidity, and authority. The details provided would only be known to a researcher who has also contributed in the development phase. I recommend this comprehensive book to engineers, graduate students, and researchers in the storage area, since details provided in computer architecture textbooks are woefully inadequate.*

—Alexander Thomasian, IEEE Fellow, New Jersey Institute of Technology and Thomasian and Associates

Memory Systems: Cache, DRAM, Disk *offers a valuable state of the art information in memory systems that can only be gained through years of working in advanced industry and research. It is about time that we have such a good reference in an important field for researchers, educators and engineers.*

—Nagi Mekhiel, Department of Electrical and Computer Engineering, Ryerson University, Toronto

This is the only book covering the important DRAM and disk technologies in detail. Clear, comprehensive, and authoritative, I have been waiting for such a book for long time.

—Yiming Hu, University of Cincinnati

Memory is often perceived as the performance bottleneck in computing architectures. Memory Systems: Cache, DRAM, Disk*, sheds light on the mystical area of memory system design with a no-nonsense approach to what matters and how it affects performance. From historical discussions to modern case study examples this book is certain to become as ubiquitous and used as the other Morgan Kaufmann classic textbooks in computer engineering including Hennessy and Patterson's* Computer Architecture: A Quantitative Approach.

—R. Jacob Baker, Micron Technology, Inc. and Boise State University.

Memory Systems: Cache, DRAM, Disk *is a remarkable book that fills a very large void. The book is remarkable in both its scope and depth. It ranges from high performance cache memories to disk systems. It spans circuit design to system architecture in a clear, cohesive manner. It is the memory architecture that defines modern computer systems, after all. Yet, memory systems are often considered as an appendage and are covered in a piecemeal fashion. This book recognizes that memory systems are the heart and soul of modern computer systems and takes a 'holistic' approach to describing and analyzing memory systems.*

The classic book on memory systems was written by Dick Matick of IBM over thirty years ago. So not only does this book fill a void, it is a long-standing void. It carries on the tradition of Dick Matick's book extremely well, and it will doubtless be the definitive reference for students and designers of memory systems for many years to come. Furthermore, it would be easy to build a top-notch memory systems course around this book. The authors clearly and succinctly describe the important issues in an easy-to-read manner. And the figures and graphs are really great—one of the best parts of the book.

When I work at home, I make coffee in a little stove-top espresso maker I got in Spain. It makes good coffee very efficiently, but if you put it on the stove and forget it's there, bad things happen—smoke, melted gasket—'burned coffee meltdown.' This only happens when I'm totally engrossed in a paper or article. Today, for the first time, it happened twice in a row—while I was reading the final version of this book.

—Jim Smith, University of Wisconsin—Madison

Memory Systems

Cache, DRAM, Disk

Memory Systems

Cache, DRAM, Disk

Bruce Jacob
University of Maryland at College Park

Spencer W. Ng
Hitachi Global Storage Technologies

David T. Wang
MetaRAM

With Contributions By

Samuel Rodriguez
Advanced Micro Devices

AMSTERDAM • BOSTON • HEIDELBERG LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann is an imprint of Elsevier

MORGAN KAUFMANN PUBLISHERS

Publisher	Denise E.M. Penrose
Acquisitions Editor	Chuck Glaser
Publishing Services Manager	George Morrison
Senior Production Editor	Paul Gottehrer
Developmental Editor	Nate McFadden
Assistant Editor	Kimberlee Honjo
Cover Design	Joanne Blank
Text Design	Dennis Schaefer
Composition	diacriTech
Interior printer	Maple-Vail Book Manufacturing Group
Cover printer	Phoenix Color

Morgan Kaufmann Publishers is an imprint of Elsevier.
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

This book is printed on acid-free paper.

Library of Congress Cataloging-in-Publication Data
Application submitted

ISBN: 978-0-12-379751-3

For information on all Morgan Kaufmann publications,
visit our Web site at *www.mkp.com* or *www.books.elsevier.com*

Printed in the United States of America
08 09 10 11 12 5 4 3 2 1

Dedication

Jacob *To my parents, Bruce and Ann Jacob, my wife, Dorinda, and my children, Garrett, Carolyn, and Nate*

Ng *Dedicated to the memory of my parents Ching-Sum and Yuk-Ching Ng*

Wang *Dedicated to my parents Tu-Sheng Wang and Hsin-Hsin Wang*

You can tell whether a person plays or not by the way he carries the instrument, whether it means something to him or not.

Then the way they talk and act. If they act too hip, you know they can't play [jack].

—Miles Davis

[...] in connection with musical continuity, Cowell remarked at the New School before a concert of works by Christian Wolff, Earle Brown, Morton Feldman, and myself, that here were four composers who were getting rid of glue. That is: Where people had felt the necessity to stick sounds together to make a continuity, we four felt the opposite necessity to get rid of the glue so that sounds would be themselves.

Christian Wolff was the first to do this. He wrote some pieces vertically on the page but recommended their being played horizontally left to right, as is conventional. Later he discovered other geometrical means for freeing his music of intentional continuity. Morton Feldman divided pitches into three areas, high, middle, and low, and established a time unit. Writing on graph paper, he simply inscribed numbers of tones to be played at any time within specified periods of time.

There are people who say, "If music's that easy to write, I could do it." Of course they could, but they don't. I find Feldman's own statement more affirmative. We were driving back from some place in New England where a concert had been given. He is a large man and falls asleep easily. Out of a sound sleep, he awoke to say, "Now that things are so simple, there's so much to do." And then he went back to sleep.

—John Cage, *Silence*

Contents

Preface

"It's the Memory, Stupid!"

If you develop an ear for sounds that are musical it is like developing an ego. You begin to refuse sounds that are not musical and that way cut yourself off from a good deal of experience.

—John Cage

In 1996, Richard Sites, one of the fathers of computer architecture and lead designers of the DEC Alpha, had the following to say about the future of computer architecture research:

> *Across the industry, today's chips are largely able to execute code faster than we can feed them with instructions and data. There are no longer performance bottlenecks in the floating-point multiplier or in having only a single integer unit. The real design action is in memory subsystems—caches, buses, bandwidth, and latency.*
>
> *An anecdote: in a recent database benchmark study using TPC-C, both 200-MHz Pentium Pro and 400MHz 21164 Alpha systems were measured at 4.2–4.5 CPU cycles per instruction retired. In other words, three out of every four CPU cycles retired zero instructions: most were spent waiting for memory. Processor speed has seriously outstripped memory speed.*
>
> *Increasing the width of instruction issue and increasing the number of simultaneous instruction streams only makes the memory bottleneck worse. If a CPU chip today needs to move 2 GBytes/s (say, 16 bytes every 8 ns) across the pins to keep itself busy, imagine a chip in the foreseeable future with twice the clock rate, twice the issue width, and two instruction streams. All these factors multiply together to require about 16 GBytes/s of pin bandwidth to keep this chip busy. It is not clear whether pin bandwidth can keep up—32 bytes every 2ns?*
>
> *I expect that over the coming decade memory subsystems design will be the* ***only*** *important design issue for microprocessors.* [Sites 1996, emphasis Sites']

The title of Sites' article is "It's the Memory, Stupid!" Sites realized in 1996 what we as a community are only now, more than a decade later, beginning to digest and internalize fully: *uh, guys, it really* ***is*** *the memory system ... little else matters right now, so stop wasting time and resources on other facets of the design.* Most of his colleagues designing next-generation Alpha architectures at Digital Equipment Corp. ignored his advice and instead remained focused on building ever faster microprocessors, rather than shifting their focus to the building of ever faster *systems*. It is perhaps worth noting that Digital Equipment Corp. no longer exists.

The increasing gap between processor and memory speeds has rendered the organization, architecture, and design of memory subsystems an increasingly important part of computer-systems design. Today, the divide is so severe we are now in one of those down-cycles where the processor is so good at

number-crunching it has completely sidelined itself; it is too fast for its own good, in a sense. Sites' prediction came true: memory subsystems design is now and has been for several years the *only* important design issue for microprocessors and systems. Memory-hierarchy parameters affect system performance *significantly* more than processor parameters (e.g., they are responsible for 2–10× changes in execution time, as opposed to 2–10%), making it absolutely essential for any designer of computer systems to exhibit an in-depth knowledge of the memory system's organization, its operation, its not-so-obvious behavior, and its range of performance characteristics. This is true now, and it is likely to remain true in the near future.

Thus this book, which is intended to provide exactly that type of in-depth coverage over a wide range of topics.

Topics Covered

In the following chapters we address the logical design and operation, the physical design and operation, the performance characteristics (i.e., design trade-offs), and, to a limited extent, the energy consumption of modern memory hierarchies.

In the cache section, we present topics and perspectives that will be new (or at least interesting) to even veterans in the field. What this implies is that the cache section is *not* an overview of processor-cache organization and its effect on performance—instead, we build up the concept of cache from first principles and discuss topics that are incompletely covered in the computer-engineering literature. The section discusses a significant degree of historical development in cache-management techniques, the physical design of modern SRAM structures, the operating system's role in cache coherence, and the continuum of cache architectures from those that are fully transparent (to application software and/or the operating system) to those that are fully visible.

DRAM and disk are interesting technologies because, unlike caches, they are not typically integrated onto the microprocessor die. Thus any discussion of these topics necessarily deals with the issue of communication: e.g., channels, signalling, protocols, and request scheduling.

DRAM involves one or more chip-to-chip crossings, and so signalling and signal integrity are as fundamental as circuit design to the technology. In the DRAM section, we present an intuitive understanding of exactly what happens inside the DRAM so that the ubiquitous parameters of the interface (e.g., t_{RC}, t_{RCD}, t_{CAS}, etc.) will make sense. We survey the various DRAM architectures that have appeared over the years and give an in-depth description of the technologies in the next generation memory-system architecture. We discuss memory-controller issues and investigate performance issues of modern systems.

The disk section builds from the bottom up, providing a view of the disk from physical recording principles to the configuration and operation of disks within system settings. We discuss the operation of the disk's read/write heads; the arrangement of recording media within the enclosure; and the organization-level view of blocks, sectors, tracks, and cylinders, as well as various protocols used to encode data. We discuss performance issues and techniques used to improve performance, including caching and buffering, prefetching, request scheduling, and data reorganization. We discuss the various disk interfaces available today (e.g., ATA, serial ATA, SCSI, fibre channel, etc.) as well as system configurations such as RAID, SAN, and NAS.

The last section of the book, *Cross-Cutting Issues,* covers topics that apply to all levels of the memory hierarchy, such as the tools of analysis and how to use them correctly, subthreshold leakage power in CMOS devices and circuits, a look at power breakdowns in future SRAMs, codes for error detection and error correction, the design and operation of virtual memory systems, and the hardware mechanisms that are required in microprocessors to support virtual memory.

Goals and Audience

The primary goal of this book is to bring the reader to a level of understanding at which the physical design and/or detailed software emulation of the entire hierarchy is possible, from cache to disk. As we argue in the initial chapter, this level of understanding

is important now and will become increasingly necessary over time. Another goal of the book is to discuss techniques of analysis, so that the next generation of design engineers is prepared to tackle the nontrivial multidimensional optimization problems that result from considering detailed side-effects that can manifest themselves at any point in the entire hierarchy.

Accordingly, our target audience are those planning to build and/or optimize memory systems: i.e., computer-engineering and computer-science faculty and graduate students (and perhaps advanced undergraduates) and developers in the computer design, peripheral design, and embedded systems industries.

As an educational textbook, this is targeted at graduate and undergraduate students with a solid background in computer organization and architecture. It could serve to support an advanced senior-level undergraduate course or a second-year graduate course specializing in computer-systems design. There is clearly far too much material here for any single course; the book provides depth on enough topics to support two to three separate courses. For example, at the University of Maryland we use the DRAM section to teach a graduate class called *High-Speed Memory Systems,* and we supplement both our general and advanced architecture classes with material from the sections on *Caches* and *Cross-Cutting Issues*. The *Disk* section could support a class focused solely on disks, and it is also possible to create for advanced students a survey class that lightly touches on all the topics in the book.

As a reference, this book is targeted toward both academics and professionals alike. It provides the breadth necessary to understand the wide scope of behaviors that appear in modern memory systems, and most of the topics are addressed in enough depth that a reader should be able to build (or at least model in significant detail) caches, DRAMs, disks, their controllers, their subsystems ... and understand their interactions.

What this means is that the book should not only be useful to developers, but it should also be useful to those responsible for long-range planning and forecasting for future product developments and their issues.

Acknowledgments and Thanks

Beyond the kind folks named in the book's Dedication, we have a lot of people to thank. The following people were enormously helpful in creating the contents of this book:

- Prof. Rajeev Barua, ECE Maryland, provided text to explain scratch-pad memories and their accompanying partitioning problem (see Chapter 3).
- Dr. Brinda Ganesh, a former graduate student in Bruce Jacob's research group, now at Intel DAP, wrote the latency-oriented section of the DRAM-performance chapter (see Section 15.5).
- Joseph Gross, a graduate student in Bruce Jacob's research group, updated the numbers in David Wang's Ph.D. proposal to produce the tables comparing the characteristics of modern DRAMs in the book's *Overview* chapter.
- Dr. Ed Growchowski, formerly of Hitachi Global Storage Technologies and currently a consultant for IDEMA, provided some of the disk recording density history charts.
- Dr. Martin Hassner, Dr. Bruce Wilson, Dr. Matt White, Chuck Cox (now with IBM), Frank Chu, and Tony Dunn of Hitatchi Global Storage Technologies provided important consultation on various details of disk drive design.
- Dr. Windsor Hsu, formerly of IBM Almaden Research Center and now with Data Domain, Inc., provided important consultation on system issues related to disk drives.
- Dr. Aamer Jaleel, a former graduate student in Bruce Jacob's research group, now at Intel VSSAD, is responsible for Chapter 4's sections on cache coherence.
- Michael Martin, a graduate student in Bruce Jacob's research group, executed simulations for the final table (Ov. 5) and formatted the four large time-sequence graphs in the *Overview* chapter.
- Rami Nasr, a graduate student at Maryland who wrote his M.S. thesis on the Fully

Buffered DIMM architecture, provided much of the contents of Chapter 14.

- Prof. Marty Peckerar, ECE Maryland, provided brilliant text and illustrations to explain the subthreshold leakage problem (see the book's *Overview* and Section 29.2).
- Profs. Yan Solihin, ECE NCSU, and Donald Yeung, ECE Maryland, provided much of the material on hardware and software prefetching (Sections 3.1.2 and 3.2.2); this came from a book chapter on the topic written by the pair for Kaeli and Yew's *Speculative Execution in High Performance Computer Architectures* (CRC Press, 2005).
- Sadagopan Srinivasan, a graduate student in Bruce Jacob's research group, performed the study at the end of Chapter 1 and provided much insight on the memory behavior of both streaming applications and multi-core systems.
- Dr. Nuengwong Tuaycharoen, a former graduate student in Bruce Jacob's research group, now at Thailand's Dhurakijpundit University, performed the experiments, and wrote the example holistic analysis at the end of the book's *Overview* chapter, directly relating to the behavior of caches, DRAMs, and disks in a single study.
- Patricia Wadkins and others in the Rochester SITLab of Hitachi Global Storage Technologies provided test results and measurement data for the book's Section III on *Disks*.
- Mr. Yong Wang, a signal integrity expert, formerly of HP and Intel, now with MetaRAM, contributed extensively to the Chapter on *DRAM System Signaling and Timing* (Chapter 9).
- Michael Xu at Hitachi Global Storage Technologies drew the beautiful, complex illustrations, as well as provided some of the photographs, in the book's Section III on Disks.

In addition, several students involved in tool support over the years deserve special recognition:

- Brinda Ganesh took the reins from David Wang to maintain *DRAMsim*; among other things, she is largely responsible for the FB-DIMM support in that tool.
- Joseph Gross also supports *DRAMsim* and is leading the development of the second generation version of the software, which is object-oriented and significantly streamlined.
- Nuengwong Tuaycharoen integrated the various disparate software modules to produce *SYSim*, a full-system simulator that gave us the wonderful time-sequence graphs at the end of the *Overview*.

Numerous reviewers spent considerable time reading early drafts of the manuscript and providing excellent critiques or our direction, approach, and raw content. The following reviewers directly contributed to the book in this way: Ashraf Aboulnaga, *University of Waterloo*; Al Davis, *University of Utah*; Diana Franklin, *California Polytechnic University, San Luis Obispo*; Yiming Hu, *University of Cincinnati*; David Kaeli, *Northeastern University*; Nagi Mekhiel, *Ryerson University, Toronto*; Michael Schuette, Ph.D., VP of Technology Development at *OCZ Technology*; Jim Smith, *University of Wisconsin—Madison*; Yan Solin, *North Carolina State University*; and Several Anonymous reviewers (i.e., we the authors were told not to use their names).

The editorial and production staff at Morgan Kaufmann/Elsevier was amazing to work with: Denise Penrose pitched the idea to us in the first place (and enormous thanks go to Jim Smith, who pointed her in our direction) and Nate McFadden and Paul Gottehrer made the process of writing, editing, and proofing go incredibly smoothly.

Lastly, Dr. Richard Matick, the author of the original memory-systems book (*Computer Storage Systems and Technology,* John Wiley & Sons, 1976) and currently a leading researcher in embedded DRAM at IBM T. J. Watson, provided enormous help in direction, focus, and approach.

Dave, Spencer, Sam, and I are indebted to all of these writers, illustrators, coders, editors, and reviewers alike; they all helped to make this book what it is. To those contributors: thank you. You rock.

Bruce Jacob, Summer 2007
College Park, Maryland