

大学计算机教育丛书（影印版）

THIRD EDITION
**COMPUTER SYSTEM
ARCHITECTURE**

计算机系统
体系结构

第3版

M. MORRIS MANO

清华大学出版社 · PRENTICE HALL
<http://www.tup.tsinghua.edu.cn>

TP2-3
M17
(3)

414333

COMPUTER SYSTEM ARCHITECTURE

THIRD EDITION

计算机系统体系结构

第3版

M. Morris Mano

California State University

Los Angeles



00414333

清华大学出版社

Prentice-Hall International, Inc.

(京)新登字 158 号

Computer system architecture 3rd ed. / M. Morris Mano

DV83/21

© 1993, 1982, 1976 by Prentice Hall, Inc.

Original edition published by Prentice Hall, Inc., a Simon & Schuster Company.

Prentice Hall 公司授权清华大学出版社在中国境内(不包括中国香港特别行政区、澳门地区和台湾地区)独家出版发行本书影印本。

本书任何部分之内容, 未经出版者书面同意, 不得用任何方式抄袭、节录或翻印。

本书封面贴有 Prentice Hall 激光防伪标签, 无标签者不得销售。

北京市版权局著作权合同登记号: 01-98-0390

图书在版编目(CIP)数据

计算机系统体系结构: 第3版: 英文/马诺(Mano, M. M.)著. - 影印版.
- 北京: 清华大学出版社, 1998.4
(大学计算机教育丛书)
ISBN 7-302-02831-1

I. 计… II. 马… III. 电子计算机-系统结构-英文 IV. TP303

中国版本图书馆 CIP 数据核字(98)第 01687 号

出版者: 清华大学出版社(北京清华大学校内, 邮编 100084)

因特网地址: www.tup.tsinghua.edu.cn

印刷者: 清华大学印刷厂

发行者: 新华书店总店北京科技发行所

开 本: 850×1168 1/32 印张: 17.125

版 次: 1998 年 5 月第 1 版 1998 年 5 月第 1 次印刷

书 号: ISBN 7-302-02831-1/TP·1488

印 数: 0001~5000

定 价: 28.00 元

出版前言

我们的大学生、研究生毕业后,面临的将是一个国际化的信息时代。他们将需要随时查阅大量的外文资料;会有更多的机会参加国际性学术交流活动;接待外国学者;走上国际会议的讲坛。作为科技工作者,他们不仅应有与国外同行进行口头和书面交流的能力,更为重要的是,他们必须具备极强的查阅外文资料获取信息的能力。有鉴于此,在国家教委所颁布的“大学英语教学大纲”中有一条规定:专业阅读应作为必修课程开设。同时,在大纲中还规定了这门课程的学时和教学要求。有些高校除开设“专业阅读”课之外,还在某些专业课拟进行英语授课。但教、学双方都苦于没有一定数量的合适的英文原版教材作为教学参考书。为满足这方面的需要,我们挑选了7本计算机科学方面最新版本的教材,进行影印出版。首批影印出版的6本书受到广大读者的热情欢迎,我们深受鼓舞,今后还将陆续推出新书。希望读者继续给予大力支持。Prentice Hall公司和清华大学出版社这次合作将国际先进水平的教材引入我国高等学校,为师生们提供了教学用书,相信会对高校教材改革产生积极的影响。

清华大学出版社
Prentice Hall 公司

1997.11



Preface

This book deals with computer architecture as well as computer organization and design. Computer architecture is concerned with the structure and behavior of the various functional modules of the computer and how they interact to provide the processing needs of the user. Computer organization is concerned with the way the hardware components are connected together to form a computer system. Computer design is concerned with the development of the hardware for the computer taking into consideration a given set of specifications.

The book provides the basic knowledge necessary to understand the hardware operation of digital computers and covers the three subjects associated with computer hardware. Chapters 1 through 4 present the various digital components used in the organization and design of digital computers. Chapters 5 through 7 show the detailed steps that a designer must go through in order to design an elementary basic computer. Chapters 8 through 10 deal with the organization and architecture of the central processing unit. Chapters 11 and 12 present the organization and architecture of input-output and memory. Chapter 13 introduces the concept of multiprocessing. The plan of the book is to present the simpler material first and introduce the more advanced subjects later. Thus, the first seven chapters cover material needed for the basic understanding of computer organization, design, and programming of a simple digital computer. The last six chapters present the organization and architecture of the separate functional units of the digital computer with an emphasis on more advanced topics.

The material in the third edition is organized in the same manner as in the second edition and many of the features remain the same. The third edition, however, offers several improvements over the second edition. All chapters except two (6 and 10) have been completely revised to bring the material up to date and to clarify the presentation. Two new chapters were added: chapter 9 on pipeline and vector processing, and chapter 13 on multiprocessors. Two sections deal with the reduced instruction set computer (RISC). Chapter 5 has been revised completely to simplify and clarify the design of the basic computer. New problems have been formulated for eleven of the thirteen chapters.

The physical organization of a particular computer including its registers,

the data flow, the microoperations, and control functions can be described symbolically by means of a hardware description language. In this book we develop a simple register transfer language and use it to specify various computer operations in a concise and precise manner. The relation of the register transfer language to the hardware organization and design of digital computers is fully explained.

The book does not assume prior knowledge of computer hardware and the material can be understood without the need of prerequisites. However, some experience in assembly language programming with a microcomputer will make the material easier to understand. Chapters 1 through 3 can be skipped if the reader is familiar with digital logic design.

The following is a brief description of the subjects that are covered in each chapter with an emphasis on the revisions that were made in the third edition.

Chapter 1 introduces the fundamental knowledge needed for the design of digital systems constructed with individual gates and flip-flops. It covers Boolean algebra, combinational circuits, and sequential circuits. This provides the necessary background for understanding the digital circuits to be presented.

Chapter 2 explains in detail the logical operation of the most common standard digital components. It includes decoders, multiplexers, registers, counters, and memories. These digital components are used as building blocks for the design of larger units in the chapters that follow.

Chapter 3 shows how the various data types found in digital computers are represented in binary form in computer registers. Emphasis is on the representation of numbers employed in arithmetic operations, and on the binary coding of symbols used in data processing.

Chapter 4 introduces a register transfer language and shows how it is used to express microoperations in symbolic form. Symbols are defined for arithmetic, logic, and shift microoperations. A composite arithmetic logic shift unit is developed to show the hardware design of the most common microoperations.

Chapter 5 presents the organization and design of a basic digital computer. Although the computer is simple compared to commercial computers, it nevertheless encompasses enough functional capabilities to demonstrate the power of a stored program general purpose device. Register transfer language is used to describe the internal operation of the computer and to specify the requirements for its design. The basic computer uses the same set of instructions as in the second edition but its hardware organization and design has been completely revised. By going through the detailed steps of the design presented in this chapter, the student will be able to understand the inner workings of digital computers.

Chapter 6 utilizes the twenty five instructions of the basic computer to illustrate techniques used in assembly language programming. Programming examples are presented for a number of data processing tasks. The relationship

between binary programs and symbolic code is explained by examples. The basic operations of an assembler are presented to show the translation from symbolic code to an equivalent binary program.

Chapter 7 introduces the concept of microprogramming. A specific microprogrammed control unit is developed to show by example how to write microcode for a typical set of instructions. The design of the control unit is carried-out in detail including the hardware for the microprogram sequencer.

Chapter 8 deals with the central processing unit (CPU). An execution unit with common buses and an arithmetic logic unit is developed to show the general register organization of a typical CPU. The operation of a memory stack is explained and some of its applications are demonstrated. Various instruction formats are illustrated together with a variety of addressing modes. The most common instructions found in computers are enumerated with an explanation of their function. The last section introduces the reduced instruction set computer (RISC) concept and discusses its characteristics and advantages.

Chapter 9 on pipeline and vector processing is a new chapter in the third edition. (The material on arithmetic operations from the second edition has been moved to Chapter 10.) The concept of pipelining is explained and the way it can speed-up processing is illustrated with several examples. Both arithmetic and instruction pipeline is considered. It is shown how RISC processors can achieve single-cycle instruction execution by using an efficient instruction pipeline together with the delayed load and delayed branch techniques. Vector processing is introduced and examples are shown of floating-point operations using pipeline procedures.

Chapter 10 presents arithmetic algorithms for addition, subtraction, multiplication, and division and shows the procedures for implementing them with digital hardware. Procedures are developed for signed-magnitude and signed-2's complement fixed-point numbers, for floating-point binary numbers, and for binary coded decimal (BCD) numbers. The algorithms are presented by means of flowcharts that use the register transfer language to specify the sequence of microoperations and control decisions required for their implementation.

Chapter 11 discusses the techniques that computers use to communicate with input and output devices. Interface units are presented to show the way that the processor interacts with external peripherals. The procedure for asynchronous transfer of either parallel or serial data is explained. Four modes of transfer are discussed: programmed I/O, interrupt initiated transfer, direct memory access, and the use of input-output processors. Specific examples illustrate procedures for serial data transmission.

Chapter 12 introduces the concept of memory hierarchy, composed of cache memory, main memory, and auxiliary memory such as magnetic disks. The organization and operation of associative memories is explained in detail. The concept of memory management is introduced through the presentation of the hardware requirements for a cache memory and a virtual memory system.

Chapter 13 presents the basic characteristics of multiprocessors. Various interconnection structures are presented. The need for interprocessor arbitration, communication, and synchronization is discussed. The cache coherence problem is explained together with some possible solutions.

Every chapter includes a set of problems and a list of references. Some of the problems serve as exercises for the material covered in the chapter. Others are of a more advanced nature and are intended to provide practice in solving problems associated with computer hardware architecture and design. A solutions manual is available for the instructor from the publisher.

The book is suitable for a course in computer hardware systems in an electrical engineering, computer engineering, or computer science department. Parts of the book can be used in a variety of ways: as a first course in computer hardware by covering Chapters 1 through 7; as a course in computer organization and design with previous knowledge of digital logic design by reviewing Chapter 4 and then covering chapters 5 through 13; as a course in computer organization and architecture that covers the five functional units of digital computers including control (Chapter 7), processing unit (Chapters 8 and 9), arithmetic operations (Chapter 10), input-output (Chapter 11), and memory (Chapter 12). The book is also suitable for self-study by engineers and scientists who need to acquire the basic knowledge of computer hardware architecture.

Acknowledgments

My thanks goes to those who reviewed the text: particularly Professor Thomas L. Casavant of the University of Iowa; Professor Murray R. Berkowitz of George Mason University; Professor Cem Ersoy of Brooklyn Polytechnic University; Professor Upkar Varshney of the University of Missouri, Kansas City; Professor Karan Watson of Texas A&M University, and Professor Scott F. Midkiff of the Virginia Polytechnic Institute.

M. Morris Mano

Contents

Preface

xv

CHAPTER ONE

Digital Logic Circuits

1

1-1	Digital Computers		1
1-2	Logic Gates		4
1-3	Boolean Algebra		7
	Complement of a Function	10	
1-4	Map Simplification		11
	Product-of-Sums Simplification	14	
	Don't-Care Conditions	16	
1-5	Combinational Circuits		18
	Half-Adder	19	
	Full-Adder	20	
1-6	Flip-Flops		22
	SR Flip-Flop	22	
	D Flip-Flop	23	
	JK Flip-Flop	24	
	T Flip-Flop	24	
	Edge-Triggered Flip-Flops	25	
	Excitation Tables	27	
1-7	Sequential Circuits		28
	Flip-Flop Input Equations	28	
	State Table	30	
	State Diagram	31	
	Design Example	32	
	Design Procedure	36	
	Problems		37
	References		39

CHAPTER TWO

Digital Components 41

2-1	Integrated Circuits	41
2-2	Decoders	43
	NAND Gate Decoder	45
	Decoder Expansion	46
	Encoders	47
2-3	Multiplexers	48
2-4	Registers	50
	Register with Parallel Load	51
2-5	Shift Registers	53
	Bidirectional Shift Register with Parallel Load	53
2-6	Binary Counters	56
	Binary Counter with Parallel Load	58
2-7	Memory Unit	58
	Random-Access Memory	60
	Read-Only Memory	61
	Types of ROMs	62
	Problems	63
	References	65

CHAPTER THREE

Data Representation 67

3-1	Data Types	67
	Number Systems	68
	Octal and Hexadecimal Numbers	69
	Decimal Representation	72
	Alphanumeric Representation	73
3-2	Complements	74
	($r-1$)'s Complement	75
	(r 's) Complement	75
	Subtraction of Unsigned Numbers	76
3-3	Fixed-Point Representation	77
	Integer Representation	78
	Arithmetic Addition	79
	Arithmetic Subtraction	80
	Overflow	80
	Decimal Fixed-Point Representation	81

3-4	Floating-Point Representation	83
3-5	Other Binary Codes	84
	Gray Code	84
	Other Decimal Codes	85
	Other Alphanumeric Codes	86
3-6	Error Detection Codes	87
	Problems	89
	References	91

CHAPTER FOUR

	Register Transfer and Microoperations	93
4-1	Register Transfer Language	93
4-2	Register Transfer	95
4-3	Bus and Memory Transfers	97
	Three-State Bus Buffers	100
	Memory Transfer	101
4-4	Arithmetic Microoperations	102
	Binary Adder	103
	Binary Adder-Subtractor	104
	Binary Incrementer	105
	Arithmetic Circuit	106
4-5	Logic Microoperations	108
	List of Logic Microoperations	109
	Hardware Implementation	111
	Some Applications	111
4-6	Shift Microoperations	114
	Hardware Implementation	115
4-7	Arithmetic Logic Shift Unit	116
	Problems	119
	References	122

CHAPTER FIVE

	Basic Computer Organization and Design	123
5-1	Instruction Codes	123
	Stored Program Organization	125
	Indirect Address	126

5-2	Computer Registers	127
	Common Bus System	129
5-3	Computer Instructions	132
	Instruction Set Completeness	134
5-4	Timing and Control	135
5-5	Instruction Cycle	139
	Fetch and Decode	139
	Determine the Type of Instruction	141
	Register-Reference Instructions	143
5-6	Memory-Reference Instructions	145
	AND to AC	145
	ADD to AC	146
	LDA: Load to AC	146
	STA: Store AC	147
	BUN: Branch Unconditionally	147
	BSA: Branch and Save Return Address	147
	ISZ: Increment and Skip If Zero	149
	Control Flowchart	149
5-7	Input-Output and Interrupt	150
	Input-Output Configuration	151
	Input-Output Instructions	152
	Program Interrupt	153
	Interrupt Cycle	156
5-8	Complete Computer Description	157
5-9	Design of Basic Computer	157
	Control Logic Gates	160
	Control of Registers and Memory	160
	Control of Single Flip-Flops	162
	Control of Common Bus	162
5-10	Design of Accumulator Logic	164
	Control of AC Register	165
	Adder and Logic Circuit	166
	Problems	167
	References	171

CHAPTER SIX

Programming the Basic Computer

6-1	Introduction	173
6-2	Machine Language	174

6-3	Assembly Language		179
	Rules of the Language	179	
	An Example	181	
	Translation to Binary	182	
6-4	The Assembler		183
	Representation of Symbolic Program		
	in Memory	184	
	First Pass	185	
	Second Pass	187	
6-5	Program Loops		190
6-6	Programming Arithmetic and Logic		
	Operations		192
	Multiplication Program	193	
	Double-Precision Addition	196	
	Logic Operations	197	
	Shift Operations	197	
6-7	Subroutines		198
	Subroutines Parameters and Data Linkage	200	
6-8	Input-Output Programming		203
	Character Manipulation	204	
	Program Interrupt	205	
	Problems		208
	References		211

CHAPTER SEVEN

	Microprogrammed Control		213
7-1	Control Memory		213
7-2	Address Sequencing		216
	Conditional Branching	217	
	Mapping of Instruction	219	
	Subroutines	220	
7-3	Microprogram Example		220
	Computer Configuration	220	
	Microinstruction Format	222	
	Symbolic Microinstructions	225	
	The Fetch Routine	226	
	Symbolic Microprogram	227	
	Binary Microprogram	229	

7-4	Design of Control Unit	231
	<i>Microprogram Sequencer</i> 232	
	Problems	235
	References	238

CHAPTER EIGHT

	Central Processing Unit	241
8-1	Introduction	241
8-2	General Register Organization	242
	<i>Control Word</i> 244	
	<i>Examples of Microoperations</i> 246	
8-3	Stack Organization	247
	<i>Register Stack</i> 247	
	<i>Memory Stack</i> 249	
	<i>Reverse Polish Notation</i> 251	
	<i>Evaluation of Arithmetic Expressions</i> 253	
8-4	Instruction Formats	255
	<i>Three-Address Instructions</i> 258	
	<i>Two-Address Instructions</i> 258	
	<i>One-Address Instructions</i> 259	
	<i>Zero-Address Instructions</i> 259	
	<i>RISC Instructions</i> 259	
8-5	Addressing Modes	260
	<i>Numerical Example</i> 264	
8-6	Data Transfer and Manipulation	266
	<i>Data Transfer Instructions</i> 267	
	<i>Data Manipulation Instructions</i> 268	
	<i>Arithmetic Instructions</i> 269	
	<i>Logical and Bit Manipulation Instructions</i> 270	
	<i>Shift Instructions</i> 271	
8-7	Program Control	273
	<i>Status Bit Conditions</i> 274	
	<i>Conditional Branch Instructions</i> 275	
	<i>Subroutine Call and Return</i> 278	
	<i>Program Interrupt</i> 279	
	<i>Types of Interrupts</i> 281	
8-8	Reduced Instruction Set Computer (RISC)	282
	<i>CISC Characteristics</i> 283	
	<i>RISC Characteristics</i> 284	

Overlapped Register Windows	285	
Berkeley RISC I	288	
Problems		291
References		297

CHAPTER NINE

Pipeline and Vector Processing		299
9-1 Parallel Processing		299
9-2 Pipelining		302
General Considerations	304	
9-3 Arithmetic Pipeline		307
9-4 Instruction Pipeline		310
Example: Four-Segment Instruction Pipeline	311	
Data Dependency	313	
Handling of Branch Instructions	314	
9-5 RISC Pipeline		315
Example: Three-Segment Instruction Pipeline	316	
Delayed Load	317	
Delayed Branch	318	
9-6 Vector Processing		319
Vector Operations	321	
Matrix Multiplication	322	
Memory Interleaving	324	
Supercomputers	325	
9-7 Array Processors		326
Attached Array Processor	326	
SIMD Array Processor	327	
Problems		329
References		330

CHAPTER TEN

Computer Arithmetic		333
10-1 Introduction		333
10-2 Addition and Subtraction		334
Addition and Subtraction with Signed-Magnitude Data	335	

	Hardware Implementation	336	
	Hardware Algorithm	337	
	Addition and Subtraction with Signed-2's Complement Data	338	
10-3	Multiplication Algorithms		340
	Hardware Implementation for Signed-Magnitude Data	341	
	Hardware Algorithm	342	
	Booth Multiplication Algorithm	343	
	Array Multiplier	346	
10-4	Division Algorithms		348
	Hardware Implementation for Signed-Magnitude Data	349	
	Divide Overflow	351	
	Hardware Algorithm	352	
	Other Algorithms	353	
10-5	Floating-Point Arithmetic Operations		354
	Basic Considerations	354	
	Register Configuration	357	
	Addition and Subtraction	358	
	Multiplication	360	
	Division	362	
10-6	Decimal Arithmetic Unit		363
	BCD Adder	365	
	BCD Subtraction	368	
10-7	Decimal Arithmetic Operations		369
	Addition and Subtraction	371	
	Multiplication	371	
	Division	374	
	Floating-Point Operations	376	
	Problems		376
	References		380

CHAPTER ELEVEN

Input-Output Organization 381

11-1	Peripheral Devices		381
	ASCII Alphanumeric Characters	383	
11-2	Input-Output Interface		385
	I/O Bus and Interface Modules	386	
	I/O versus Memory Bus	387	

	<i>Isolated versus Memory-Mapped I/O</i>	388	
	<i>Example of I/O Interface</i>	389	
11-3	Asynchronous Data Transfer		391
	Strobe Control	391	
	Handshaking	393	
	Asynchronous Serial Transfer	396	
	Asynchronous Communication Interface	398	
	First-In, First-Out Buffer	400	
11-4	Modes of Transfer		402
	Example of Programmed I/O	403	
	Interrupt-Initiated I/O	406	
	Software Considerations	406	
11-5	Priority Interrupt		407
	Daisy-Chaining Priority	408	
	Parallel Priority Interrupt	409	
	Priority Encoder	411	
	Interrupt Cycle	412	
	Software Routines	413	
	Initial and Final Operations	414	
11-6	Direct Memory Access (DMA)		415
	DMA Controller	416	
	DMA Transfer	418	
11-7	Input-Output Processor (IOP)		420
	CPU-IOP Communication	422	
	IBM 370 I/O Channel	423	
	Intel 8089 IOP	427	
11-8	Serial Communication		429
	Character-Oriented Protocol	432	
	Transmission Example	433	
	Data Transparency	436	
	Bit-Oriented Protocol	437	
	Problems		439
	References		442

CHAPTER TWELVE

Memory Organization 445

12-1	Memory Hierarchy	445
12-2	Main Memory	448
	RAM and ROM Chips	449