1. Record Nr. UNINA9910974226003321

Autore	Patterson David A
Titolo	Computer organization and design : the hardware/software interface / / David A. Patterson and John L. Hennessy
Pubbl/distr/stampa	San Francisco, Calif., : Morgan Kaufmann London, : Elsevier Science [distributor], c2007
ISBN	1-282-54030-0 9786612540301 0-08-055033-9
Edizione	[3rd ed. revised.]
Descrizione fisica	1 online resource (739 p.)
Collana	Morgan Kaufmann series in computer architecture and design
Altri autori (Persone)	HennessyJohn L
Disciplina	004.21
Soggetti	Computer organization System design - Data processing Computer architecture
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Previous ed.: 1997. Includes index.
Nota di contenuto	Front cover; Title page; Copyright page; Table of Contents; Preface; About This Book; About the Other Book; Changes for the Third Edition; Instructor Support; Concluding Remarks; Acknowledgments for the Third Edition; 1 Computer Abstractions and Technology; 1.1 Introduction; Classes of Computing Applications and Their Characteristics; What You Can Learn in This Book; 1.2 Below Your Program; From a High-Level Language to the Language of Hardware; 1.3 Under the Covers; Anatomy of a Mouse; Through the Looking Glass; Opening the Box; A Safe Place for Data; Communicating with Other Computers Technologies for Building Processors and Memory1.4 Real Stuff: Manufacturing Pentium 4 Chips; 1.5 Fallacies and Pitfalls; Road Map for This Book; 1.6 Concluding Remarks; 1.7 Historical Perspective and Further Reading; 1.8 Exercises; Computers in the Real World; 2 Instructions: Language of the Computer; 2.1 Introduction; 2.2 Operations of the Computer Hardware; 2.3 Operands of the Computer Hardware; Memory Operands; Constant or Immediate Operands; 2.4

Representing Instructions in the Computer; MIPS Fields; 2.5 Logical Operations; 2.6 Instructions for Making Decisions; Loops Case/Switch Statement2.7 Supporting Procedures in Computer Hardware; Using More Registers; Nested Procedures; Allocating Space for New Data on the Stack; Allocating Space for New Data on the Heap; 2.8 Communicating with People; Characters and Strings in Java; 32-Bit Immediate Operands; 2.9 MIPS Addressing for 32-Bit Immediates and Addresses; Addressing in Branches and Jumps; MIPS Addressing Mode Summary; Decoding Machine Language; 2.10 Translating and Starting a Program; Compiler; Assembler; Linker; Loader; Dynamically Linked Libraries; Starting a Java Program; High-Level Optimizations
2.11 How Compilers OptimizeLocal and Global Optimizations; 2.12 How Compilers Work: An Introduction; 2.13 A C Sort Example to Put It All Together; The Procedure; Array Version of Clear; 2.14 Implementing an Object-Oriented Language; 2.15 Arrays versus Pointers; Pointer Version of Clear; Comparing the Two Versions of Clear; 2.16 Real Stuff: IA-32 Instructions; The Intel IA-32; IA-32 Integer Operations; IA-32 Instruction Encoding; IA-32 Conclusion; 2.17 Fallacies and Pitfalls; 2.18 Concluding Remarks; 2.19 Historical Perspective and Further Reading; 2.20 Exercises; Computers in the Real World
3 Arithmetic for Computers3.1 Introduction; 3.2 Signed and Unsigned Numbers; Summary; 3.3 Addition and Subtraction; Summary; 3.4 Multiplication; Sequential Version of the Multiplication Algorithm and Hardware; Signed Multiplication; Faster Multiplication; Multiply in MIPS; Summary; 3.5 Division; A Division Algorithm and Hardware; Signed Division; Faster Division; Divide in MIPS; Summary; 3.6 Floating Point; Floating-Point Representation; Floating-Point Addition; Floating-Point Multiplication; Floating-Point Instructions in MIPS; Accurate Arithmetic; Summary
3.7 Real Stuff: Floating Point in the IA-32

Sommario/riassunto

What's New in the Third Edition, Revised Printing The same great book gets better! This revised printing features all of the original content along with these additional features: Appendix A (Assemblers, Linkers, and the SPIM Simulator) has been moved from the CD-ROM into the printed book Corrections and bug fixesThird Edition featuresNew pedagogical featuresUnderstanding Program Performance -Analyzes key performance issues from the programmer's perspective Check Yourself Questions -Helps students assess their