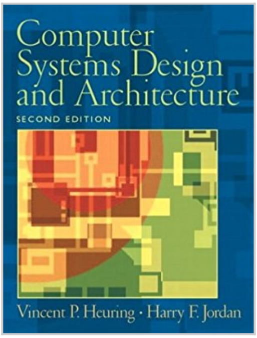
Computer Systems Design and Architecture
SECOND EDITION
Vincent P. Heuring · Harry F. Jordan

Computer Systems Design and Architecture (2nd Edition)

By Vincent P. Heuring, Harry F. Jordan

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan

The authors take a “No Mysteries” approach to computer systems. They interrelate three different viewpoints to provide a unique understanding of the subject: the perspective of the logic designer, the assembly language programmer, and the computer architect. The book has up-to-the-minute coverage of the latest developments in microprocessors, including ALU, pipelining, memory hierarchy, networks and the Internet. And, rather than focusing on a single type of architecture, Heuring and Jordan examine both CISC and RISC models at the ISA level using the unambiguous language of RTN (Register Transfer Notation), allowing for a more in-depth appreciation of different machine structures and functions.   The authors examine general purpose machine, machine languages, and digital logic, some real machines, processor design, processor design—advanced topics, computer arithmetic and the arithmetic unit, memory system design, input and output, peripheral devices, communications, networking and the Internet.   For computer science and engineering professionals.

Computer Systems Design and Architecture (2nd Edition)

By Vincent P. Heuring, Harry F. Jordan

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan

The authors take a “No Mysteries” approach to computer systems. They interrelate three different viewpoints to provide a unique understanding of the subject: the perspective of the logic designer, the assembly language programmer, and the computer architect. The book has up-to-the-minute coverage of the latest developments in microprocessors, including ALU, pipelining, memory hierarchy, networks and the Internet. And, rather than focusing on a single type of architecture, Heuring and Jordan examine both CISC and RISC models at the ISA level using the unambiguous language of RTN (Register Transfer Notation), allowing for a more in-depth appreciation of different machine structures and functions.   The authors examine general purpose machine, machine languages, and digital logic, some real machines, processor design, processor design—advanced topics, computer arithmetic and the arithmetic unit, memory system design, input and output, peripheral devices, communications, networking and the Internet.   For computer science and engineering professionals.

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan Bibliography

- Sales Rank: #1015044 in Books
- Published on: 2003-12-06
- Original language: English
- Number of items: 1
- Dimensions: 9.20" h x 1.30" w x 8.00" l, 2.50 pounds
- Binding: Paperback
- 608 pages

Editorial Review

From the Inside Flap
Computer Systems Design and Architecture

Preface

The computer ushers us into the information age. Born a mere fifty years ago, it now exerts a profound influence on almost every facet of our lives. What is the nature of this machine, this beast? How does it work, inside? How is it programmed internally? What are the nature of its connections to the outside world? These are the questions that this book will help you answer, and we hope that when you have mastered it you will be left with no mysteries about how computers work. We feel that one of the best ways to learn how a computer works is to design one, so throughout most of the book we have taken the perspective of the designer rather than that of the observer or the critic.

Computers are arguably the most complex systems humankind has ever built, and like all complex systems they can be appreciated from many viewpoints. A building can be examined for its overall architectural design, and the way that design affects its overall functioning. It can also be examined from the different viewpoint of how the size and shape of its rooms and halls relate to the design of its heating and air conditioning systems. Likewise the computer can be examined from the viewpoint of its overall structure and functioning, referred to as its architecture. It can be examined from the different viewpoint of one who is programming it in its machine, or assembly language. And it can be examined from the viewpoint of its lowest abstract logical structure-its design at the logic gate level.

All of these viewpoints are interrelated, and therefore important for mastery of the subject; thus in this book we adopt all three: the viewpoint of the computer architect, the viewpoint of the assembly language programmer, and the viewpoint of the logic designer. We believe that the synthesis of these three views will give you a depth and richness of understanding of the subject that will serve you well, whether your main interest be in computer design, computer science, or computer programming.

For more information on what this book covers and how it covers it, read on:

TO THE STUDENT We assume that you have had experience with computers as an end-user, and that you have written programs in some high level language such as Pascal, C, or Fortran. We also assume that you have had exposure to digital logic circuits. A knowledge of logic circuits is necessary for the understanding of the material in this book. For those who have not had such exposure, or who are uncertain about whether their background in this area is adequate, we have included an appendix, Appendix A: Digital Logic Circuits, which should provide you with sufficient background for understanding this text.

The focus of this book is on the design of computer systems--design at the gate level, design at the instruction-set-architecture level, and design at the computer system level. We believe that this integrated approach provides a depth and richness of understanding of the computer system that will serve you well, whether your main interest is as a Computer Engineer or as a Computer Scientist.

TO THE INSTRUCTOR This book is suitable for an introductory course on computer design at the junior, senior, or introductory graduate level. We assume that that the student has had at least an introductory course in some higher-level programming language such as C or Pascal, and a semester of logic design. However there is a comprehensive Appendix on Digital Logic Design, written by Professor Miles Murdocca, Rutgers University, which provides sufficient background material that the course can be taught to students without previous digital design experience. Appropriate topics for such a book have changed considerably in recent years, as desktop computers have evolved from simple, stand-alone units into complex systems attached to high-speed networks and internetworks. Earlier generations of microprocessors had almost trivial internal structure. Present designs contain multiple pipelined functional units with support for multiple processors and memories. Areas of computer design and architecture that were barely touched upon in the not-so-distant past have become major topics for discussion. Introductory compiler courses now routinely discuss optimization for pipelined processors. Users worry about whether they should add level 2 cache memory to their PCs. Support personnel wearily try to explain to the computer user how to configure the subnet mask for their network slip connection.

The topics of pipelined processor design, the memory hierarchy, and networks and internetworking are moving to center stage in the arena of computer design and architecture. Therefore we devoted the major parts of three chapters to treatment of these subjects. Given the focus on computer design and computer architecture, we approach the study of the computer from three viewpoints: the view of the assembly/machine language programmer, the view of the logic designer, and the view of the system architect. In covering the topic of gate-level computer design, we follow a model architecture through the design process, from the instruction set design level to the processor design level. Given the choice of using either a commercial machine with all of the complicating features that are necessary to make such a machine commercially successful, or using a model design that introduces sufficient practical features to make it both interesting and relevant to the subject, we chose the latter. The model machine, SRC, for Simple RISC Computer, is a 32-bit machine with an instruction set architecture that is similar to, but simpler than the current generation of RISCs. We adopt the view that it is best to use a formal description language in describing machine structure and function. There are many languages from which to choose. We selected a variant of the ISP language, RTN. Many other choices could have been made, but most of the languages used by practitioners are aimed more at hardware description and less at machine behavior and function. RTN is simple, easy to learn, and is at the appropriate description level.

Using this book with your curriculum and syllabus

There are probably almost as many different curricula and syllabi for computer design and architecture as there are schools and instructors in which the subject is taught. As you evaluate the material in this book you may discover:

Some topics may have been covered in a prerequisite course, and so can be omitted from discussion. Some topics need not be covered because they are covered in a subsequent course or courses. Some topics will be omitted or emphasized because of the aims of the course or the instructor. For example, computer science curricula may stress architectural topics and de-emphasize the gate-level design material, whereas computer engineering curricula would stress gate-level design. Some instructors will choose to emphasize or de-emphasize certain topics or change the topic ordering because of their particular philosophy of how the material should be presented.

We have tried to make the topic coverage sufficiently inclusive and self-contained so that the book will be adaptable to a wide range of curricula and syllabi. The first two chapters set the stage for the rest of the book, and should be covered first. Chapter 3 treats real machine designs and is essential for any student without a sold background in CISC and RISC instruction sets. Chapters 4 and 5 pursue gate-level design of the CPU. Chapters 6-10 are virtually stand-alone, and can be selected to suit the needs of your particular curriculum and syllabus.

Chapter descriptions

Chapter 1 takes just a peek at the machine from all three levels, and provides an overview on the subject.

Chapter 2 begins with a discussion of the relationship between machines and machine languages--how the nature of the instruction set influences the rest of the CPU structure. At this point in the text we introduce a language for formally describing machine structure and function. That is, we provide a way to unambiguously describe both the machine hardware and how instructions run on it. That language, RTN, is a simple one, and it is at the "just right" level to describe the running of instructions on hardware. We introduce RTN by using it to describe a generic model machine, SRC, that has a 32-bit ISA similar to the current crop of 32-bit commercial machines, but without many of the complicating factors. The chapter concludes by switching to the logic circuit level and putting a computer design spin on conventional logic circuit design by a brief discussion of how RTN operations translate into logic designs.

The goal of Chapter 3 is to provide the student with a concrete understanding of the CISC-RISC difference. It discusses two commercial machines at the ISA (and RTN) level, the CISC Motorola MC68000, and the RISC SPARC The chapter provides familiarity with several different machine architectures at the ISA level, so as to help develop an appreciation for two philosophically different approaches to machine design. We also introduce some practical issues such as upward compatibility and the way they influence machine design.

Chapter 4 is in many ways the keystone chapter of the book. It describes the interface between the instruction set and the hardware that it runs on, at the gate level. This description unfolds by developing a 1-bus design for the SRC introduced in Chapter 2, with RTN descriptions of machine functioning as a guide. Beginning with the ISA registers, the RTN description adds additional registers that are hidden at the ISA level as it treats how instructions are fetched decoded and executed, again at gate level. The discussion proceeds with design of the "soul of the machine," the control unit, and its heartbeat, the system clocking and timing. This chapter also begins the discussion of how hardware design influences instruction execution speed by examining alternative 2- and 3- bus designs. Two other aspects of processor design are covered in this chapter: the hardware reset, and the exception process. Our goal with Chapters 4 was "no mysteries." At the end of the chapter the student should understand exactly how the central processing unit works at the gate level.

Chapter 5 covers pipelining of the CPU, multiple-instruction-issue machines, and microcoded control unit design. Nearly every current and planned processor design employs pipelining in its CPU, and a thorough understanding of how pipelining works is a necessity for everyone from compiler writers to machine programmers and architects. We first present an overview of the important issues in pipelining and then show the design process by way of a pipelined design for the SRC. We then present a short discussion of instruction-level parallelism. We treat superscalar operation, where there are multiple functional units in the CPU that are capable of parallel operation, and VLIW, very long instruction word machines whose instruction words contain a number of processing steps that are executed in parallel. The chapter concludes with a discussion of microcoding. Microcoding is not used much in general purpose microprocessor chips at present, but in addition to its use in special purpose designs it is important in its own right, and it presents an interesting perspective on computer design.

Chapter 6 covers the design of the arithmetic and logic part of the computer. The design of this very important CPU component has a major impact on overall system performance. Since much of the ALU¹s performance is based on the underlying algorithms that are implemented in the hardware design, the discussion proceeds from data type, integer, for example, to algorithm, addition, for example, to the hardware implementation, a carry lookahead fast adder, for example. Both integer and floating point data types are covered in the chapter. There follows a section on how branch instructions use the ALU and a discussion of logical operations and overall ALU design. The chapter concludes with a discussion of floating point arithmetic.

Chapter 7 discusses the design of the memory hierarchy in detail. Beginning with the simplest 1-bit RAM and ROM cells, the chapter builds those cells into chips, chips into boards, and boards into modules. There is a discussion of the general nature of the relationship between two adjacent levels in the hierarchy, and following that, a discussion of cache design and the interaction between cache and main memory. This is followed by a discussion of virtual memory, the process of allowing the memory space to spill out from main memory onto the disk. The chapter concludes with a discussion of memory as a system.

Chapter 8 discusses the details of the machine's IO system. The chapter begins with a treatment of several kinds of buses, treating both bus signals and bus timing, and proceeds with a discussion of the two principal kinds of generic IO interfaces, serial and parallel. The chapter then covers the relationship between the machine interrupt structure and the IO system, and between these two and DMA, direct memory access, by which an external device can access main memory without CPU intervention.

Chapter 9 covers what's on the other end of the IO system: peripheral devices. It treats the structure and functioning of disk drives, video and other interactive display devices, printers, mice, and the interfaces to analog devices. The emphasis will be on how these devices actually work, and the nature of the interface between them, the CPU, and the outside world. Peripheral device performance is covered as a main aspect of interest.

Chapter 10 concludes the book with a discussion of computer-to-computer communications. In the present era no treatment of computer systems design and architecture is complete without a fairly in-depth discussion of computer communications and networking. We begin with a discussion of network structure and topology. Following that we present three examples of contemporary machine communications. The first example is the RS-232 serial data communications protocol that allows point-to-point communications between two computers or between a computer and a terminal. The second example is the Ethernet local area network, LAN for short. We discuss the Ethernet communications protocol at both the physical level and the data link layer, including the Ethernet packet structure. The third example of a communications system is the Internet--probably the largest and most important computer communications system on the planet. We discuss the TCP/IP Internet protocol, and Internet addresses and addressing. The chapter, and the book conclude with a very brief discussion of Internet applications and Internet futures.

Instructional support materials

A printed solutions manual that includes a set of electronic transparencies, and a collection of software support tools are available to adopters from Addison-Wesley. The set of 600 electronic transparencies is available in both Adobe Acrobat and Microsoft PowerPoint formats. The transparencies include the book's main points, which are presented in a lecture outline format, and all figures and tables from the book. Using the free Acrobat Reader, the transparencies in the Acrobat format can be viewed and printed in various ways from PC, Mac, and UNIX platforms. Instructors who have access to PowerPoint can modify the electronic transparencies in the PowerPoint format. A growing collection of software support tools--including an SRC assembler and simulator written in ANSI C--are available to adopters.

Errors

In spite of the good efforts of the authors, editors, reviewers, and class testers, this book undoubtedly contains errors. Errors may be reported by email to: csdabugs@colorado.edu, or by mail to: Addison Wesley Longman Electrical and Computer Engineering Editorial Department 2725 Sand Hill Road, Menlo Park, CA

94025.

Acknowledgments

We first wish to thank our Editor, Tim Cox, for seeing our vision, and helping us realize it; our Assistant Editor, Laura Cheu, "she who gets things done" and our Executive Editor, Dan Joraanstad, the "eye in the sky." We also wish to single out for special acknowledgement our Developmental Editor, Laurie McGuire, who did an excellent job of improving both the syntax and the semantics of this book. Deserving special thanks is our secretary, Maejean Ruehlman, who was invaluable helping us get the manuscript out the door. Thanks also to the entire Addison Wesley Longman team for their enthusiasm and commitment to this book, especially Lisa Weber, Ari Davidow, Kelly Murphy, Kathleen Cameron, and Susan Slater.

We wish to also thank our reviewers, who were most helpful in suggesting changes to the original manuscript of this book: George Adams (School of Electrical and Computer Engineering, Purdue University), Dr. Hassan Barada (Computer Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia), Thomas L. Casavant (Electrical and Computer Engineering, University of Iowa), Dr. Peter Drexel (Department of Computer Science, Plymouth State College), Michael A. Driscoll (Department of Electrical Engineering, Portland State University), J. Kelly Flanagan (Computer Science Department, Brigham Young University), William Tsun-yuk Hsu (Computer Science Department, San Francisco State University), Mariusz Jankowski (Department of Engineering, University of Southern Maine), Mike Keenan (Computer Science Department, Virginia Tech), Professor Steven P. Levitan (Department of Electrical Engineering, University of Pittsburgh), Jyh-Charn (Steve) Liu (Computer Science Department, Texas A&M University), Norman Matloff (Department of Computer Science, University of California at Davis), Walid Najjar (Computer Science Department, Colorado State University), Bernhard Weinberg (Computer Science, Michigan State University), D. Scott Wills (Electrical and Computer Engineering, Georgia Institute of Technology), Craig M. Wittenbrink (Board of Computer Engineering, University of California, Santa Cruz), David Wood (Computer Science Department, University of Wisconsin). Also to be thanked are the faculty and students who used the Fall 1995 and Spring 1996 Class Test Editions for the many helpful suggestions and error reports: Douglas J. Fouts (Department of Electrical and Computer Engineering, U.S. Naval Postgraduate School), John F. Passafiume (Department of Computer Science, Clemson University). Finally we wish to thank Professor Miles Murdocca of Rutgers University for contributing Appendix A, Digital Logic.

Vincent P. Heuring and Harry F. Jordan Boulder, Colorado April 23, 1996 heuring@colorado.edu harry@colorado.edu.

From the Back Cover

The authors take a “No Mysteries” approach to computer systems. They interrelate three different viewpoints to provide a unique understanding of the subject: the perspective of the logic designer, the assembly language programmer, and the computer architect. The book has up-to-the-minute coverage of the latest developments in microprocessors, including ALU, pipelining, memory hierarchy, networks and the Internet. And, rather than focusing on a single type of architecture, Heuring and Jordan examine both CISC and RISC models at the ISA level using the unambiguous language of RTN (Register Transfer Notation), allowing for a more in-depth appreciation of different machine structures and functions.The authors examine general purpose machine, machine languages, and digital logic, some real machines, processor design, processor design—advanced topics, computer arithmetic and the arithmetic unit, memory system design, input and output, peripheral devices, communications, networking and the Internet.For computer science and engineering professionals.

About the Author

Harry Jordan is Professor in the Departments of Electrical and Computer Engineering and of Computer Science at the University of Colorado at Boulder. His interests in computer systems center on the interface between hardware and software. He has worked on the application and performance of multiple instruction stream computers, emphasizing the architecture and performance of such machines to the structure of parallel programming languages and algorithms. He developed one of the earliest high-level language extensions for parallel processing.

His most recent project was to build and operate an optical, stored program, digital computer. He developed designs using optical fiber for both data storage and interconnection, and successfully ran a prototype.

Users Review

From reader reviews:

Ivory Hughes:

Do you have favorite book? In case you have, what is your favorite's book? Reserve is very important thing for us to be aware of everything in the world. Each e-book has different aim or maybe goal; it means that reserve has different type. Some people truly feel enjoy to spend their the perfect time to read a book. These are reading whatever they take because their hobby is reading a book. Consider the person who don't like looking at a book? Sometime, man or woman feel need book once they found difficult problem or exercise. Well, probably you will need this Computer Systems Design and Architecture (2nd Edition).

Priscilla McNeil:

Do you have something that you want such as book? The book lovers usually prefer to decide on book like comic, quick story and the biggest an example may be novel. Now, why not trying Computer Systems Design and Architecture (2nd Edition) that give your enjoyment preference will be satisfied simply by reading this book. Reading practice all over the world can be said as the opportinity for people to know world considerably better then how they react when it comes to the world. It can't be claimed constantly that reading practice only for the geeky particular person but for all of you who wants to end up being success person. So , for all of you who want to start reading as your good habit, you are able to pick Computer Systems Design and Architecture (2nd Edition) become your own starter.

Bradford Bryant:

Beside this particular Computer Systems Design and Architecture (2nd Edition) in your phone, it might give you a way to get nearer to the new knowledge or details. The information and the knowledge you will got here is fresh from oven so don't be worry if you feel like an old people live in narrow community. It is good thing to have Computer Systems Design and Architecture (2nd Edition) because this book offers to your account readable information. Do you oftentimes have book but you do not get what it's exactly about. Oh come on, that will not end up to happen if you have this within your hand. The Enjoyable arrangement here cannot be questionable, just like treasuring beautiful island. Use you still want to miss this? Find this book along with read it from right now!

Pamela Stanley:

A lot of book has printed but it is unique. You can get it by internet on social media. You can choose the most beneficial book for you, science, witty, novel, or whatever through searching from it. It is called of book Computer Systems Design and Architecture (2nd Edition). Contain your knowledge by it. Without leaving behind the printed book, it may add your knowledge and make you happier to read. It is most critical that, you must aware about reserve. It can bring you from one destination to other place.

Download and Read Online Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan #7LQIYT40E5R

Read Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan for online ebook

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan books to read online.

Online Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan ebook PDF download

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan Doc

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan Mobipocket

Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan EPub

7LQIYT40E5R: Computer Systems Design and Architecture (2nd Edition) By Vincent P. Heuring, Harry F. Jordan