

---

# **MICROPROCESSORS/ MICROCOMPUTERS**

**ARCHITECTURE, SOFTWARE, AND SYSTEMS**

---

**ADI J. KHAMBATA**

73.876  
K45

---

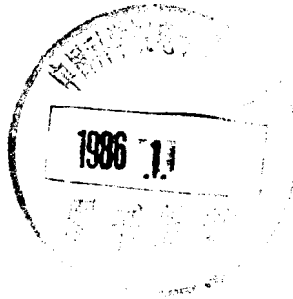
# **MICROPROCESSORS/ MICROCOMPUTERS**

**ARCHITECTURE, SOFTWARE, AND SYSTEMS**

---

**DI J. KHAMBATA**

PAUL TECHNICAL-VOCATIONAL INSTITUTE



**JOHN WILEY & SONS**

NEW YORK CHICHESTER BRISBANE TORONTO SINGAPORE

8650004

DR 63/02

Copyright © 1982, by John Wiley & Sons, Inc.

All rights reserved. Published simultaneously in Canada.

Reproduction or translation of any part of this work beyond that permitted by Sections 107 and 108 of the 1976 United States Copyright Act without the permission of the copyright owner is unlawful. Requests for permission or further information should be addressed to the Permissions Department, John Wiley & Sons.

**Library of Congress Cataloging in Publication Data:**  
Khambata, Adi J.

Microprocessors/microcomputers.

Includes indexes.

1. Microprocessors. 2. Microcomputers.

I. Title.

QA76.5.K43      001.64'04      81-11360  
AACR2

Printed in the United States of America

10 9 8 7 6 5 4 3 2

---

# PREFACE

Advances in semiconductor technology have revolutionized the computer field; this is impacting many aspects of American life and industrial activities. The availability of inexpensive computing capability permits applications that were not considered potential candidates for computerization a few years ago. Microprocessors/microcomputers appear in many consumer products, from computer games and kitchen appliances to automobiles. The advent of these low-cost computing systems has led to the development of still another field, until now completely unknown and foreign to the traditional computer industry: computer hobbyists and home computer use.

Understandably, this has created a voracious appetite for microcomputer education. Many segments of industry that were previously not concerned with computers are in the midst of a digital world, requiring personnel trained in the field of microcomputers. Several companies have set up specialized, in-house, microcomputer training courses. Others are relying on established educational institutions to supply them with trained graduates in this field. The major academic problem encountered by most educational institutions is the formation of the required curriculum and the availability of the appropriate teaching materials. The literature abounds with excellent books and other publications on microprocessors/microcomputers; however, most of them are not suitable for classroom use, nor can curriculum be developed around them. This book, based on microcomputer courses that I designed and taught at the St. Paul Technical-Vocational Institute, St. Paul, Minnesota, and at local industries over the past several years, will fill this need.

This book satisfies the requirements of four distinct user groups. First, it can be used in community colleges, vocational schools, and other two-year educational institutions. A group of microprocessor courses can be designed around the basic material in the book. Product supplements to the basic textbook provide sufficient background for the practical application of specific microcomputer systems in the laboratory. Second, each topic starts out at a very fundamental level; the presentations and discussions proceed to greater depth and higher levels, so it could also be used at institutions of higher learning such as four-year colleges and universities. Third, many persons working in industry need and desire to learn about and use microcomputers. Unfortunately, not all of these people can attend courses offered by educational institutions. Self-study may be their only option, and the reasonably detailed explanations and worked-out examples in this book will be helpful to these people. Finally, hobbyists will find this book useful because it is written so that they can scan a subject lightly or pursue it in greater depth, depending on desire and requirements.

It is assumed that students who use this book have some prior knowledge of digital logic and basic logic functions such as AND gates, OR gates, flip-flops, and logic

blocks (registers, counters, etc.). These items are not included here, since they are adequately covered in many other excellent publications. The book is divided into two sections. The first covers the hardware aspects of microcomputers, including the central processor (CPU) and the commonly used schemes for interfacing the CPU with the outside world. The second section covers the software aspects of microcomputers. This includes a chapter on BASIC, which is presently the most popular higher-level language used in microcomputer systems.

Chapter One is a fundamental review of digital computers. The architecture of the CPU and the sequence of basic operations are covered here. Number systems and binary data coding are discussed in Chapter Two. The pure binary system, the octal system, the binary-coded-decimal system (BCD), and the hexadecimal system are included. The CPU architecture is the topic of Chapter Three. Busses, which are an important feature of microprocessors, are covered along with machine and instruction cycles. The principal registers and counters of the CPU and their respective functions are then presented. Chapter Four deals with microprocessor instructions. The basic instruction formats for both the memory-reference and the nonmemory-reference instructions are treated. The various addressing modes are presented in Chapter Five. Special attention is given to the indirect addressing mode, which is troublesome to some students. Chapter Six discusses how instructions are executed by the CPU, the microsteps or microinstructions that comprise a macroinstruction are shown, and the basic concept of a microprogrammable microprocessor is introduced. Memory chips, which are used in microcomputer systems, are briefly presented in Chapter Seven. Programmed I/O transfers (both conditional and unconditional) are the subject of Chapter Eight. Chapter Nine describes the interrupt I/O. The vectored interrupt, the software-pollled interrupt, and the popular daisy-chained interrupt are examined. Multi-level priority interrupts are also included. Direct Memory Access (DMA) is described in Chapter Ten. Serial I/O transfers are the subject of Chapter Eleven; conversion and synchronization logic and the problems of data identification in serial bit streams are studied in this chapter. Programmable I/O interfaces are described in Chapter Twelve; programmable interfaces for both serial and parallel transfers are included. Chapter Thirteen analyzes D/A and A/D converters. Many microprocessors are required to interface with equipments that have nondigital signals, and so appropriate D/A and A/D interfaces are required.

Chapter Fourteen (which begins the software section) introduces the microcomputer software development cycle. Problem definition and flowcharting are the subject of Chapter Fifteen. Chapter Sixteen shows how to organize the data for processing purposes; exponential notation, the sign convention, and floating-point operations are also presented. The data transformation process is the subject of Chapter Seventeen, which also describes the fundamental functional statements. Chapter Eighteen is entirely devoted to BASIC (Beginner's All-Purpose Symbolic Instruction Code), the higher-level language widely used in microcomputer systems today. Assemblers and interpreters are dealt with in Chapter Nineteen. Interpreters are used frequently in home computers, so this chapter is of special interest to hobbyists. Chapter Twenty briefly describes operating systems and systems software.

To aid readers in learning the material, several chapter elements have been incorporated into the text. These include numerous worked-out examples and figures, end-

of-chapter summaries, review questions, and problems. Three appendixes are also included. Appendix A gives the popular ASCII character set and the associated codes. The subject of system testing and checkout and the various testing approaches, commonly used with microcomputer systems, is examined in Appendix B. The logic analyzer, briefly treated in Appendix C, is the most powerful microcomputer troubleshooting tool available today. A glossary of important terms follows the appendixes.

In addition to offering theoretical descriptions and discussions, it is desirable to describe at least one real, existing microcomputer system. However, including a description of any one product in this book would immediately have dated the publication and also made it unacceptable to other users who may be committed to another product. This dilemma was resolved by adopting a unique approach. A series of separate, soft-cover, supplements, each covering a popular microcomputer system, will accompany the basic text. Four such supplements are presently planned. They will cover the Zilog Z-80, the popular Intel 8080 series, the Motorola MC6800, and the 6502 by MOS Technology. This approach will enable us to introduce additional supplements on future products that find acceptance in educational institutions as well as update existing supplements as needed. It also allows users of this book to purchase only the supplements that are of direct use for their specific needs. In addition to the paperback supplements, separate laboratory manuals—on the most popular microcomputers—will also be published.

When writing a book such as this, it often becomes necessary to borrow some material from prior publications of computer manufacturers and other publishers. I wish to express my appreciation and thanks to the following for giving me permission to use material from their respective publications in this textbook: McGraw-Hill Book Company, New York, Intel Corporation, Santa Clara, California, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, and John Wiley & Sons, Inc., Publishers, New York.

I would like to thank the many people who helped me during the writing and publication of this book. First, Irving L. Kosow, Series Editor at Wiley, meticulously edited the original manuscript and helped me transform it into a finished product. Judy Green, Engineering Technology Editor, and her staff were extremely helpful during the publication process. Their efforts and cooperation are most appreciated. Dr. George Richter, Technical Division Manager at St. Paul Technical-Vocational Institute, constantly encouraged and supported me. Several of my students and former engineering colleagues at Sperry Univac reviewed parts of the manuscript and offered many valuable comments and suggestions; special contributions were made by Philip Gaines, Walter Knights, and Richard Paske. I would also like to thank the following reviewers, whose comments and suggestions were invaluable in preparing the final version of the manuscript: Louis Gross, Columbus Technical Institute; James King, Joliet Junior College; Arthur Seidman, Pratt Institute; Dave Terrell, ITT Technical Institute; J.W. Toliver, University of Houston; and Charles Van Buren, DeVry Technical Institute.

This book is a Khambata family project: my wife Ruth and daughter Pixie typed the manuscript; my son Danny, assisted by his wife Renee, drew the diagrams; and my son Jim and his wife Shelly did all the proofreading. I thank all of them.

Adi J. Khambata

---

# CONTENTS

## Section One MICROCOMPUTER ARCHITECTURE

### 1 FUNDAMENTALS OF DIGITAL COMPUTERS 5

- 1-1 INTRODUCTORY REMARKS 5
- 1-2 Introduction to the Digital Computer 6
  - 1-2.1 The Typical System 6
  - 1-2.2 The Memory System 7
  - 1-2.3 The Central Processing Unit (CPU) 8
  - 1-2.4 The Input-Output Ports (I/O) 9
- 1-3 THE CPU ARCHITECTURE 9
  - 1-3.1 The Functional Subsystems 9
  - 1-3.2 Registers and Counters 9
    - 1-3.2.1 The Accumulator 11
    - 1-3.2.2 The Program Counter 12
    - 1-3.2.3 The Instruction Register and Decoder 13
    - 1-3.2.4 The Address Register/Counter 14
  - 1-3.3 The Arithmetic Logic Unit (ALU) 14
  - 1-3.4 Timing and Control 15
- 1-4 THE SEQUENCE OF BASIC OPERATIONS 15
  - 1-4.1 The Basic Timing Sequence 15
  - 1-4.2 Instruction Fetch/Execution Sequence 16
  - 1-4.3 Data Memory Read Operation 17
  - 1-4.4 Data Memory Write Operation 17
  - 1-4.5 Input/Output Operations 17
  - 1-4.6 Interrupts 18
- 1-5 DIGITAL COMPUTER IN BLOCK DIAGRAM FORM 18
  - 1-5.1 Realistic Organization 18
  - 1-5.2 The Microcomputer Organization 18
- 1-6 THE MICROCOMPUTER STRUCTURE 21
- 1-7 SUMMARY 22
- 1-8 REVIEW QUESTIONS 22

### 2 NUMBER SYSTEMS AND BINARY DATA CODING IN MICROCOMPUTERS 24

- 2-1 BINARY DATA AND NUMBER SYSTEMS 24
- 2-2 THE PURE BINARY NUMBERS 26
  - 2-2.1 Binary Addition 28
  - 2-2.2 Binary Subtraction in Ones Complement 30
    - 2-2.2.1 The Sign Convention 30

2-2.2.2 The Complement Method	30
2-2.3 Binary Subtraction in Twos Complement	32
2-3 THE OCTAL SYSTEM	34
2-3.1 Why the Octal System?	34
2-3.2 Notation in the Octal System	35
2-3.3 Decimal-to-Octal Conversion	37
2-3.4 Octal-to-Decimal Conversion	38
2-4 THE BINARY-CODED-DECIMAL SYSTEM (BCD)	38
2-4.1 Why the BCD System?	38
2-4.2 Notation in the BCD System	39
2-4.3 Decimal-to-BCD Conversion	39
2-4.4 BCD-to-Pure Binary Conversion	40
2-4.5 BCD Addition	42
2-4.6 BCD Subtraction	44
2-4.7 Character Representation in BCD	45
2-5 THE HEXADECIMAL SYSTEM	45
2-6 INTERPRETED BINARY DATA	46
2-7 CHARACTER CODES	47
2-8 SUMMARY	47
2-9 PROBLEMS AND EXERCISES	48
<b>3 THE MICROPROCESSOR ARCHITECTURE</b>	<b>50</b>
3-1 MICROPROCESSOR INTERNAL BUS STRUCTURES	50
3-1.1 What are Busses?	50
3-1.2 The Memory Busses	53
3-1.3 The I/O Busses	55
3-1.4 The Memory-I/O Shared Busses	56
3-1.5 Multiplexed Bus System Configuration	56
3-2 THE MACHINE AND INSTRUCTION CYCLES	58
3-2.1 The Clocking System and Sync Pulse	58
3-2.2 The Machine Cycle	60
3-2.3 The Instruction Cycle	60
3-2.3.1 The Fixed Instruction Cycle	60
3-2.3.2 The Variable Instruction Cycle	62
3-3 INSTRUCTION FLOW IN THE CPU	65
3-4 DATA FLOW IN THE MICROCOMPUTER	66
3-5 REGISTERS AND COUNTERS	67
3-5.1 General Comments	67
3-5.2 The Stack	67
3-5.2.1 The Cascade Stack	67
3-5.2.2 The Stack Pointer	70
3-5.3 The Status Register	72
3-5.3.1 The Sign Flag	73
3-5.3.2 The Overflow Flag	73
3-5.3.3 The Carry Flag	76
3-5.3.4 The Zero Flag	77
3-5.3.5 The Parity Flag	77
3-5.4 General-Purpose Registers (Scratch pad)	77



3-5.5 Other ALU Registers	77
3-6 SUMMARY	78
3-7 REVIEW QUESTIONS	78
3-8 PROBLEMS AND EXERCISES	80
<b>4 MICROPROCESSOR INSTRUCTIONS</b>	<b>82</b>
4-1 WHAT ARE INSTRUCTIONS?	82
4-2 THE INSTRUCTION SET	84
4-2.1 Introductory Remarks	84
4-2.1.1 Categorization by Computer Section	85
4-2.1.2 Categorization by Functions Performed	85
4-2.2 Transfer Data, Arithmetic and Logic Instructions	86
4-2.2.1 Transfer Data Instructions	86
4-2.2.2 Arithmetic Instructions	88
4-2.2.3 Logic Instructions	89
4-2.3 Transfer of Control Instructions	89
4-2.3.1 The HALT Instruction	90
4-2.3.2 The JUMP Instruction	90
4-2.3.3 The BRANCH Operation	92
4-2.3.4 The SKIP Instruction	92
4-2.3.5 The INCREMENT AND SKIP IF ZERO (ISZ) Instruction	93
4-2.4 Subroutine Linking Instructions	97
4-2.5 Operation Instructions	97
4-2.6 I/O Instructions	99
4-3 INSTRUCTION ENCODING	100
4-3.1 Machine Language or Machine Codes	100
4-3.2 Mnemonic Codes (Assembly Language)	100
4-3.3 Assemblers and Interpreters	101
4-4 BASIC INSTRUCTION FORMATS	102
4-4.1 Nonmemory Reference Instructions (NMRI)	102
4-4.1.1 Register-to-Register Transfers	103
4-4.1.2 Nontransfer Instructions	104
4-4.1.3 Multiword NMRI Instructions	104
4-4.1.4 Multiword Instruction Identification	104
4-4.2 Memory Reference Instructions (MRI)	106
4-5 SUMMARY	107
4-6 REVIEW QUESTIONS	108
4-7 PROBLEMS AND EXERCISES	110
<b>5 ADDRESSING MODES IN MICROCOMPUTERS</b>	<b>112</b>
5-1 INTRODUCTORY REMARKS	112
5-2 DIRECT ADDRESSING	113
5-3 INDIRECT ADDRESSING	113
5-3.1 Using Internal CPU Pointers	113
5-3.2 Using Base Page Pointers	119
5-4 PAGE RELATIVE ADDRESSING MODES	122
5-4.1 Current Page Relative Addressing	122

5-4.2	Page 0 Relative Addressing	124
5-4.3	Limitations of the Page Relative Addressing Modes	125
5-5	PROGRAM COUNTER RELATIVE ADDRESSING MODE	126
5-6	IMMEDIATE ADDRESSING	131
5-7	INDEXED ADDRESSING	132
5-7.1	Direct Indexed Addressing	132
5-7.2	Indexed Indirect Addressing (Preindexing)	137
5-7.3	Indirect Indexed Addressing (Postindexing)	138
5-8	OTHER TERMS USED IN ADDRESSING MODES	140
5-9	SUMMARY	141
5-10	REVIEW QUESTIONS	141
5-11	PROBLEMS AND EXERCISES	143
<b>6</b>	<b>INSTRUCTION EXECUTION AND MICROSEQUENCES</b>	<b>148</b>
6-1	INTRODUCTORY REMARKS	148
6-2	HOW THE CPU EXECUTES INSTRUCTIONS	149
6-2.1	The Macroinstructions	149
6-2.2	The Microinstructions	149
6-2.3	The Internal Data Bus	149
6-3	SIMPLE GENERALIZED EXAMPLES OF MICROINSTRUCTIONS	149
6-4	GENERALIZED REALISTIC $\mu$ P CPU	156
6-4.1	The Scratch-pad Memory	156
6-4.2	The Index Register	158
6-4.3	The Cascade Stack	158
6-4.4	Communicating with the Shared Address Bus	158
6-4.5	Communicating with the Shared Data Bus	159
6-4.6	Realistic, Typical $\mu$ P CPU Block Diagram	161
6-4.7	The Control Unit and Control Signals	161
6-4.7.1	Control Signals Functions	161
6-4.7.2	Data Transfer Control Logic	163
6-5	MICROPROGRAMMING A $\mu$ P CPU	165
6-5.1	The CPU Features and the Block Diagram	165
6-5.2	The Mode Control Signals	167
6-5.3	The ALU Function Initiate Signals	167
6-5.4	The Data Transfer Signals	168
6-6	EXAMPLES OF MICROPROGRAMMING THE LEO $\mu$ P	170
6-7	THE MICROPROGRAMMABLE MACHINE	175
6-7.1	The Control Memory	175
6-7.2	Advantages and Disadvantages	176
6-7.2.1	Advantages	176
6-7.2.2	Disadvantages	176
6-8	SUMMARY	177
6-9	REVIEW QUESTIONS	178
6-10	PROBLEMS AND EXERCISES	179
<b>7</b>	<b>MICROCOMPUTER MEMORIES</b>	<b>181</b>
7-1	INTRODUCTION	181
7-1.1	Microcomputer Memory Philosophy	181

7-1.2	Chip Memories	181
7-2	DEFINITIONS OF MEMORY TYPES	181
7-3	ROMS IN MICROCOMPUTERS	183
7-3.1	ROMS	183
7-3.2	PROMS	184
7-3.3	EPROMS	184
7-3.4	EAROMS	185
7-3.5	The Use of ROMS	185
7-4	ORGANIZATION OF THE PROGRAM MEMORY	186
7-4.1	Introductory Comments	186
7-4.2	Word-Organized ROMs	186
7-4.3	The Address Word	188
7-4.4	The Chip Select Scheme	188
7-5	ORGANIZATION OF THE DATA MEMORY	190
7-5.1	General Comments	190
7-5.2	Bit-Organized RAMS	192
7-6	STANDBY POWER FOR VOLATILE RAMS	192
7-6.1	Power Failure/Return Operation	192
7-6.2	Typical Power-Fail Sensing Circuit	193
7-7	SUMMARY	195
7-8	REVIEW QUESTIONS	196
<b>8</b>	<b>PARALLEL I/O TRANSFERS—PROGRAMMED I/O</b>	<b>198</b>
8-1	DATA TRANSFERS	198
8-2	INTERFACE NETWORK CHIPS	199
8-2.1	The I/O Ports	199
8-2.2	The I/O Device Chips and Port Expansion	200
8-3	WHAT IS PROGRAMMED I/O?	200
8-4	THE DATA TRANSFER SCHEME	202
8-4.1	The Transfer Instruction	202
8-4.1.1	The IOT Instruction Format	203
8-4.1.2	Typical IOT Instruction	204
8-4.2	The Block Diagram of the Programmed I/O Chip	205
8-4.3	The Output Transfer	207
8-4.4	The Input Transfer	208
8-5	UNCONDITIONAL PROGRAMMED I/O TRANSFER	209
8-5.1	General Comments	209
8-5.2	Input Operation	209
8-5.3	Output Operation	213
8-6	CONDITIONAL PROGRAMMED I/O TRANSFER	214
8-6.1	General Remarks	214
8-6.2	The Protocol in Conditional Transfers	215
8-6.3	The Status Check Logic	215
8-7	SUMMARY	219
8-8	REVIEW QUESTIONS	219
<b>9</b>	<b>PARALLEL I/O TRANSFERS—INTERRUPT I/O</b>	<b>221</b>
9-1	WHY INTERRUPT I/O?	221

9-2 SEQUENCE OF EVENTS	222
9-3 CPU RESPONSE TO INTERRUPTS	224
9-3.1 Sequence of Events	224
9-3.2 INT REQ Querying Logic	225
9-4 INTERRUPT SERVICE INITIATION	226
9-4.1 Sequence of Events	226
9-4.2 Interrupt Handling Subroutine	228
9-4.3 Branching to Subroutine	230
9-4.3.1 Using a BRANCH Instruction	230
9-4.3.2 Using an Address Pointer	232
9-4.3.3 Using an Externally Supplied Address	233
9-4.4 Status Saving and Restoring	234
9-4.5 Identifying the Interrupting Peripheral	235
9-5 THE VECTORED INTERRUPT	236
9-5.1 Introductory Remarks	236
9-5.2 The Sequence of Events	236
9-5.3 Interface Chip Logic	238
9-5.3.1 INT REQ and Peripheral Identification	238
9-5.3.2 Peripheral Selection and Activation	240
9-5.3.3 Data Input Operation	240
9-5.3.4 Data Output Operation	240
9-6 THE SOFTWARE-POLLED INTERRUPT	241
9-6.1 Introductory Remarks	241
9-6.2 The Sequence of Events	241
9-6.3 Interrupt Handling Subroutine	243
9-6.4 Polling Operation in the CPU	245
9-6.5 Interface Chip Logic	246
9-7 THE HARDWARE-POLLED INTERRUPT (DAISY-CHAIN)	246
9-7.1 The Basic Principle	246
9-7.2 The Logic Diagram	249
9-8 MULTILEVEL PRIORITY INTERRUPTS	251
9-8.1 The Operation	251
9-8.2 Interrupt System Block Diagram	253
9-8.3 Multilevel Priority Interrupt Chip	254
9-9 SUMMARY	256
9-10 REVIEW QUESTIONS	258
<b>10 PARALLEL I/O TRANSFERS—DIRECT MEMORY ACCESS</b>	<b>261</b>
10-1 DATA TRANSFERS AND THE TIME ELEMENT	261
10-2 DMA INITIATION	263
10-2.1 Memory Status Indication Mode	264
10-2.2 Cycle Stealing Mode	264
10-3 SINGLE-WORD DMA TRANSFER	265
10-3.1 Using Single Auxiliary Memory	265
10-3.2 Using Several Auxiliary Memories	266
10-4 DMA TRANSFER OF SINGLE DATA BLOCK	267
10-4.1 Basic Requirements for Block Transfer Execution	267
10-4.2 Flowchart of DMA Block Transfers	268

10-5 DMA TRANSFER OF SEVERAL DATA BLOCKS	270
10-5.1 The Situation	270
10-5.2 The Flowchart	270
10-6 DMA LOGIC ACTIVATION IN CPU	272
10-6.1 Using CPU Registers	272
10-6.2 CPU Response to DMA REQ Signal	273
10-7 OPERATION OF THE INTERFACE CHIP	274
10-7.1 DMA REQ and Response	274
10-7.2 Auxiliary Memory Lockin	276
10-7.3 Word and Block Count	276
10-7.4 Data Memory Access Address	277
10-7.5 The Transfer Operation of the First Word	277
10-7.6 Data Memory Address Update	278
10-7.7 Word Counter Update	278
10-7.8 Block Counter Update	278
10-7.9 Termination of DMA Operation	278
10-7.10 Decrement and Terminate DMA Logic	279
10-8 SUMMARY	280
10-9 REVIEW QUESTIONS	281
<b>11 SERIAL I/O TRANSFERS</b>	<b>283</b>
11-1 INTRODUCTION	283
11-2 CONVERSION AND SYNCHRONIZATION LOGIC	283
11-2.1 Block Diagram of Serial I/O	283
11-2.2 Serial/Parallel Conversion	284
11-2.3 Level Normalization	285
11-2.4 Level-to-Bit Conversion and Synchronization	290
11-2.5 Synchronization of Random Input Pulses	291
11-3 DATA IDENTIFICATION IN SERIAL BIT STREAM	294
11-3.1 The Problem	294
11-3.2 Byte Start Identification Method	295
11-3.3 Synchronous and Asynchronous Serial Transmissions	295
11-3.4 Byte Capture Logic for Serial Bit Stream	296
11-4 SUMMARY	298
11-5 REVIEW QUESTIONS	299
<b>12 PROGRAMMABLE I/O INTERFACES</b>	<b>302</b>
12-1 WHAT ARE PROGRAMMABLE INTERFACES?	302
12-2 PROGRAMMABLE INTERFACES FOR SERIAL TRANSFERS	303
12-2.1 The UART	303
12-2.1.1 Basic Functions and Features	303
12-2.1.2 Timing and Synchronization	304
12-2.1.3 Baud Rate and Data Rate	306
12-2.1.4 Data Buffering	309
12-2.1.5 UART Error Indications	311
12-2.1.6 UART Initialization	312
12-2.1.7 Commands and Flags	313
12-2.1.8 UART Block Diagram	315

12-2.2 The USART	317
12-3 PROGRAMMABLE INTERFACES FOR PARALLEL TRANSFERS	318
12-3.1 Basic Functions and Features	318
12-3.2 Additional Features and Capabilities	318
12-3.3 The INTEL 8255 Programmable Peripheral Interface (PPI)	320
12-3.3.1 The Block Diagram	320
12-3.3.2 Modes of Operation	321
12-3.3.3 Control Word—Mode Definition	324
12-3.3.4 Control Word—Control Port (Bit-Set/Reset)	324
12-4 SUMMARY	325
12-5 REVIEW QUESTIONS	326
12-6 PROBLEMS AND EXERCISES	328
<b>13 DIGITAL-TO-ANALOG AND ANALOG-TO-DIGITAL CONVERTERS</b>	330
13-1 INTRODUCTORY REMARKS	330
13-2 DIGITAL-TO-ANALOG CONVERTERS (DACs)	331
13-2.1 Voltage Output DACs	331
13-2.1.1 The Resistor Summing Network	331
13-2.1.2 The Resistor Ladder Network	334
13-2.1.3 The BCD-Weighted DAC	335
13-2.2 Current Output DAC	337
13-2.3 DAC Operating Parameters	340
13-3 ANALOG-TO-DIGITAL CONVERTERS	344
13-3.1 The Comparator	344
13-3.2 The Ramp-Voltage Method	344
13-3.3 Successive-Approximation Method	347
13-3.3.1 Simple-Counter Method	348
13-3.3.2 The Sequencing Method	349
13-3.4 A/D Converter Operating Parameters	354
13-4 SUMMARY	355
13-5 REVIEW QUESTIONS	356
13-6 PROBLEMS AND EXERCISES	357

## **Section Two MICROCOMPUTER SOFTWARE**

<b>14 INTRODUCTION TO MICROCOMPUTER SOFTWARE</b>	363
14-1 INTRODUCTION	363
14-1.1 What is Software?	363
14-1.2 Software versus Hardware	363
14-1.3 Areas of Software Activities	364
14-1.4 Software Development Cycle	364
14-2 SYSTEMS SOFTWARE	367
14-2.1 The Machine Language	367
14-2.2 The Assembler	368
14-2.3 The Compiler	370
14-2.4 The Interpreter	371

14-2.5 The Text Editor	372
14-2.6 Loaders	372
14-2.7 Linkage Editors and Library Loaders	374
14-2.8 Testing, Debugging, and Diagnostic Programs	374
14-2.9 Executive Programs	376
14-3 SUMMARY	376
14-4 REVIEW QUESTIONS	377
<b>15 PROBLEM DEFINITION AND FLOWCHARTING</b>	<b>379</b>
15-1 PROBLEM STATEMENT	379
15-1.1 The Need	379
15-1.2 An Example	380
15-2 FLOWCHARTS	382
15-2.1 What is a Flowchart?	382
15-2.2 When is a Flowchart Prepared?	383
15-2.3 Program Flowchart Symbols	383
15-2.4 Systems Flowchart Symbols	384
15-3 LEVELS OF FLOWCHARTING	386
15-3.1 Concept-Level Flowchart	386
15-3.2 Algorithm-Level Flowchart	387
15-3.3 Instruction-Level Flowchart	391
15-4 PROGRAM LOOPING	396
15-4.1 What are Iterative Loops?	396
15-4.2 Unconditional Iterative Loops	396
15-4.3 Conditional Iterative Loops	397
15-5 SUMMARY	399
15-6 REVIEW QUESTIONS	402
15-7 PROBLEMS AND EXERCISES	404
<b>16 ORGANIZING THE DATA</b>	<b>406</b>
16-1 DATA AND INFORMATION	406
16-1.1 Difference between Data and Information	406
16-1.2 The ASCII Code	407
16-1.3 Packing and Unpacking Data	408
16-1.4 Multiple Precision Capability	410
16-2 NUMERICAL QUANTITIES IN SCIENTIFIC NOTATION	411
16-2.1 Exponential Notation	411
16-2.2 Sign Convention	412
16-2.3 Floating-Point Operations	414
16-3 SUMMARY	422
16-4 REVIEW QUESTIONS	423
16-5 PROBLEMS AND EXERCISES	424
<b>17 ORGANIZING THE DATA TRANSFORMATION PROCESS</b>	<b>426</b>
17-1 ORGANIZATION OF DATA STRUCTURES	426
17-1.1 Graphical Representation of Data Structures	426
17-1.2 Systematic Organization of Data	427

17-1.2.1	Types and Items	427
17-1.2.2	Arrays	429
17-1.2.3	Lists	434
17-1.2.4	Data Structures and Memory Requirements	438
17-2	FUNDAMENTAL FUNCTIONAL STATEMENTS	441
17-3	THE PROBLEM DEFINITION PROCESS	442
17-3.1	Generalized Statements	442
17-3.1.1	Single-Pass Structure	442
17-3.1.2	Iterative Loops Structure	443
17-3.2	Using Subroutines	445
17-3.3	Miscellaneous Statements	448
17-4	SUMMARY	448
17-5	REVIEW QUESTIONS	449
17-6	PROBLEMS AND EXERCISES	450
18	INTRODUCTION TO BASIC	453
18-1	INTRODUCTORY REMARKS	453
18-2	THE STRUCTURE OF BASIC	454
18-2.1	Statements and Line Numbers	454
18-2.2	The Reserved or Key Words	455
18-2.3	The BASIC Operators	456
18-2.3.1	Arithmetic Operators	456
18-2.3.2	Comparison Operators	456
18-2.3.3	Other Operators	457
18-2.4	Assigned Names	457
18-2.4.1	Numeric Variable Names	457
18-2.4.2	Alphabetical Character Strings	458
18-2.4.3	Literal Constants	458
18-2.5	Storage of Arrays	458
18-3	CATEGORIES OF BASIC STATEMENTS	459
18-4	FUNDAMENTAL STATEMENTS OF BASIC	459
18-4.1	The REM Statement	460
18-4.2	The LET Statement	461
18-4.3	The PRINT Statement	463
18-4.4	The INPUT Statement	468
18-4.5	The GØ TØ Statement	468
18-4.6	The FØR/NEXT Statement	471
18-4.7	The IF/THEN Statement	474
18-4.8	The GØSUB and RETURN Statements	476
18-4.9	The READ/DATA Statements	478
18-4.1	The STØP/END Statements	479
18-5	HIERARCHY OF BASIC OPERATORS	481
18-6	CONCLUDING REMARKS	483
18-7	SUMMARY	483
18-8	REVIEW QUESTIONS	484
18-9	PROBLEMS AND EXERCISES	485



**19 INTRODUCTION TO ASSEMBLERS AND INTERPRETERS 488**

- 19-1 INTRODUCTION TO ASSEMBLY LANGUAGE 488
  - 19-1.1 Process of Program Assembling 488
  - 19-1.2 Problems in Manual Assembly 489
- 19-2 THE ASSEMBLER PROGRAM 490
  - 19-2.1 The Objectives 490
  - 19-2.2 The Translation Process 490
  - 19-2.3 Assembly Language Syntax 492
    - 19-2.3.1 Statement Structure 492
    - 19-2.3.2 Label Field 492
    - 19-2.3.3 Operation Code Field 493
    - 19-2.3.4 Argument Field 493
    - 19-2.3.5 Comment Field 493
    - 19-2.3.6 Field Identification 493
  - 19-2.4 Assembly Language Directives (Pseudo-operations) 494
  - 19-2.5 Symbol Tables or Dictionaries 496
    - 19-2.5.1 Fixed Tables 497
    - 19-2.5.2 Dynamic Tables 497
  - 19-2.6 Macro Assembler 497
- 19-3 MECHANICS OF ASSEMBLER OPERATION 501
  - 19-3.1 Single-Pass Assembler 501
  - 19-3.2 Two-Pass Assembler 501
  - 19-3.3 Resident or Self-Assemblers 502
  - 19-3.4 Nonresident Assemblers 502
  - 19-3.5 Cross-Assemblers 502
- 19-4 THE INTERPRETER 503
  - 19-4.1 What is an Interpreter? 503
  - 19-4.2 Advantages of Interpreters 503
  - 19-4.3 Shortcoming of Interpreters 505
- 19-5 THE "BASIC" INTERPRETER 505
  - 19-5.1 File Manipulation Commands 505
  - 19-5.2 Line Editing Commands 506
- 19-6 SUMMARY 507
- 19-7 REVIEW QUESTIONS 508

**20 OPERATING SYSTEMS AND SYSTEMS SOFTWARE 511**

- 20-1 OPERATING THE COMPUTER SYSTEM 511
  - 20-1.1 Manual or Stand-Alone System 511
  - 20-1.2 Systems Software 512
- 20-2 OPERATING SYSTEMS 512
  - 20-2.1 What are Operating Systems? 512
  - 20-2.2 Advantages and Shortcomings 512
- 20-3 PRINCIPAL TASKS PERFORMED BY OS 513
  - 20-3.1 The Program Development Task 513
  - 20-3.2 The Job Control Task 514
    - 20-3.2.1 Basic Batch Processing 514
    - 20-3.2.2 Queued Sequential Batch Processing 515
    - 20-3.2.3 Priority Queued Batch Processing 515