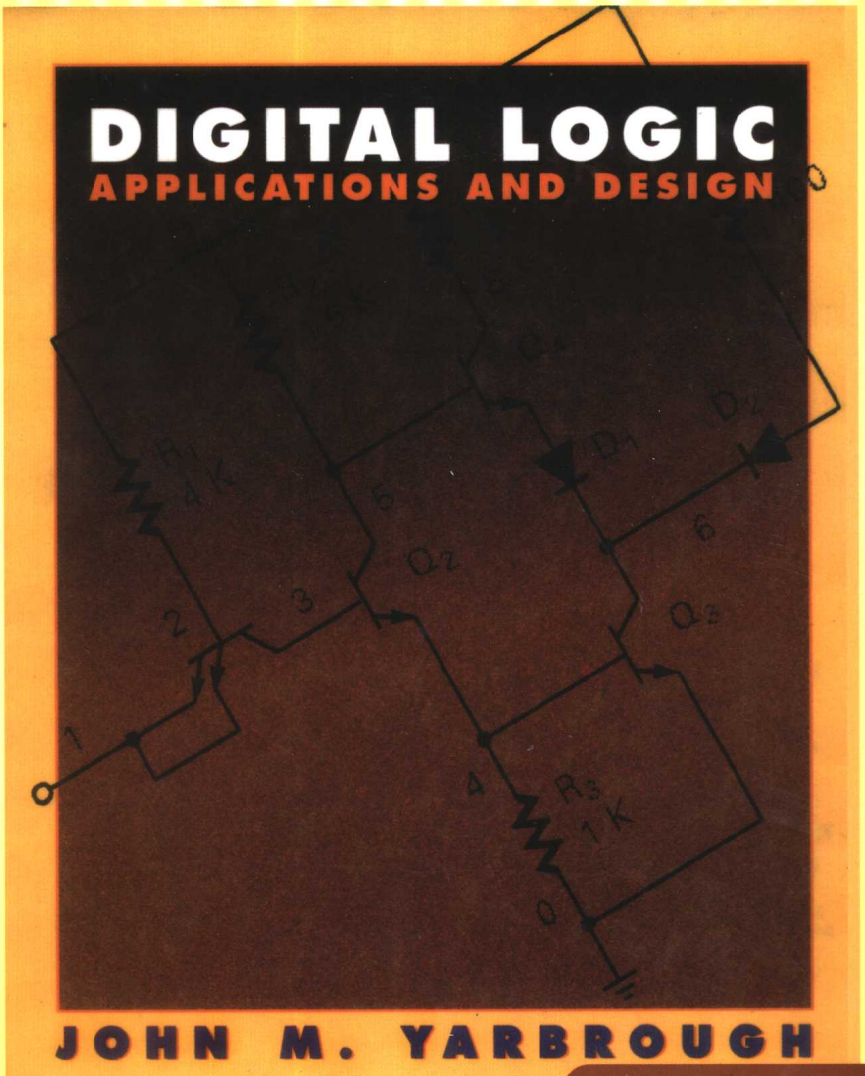
经 典 原 版 书 库

数字逻辑 应用与设计

（英文版）



（美）John M. Yarbrough 著

机械工业出版社
China Machine Press

中信出版社
CITIC PUBLISHING HOUSE

THOMSON

经 典 原 版 书 库

数字逻辑 应用与设计

（英文版）

Digital Logic Applications and Design

（美） John M. Yarbrough 著

机械工业出版社 China Machine Press

中信出版社 CITIC PUBLISHING HOUSE

John M. Yarbrough: Digital Logic Applicatios and Design

Original copyright © 1997 by PWS Publishing Company. All rights reserved.

First published by PWS Publishing Company, a division of Thomson Learning, United States of America.

Reprinted for People's Republic of China by Thomson Asia Pte Ltd and China Machine Press and CITIC Publishing House under the authorization of Thomson Learning. No part of this book may be reproduced in any form without the the prior written permission of Thomson Learning and China Machine Press.

本书影印版由美国汤姆森学习出版集团授权机械工业出版社和中信出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2001-5322

图书在版编目（CIP）数据

数字逻辑应用与设计/（美）亚伯勒（Yarbrough, J. M.）著. - 北京：机械工业出版社，2002.8

（经典原版书库）

书名原文：Digital Logic Applications and Design

ISBN 7-111-10837-X

Ⅰ. 数… Ⅱ. 亚… Ⅲ. 数学逻辑 - 英文 Ⅳ. TP302.2

中国版本图书馆CIP数据核字（2002）第063167号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：华章

北京忠信诚印刷厂印刷·新华书店北京发行所发行

2002年8月第1版第1次印刷

787mm × 1092mm 1/16 · 44.75印张

印数：0 001-3 000册

定价：69.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

PREFACE

The purpose of *Digital Logic: Applications and Design* is to provide electrical, electronic, and computer engineering students, and engineering technology students with material fundamental to the design and analysis of digital circuits. It can also be used by practicing engineers, technologists, and technicians or anyone else, for that matter, with a basic science background, for self-study or as a review.

The typical audience might consist of college sophomores taking the first or second of a series of quarter or semester course(s) in digital logic and circuits. The material is comprehensive; that is, it covers classical combinational and sequential logic. It also covers material on PAL, PLA, GAL, and EPLD realization of logic circuits and has a chapter on circuit analysis of TTL, ECL, and CMOS logic families.

There are 10 chapters in the book, starting with basic principles and number systems, continuing on through combinational logic topics, sequential circuit analysis and design, asynchronous sequential circuit material, and ending with coverage of digital switching circuits. The organization of the book is as follows:

Chapter 1 introduces digital concepts, number systems, binary codes, and arithmetic.

Chapter 2 develops logic symbols and Boolean algebra.

Chapter 3 defines combinational logic and the generation of switching equations from truth tables. It continues in the simplification of switching equations using Karnaugh maps, Quine-McClusky, and map-entered variable algorithms.

Chapter 4 continues the treatment of combinational logic with design problems, multiplexers, decoders, adders, subtractors, and arithmetic logic units.

Chapter 5 treats flip-flops, including timing specifications. The chapter continues with simple counters, MSI integrated circuit counters, and registers.

Chapter 6 introduces synchronous sequential circuit design. Included are Mealy and Moore sequential circuit models, state diagram notation, development of transition and excitation tables, and derivation of excitation and output equations. Analysis of synchronous circuits and the design of counters is included.

Chapter 7 continues with such sequential circuit design topics as state equivalence and reduction, state assignment, algorithm state machines, and linked sequential machines.

Chapter 8 develops asynchronous sequential circuit analysis and design. Fundamental and pulse mode circuits are discussed, analysis of asynchronous circuits, derivation of flow tables, state assignment problems, and design examples are included. Some discussion of data synchronizers and mixed-operating mode logic is given.

Chapter 9 introduces memory chips as a means of realizing combinational and sequential circuits. PLD design is discussed and examples given. The chapter also introduces two field programmable gate array (FPGA) architectures; the Xilinx and Actel FPGA devices.

Chapter 10 concludes the text by presenting a fairly comprehensive treatment of digital integrated circuit logic families. Included are discussions of TTL, ECL, and CMOS circuits. In addition, tristate, open collector, and mixed logic subfamily interfacing is presented.

The electronics aspect of digital design and analysis is absent from many texts on digital logic. Combinational and sequential logic design from a strictly logic perspective is, in my view, incomplete. Digital integrated circuits are electronic devices. Factors such as fan-out considerations, power supply current, propagation delay, and timing analysis, all affect the design of working systems. To address the electronics as well as the logic aspects of digital design, the entire book uses actual IC devices. Becoming familiar with the wide array of ICs available to the logic designer is as important as knowing Boolean algebra and state diagram notation. I have attempted to show the internal logic as well as the logic symbols for many of the SSI and MSI devices commonly used.

Even though PLDs, FPGAs, and ASIC devices are the choices of industry logic designers, an understanding of fundamental principles imparted by using standard ICs prepares the student for dealing with ASIC libraries. Once the logic design principles are learned, any FPGA or ASIC design system can be mastered and used. It would be difficult to move directly from Boolean algebra, combinational, and sequential logic design principles into designing circuits using FPGAs and ASIC devices. FPGA and ASIC design platforms and software are still too expensive to find their way into undergraduate curricula. Writing a text using a given hardware platform and design software would necessarily exclude those not using that particular hardware and software combination.

I know that *almost* anyone can connect a series of digital integrated circuits to provide some useful result under ideal conditions. But what about nonideal conditions? The answer is found in Chapter 10 on "Digital Integrated Circuits," whose purpose is to develop the connection between the "logic" and the electronics that implement that logic. For those who have a good transistor circuits background, much of the early part of Chapter 10 can be eliminated by going straight to the specifications and characteristics of the IC themselves. For students more concerned with digital logic as a prelude to computer architecture courses and software, Chapter 10 can be skipped altogether.

COURSE PARTITIONING

Several partitions are possible with the material presented in the book, depending on relative emphasis, length of lecture and laboratory course (quarter or semester), and the amount of credit given. Some examples are shown here.

此为试读,需要完整PDF请访问: www.ertongbook.com

1. Quarter calendar, three-course sequence with emphasis on circuits as logic. Three lectures and three laboratory courses.
   a. First quarter: 3 credit lecture, 2 credit laboratory.
      Chapter 1. "Digital Concepts and Number Systems"
      Chapter 2. "Boolean Switching Algebra"
      Chapter 3. "Principles of Combinational Logic"
      Chapter 10. "Digital Integrated Circuits" (portions of Chapter 10 may be treated lightly or omitted depending on the emphasis of IC families and transistor circuits)
   b. Second quarter: 3–4 credit lecture, 1–2 credit laboratory.
      Chapter 4. "Analysis and Design of Combinational Logic"
      Chapter 5. "Flip-Flops, Simple Counters, and Registers"
      Chapter 6. "Introduction to Sequential Circuits"
   c. Third quarter: 3–4 credit lecture, 1–2 credit laboratory.
      Chapter 7. "Sequential Circuit Design"
      Chapter 8. "Asynchronous Sequential Circuits"
      Chapter 9. "Programmable Logic and Memory"
2. Semester calendar with the same coverage as given in the three-term course sequence.
   a. First semester.
      Chapter 1. "Digital Concepts and Number Systems"
      Chapter 2. "Boolean Switching Algebra"
      Chapter 10. "Digital Integrated Circuits"
      Chapter 3. "Principles of Combinational Logic"
      Chapter 4. "Analysis and Design of Combinational Logic"
   b. Second semester.
      Chapter 5. "Flip-Flops, Simple Counters, and Registers"
      Chapter 6. "Introduction to Sequential Circuits"
      Chapter 7. "Sequential Circuit Design"
      Chapter 8. "Asynchronous Sequential Circuits"
      Chapter 9. "Programmable Logic and Memory"

Laboratory assignments can be made from selected problems given at the end of each chapter. For example, Chapter 5, "Flip-Flops, Simple Counters, and Registers," might yield the following laboratory assignments:

1. Latches and flip-flops (problems 3, 4, 5, 6, 7, 8, 9)
   a. Construct an S′R′ latch from NAND gates. Verify the characteristic table and derive the characteristic equation.
   b. Add additional NAND gates to convert the S′R′ latch to a level triggered *J–K* flip-flop. Verify the operation by constructing a characteristic table. Identify the operation when $J = K = 1$. Suggest a solution for the problem encountered (convert the level trigger to a pulse trigger).
   c. Convert the pulse triggered *J–K* flip-flop to a *D* flip-flop and a *T* flip-flop. Verify the operation of each by creating characteristic tables. Derive the characteristic equations.
   d. Predict and measure power consumption and propagation delay.
2. Counters (problem 25)
   a. Design a modulo-8 counter using standard IC flip-flops.
      1. Construct the state diagram.

2. Determine the number of flip-flops needed.
3. Make a binary state assignment (000, 001, 010, 111).
4. Create a state table and a transition table.
5. Using *D* flip-flops create an excitation table (compare the transition and excitation tables).
6. Generate the simplified excitation equations.
7. Construct the circuit using TTL integrated circuit packages (*D* flip-flops, AND, NAND, etc. gates).

b. Predict the timing diagram for the modulo-8 counter.

c. Test the circuit operation.

1. Single step through each state and verify state transitions.
2. Connect a function generator to the clock inputs. Set the function generator to produce a square wave between 0 and 5 V at a frequency of 10 KHz. Measure the *Q* outputs of the flip-flops using an oscilloscope and produce a timing diagram.

3. Johnson counter (problem 26)

4. Special purpose pulse generator (problem 33)

5. Register system (problem 43)

SUPPLEMENTAL PACKAGE

- A solutions manual to all text problems (only odd answers are given in the text)
- For selected text illustrations, a set of transparency masters to use as overheads
- A lab manual is being developed, which is unique because other texts at this level do not have one (ask your West sales representative when it will be available)

ACKNOWLEDGMENTS

No author can complete a text without the assistance of many people. I would like to thank Jack Stewart, a student at Oregon Institute of Technology; the publishing professionals at West, and the following reviewers and accuracy checkers who deserve particular mention:

Gail Allwine
Gonzaga University (WA)

Kenneth J. Ayala
Western Carolina University (NC)

Dee Bailey
Mt. Hood Community College (OR)

William E. Barnes
New Jersey Institute of Technology (NJ)

Robert Belge
Syracuse University (NY)

John Brannon
Embry–Riddle Aeronautical University (FL)

Thomas Casavant
University of Iowa (IA)

Aaron Collins
Clemson University (SC)

Darrow Dawson
University of Missouri–Rolla (MO)

Richard Eason
University of Maine (ME)

Joe Enekes
Humber College (Ontario, Canada)

Milos Ercegovac
University of California–Los Angeles (CA)

Barry S. Fagin
Dartmouth College (NH)

Verne Hansen
Weber State University (UT)

Ralph Horvath
Michigan Technological University (MI)

Michael Horn
University of Hartford (CT)

Ernest Joerg
Westchester Community College (NY)

Harold Klock
Ohio University (OH)

Willard Korfhage
Polytechnic University (NY)

Sura Lekhakul
St. Cloud State University (MN)

Albert L. McHenry
Arizona State University (AZ)

James Migel
Red River Community College
(Manitoba, Canada)

John Nagi
Hudson Valley Community College
(NY)

John R. Pavlat
Iowa State University (IA)

James Rankin
St. Cloud State University (MN)

Dennis Suchecki
San Diego Mesa College (CA)

CONTENTS

Preface xi

CHAPTER 1 Digital Concepts and Number Systems 1

Introduction 1

1.1 Digital and Analog: Basic Concepts 1

1.2 Some History of Digital Systems 4

1.3 Impact of Digital Technology on Society 6

1.4 Defining the Problem, an Introduction to Algorithms 7

1.5 Digital Systems Overview 9

1.6 Introduction to Number Systems 9

1.7 Positional Number Systems 10
- 1.7.1 Decimal Numbers 10
- 1.7.2 Binary Numbers 11
- 1.7.3 Octal Numbers 12
- 1.7.4 Hexadecimal Numbers 12
- 1.7.5 Counting in Base *r* 13

1.8 Number System Conversion 14
- 1.8.1 Binary to Hexadecimal Conversion 15
- 1.8.2 Hexadecimal and Octal to Binary Conversion 16
- 1.8.3 Binary to Decimal Conversion 16
- 1.8.4 Successive Division Radix Conversion 17
- 1.8.5 Fractional Radix Conversion, Successive Multiplication 18
- 1.8.6 Radix Conversion Algorithm 20
- 1.8.7 Decimal to Any Radix 21
- 1.8.8 Any Radix to Decimal 23

1.9 Binary Codes 23
- 1.9.1 Natural Binary Coded Decimal 23
- 1.9.2 Binary Codes (Weighted) 24
- 1.9.3 BCD Self-Complementing Codes 25

1.9.4 Unit Distance Code 26
1.9.5 Alphanumeric Codes 28
1.9.6 Signed Number Binary Codes 28
1.9.7 Signed Magnitude Codes 28
1.9.8 Complement Codes 29

1.10 Arithmetic 32
1.10.1 Binary Arithmetic 32
1.10.2 Binary Arithmetic Using Complement Codes 35
1.10.3 Hexadecimal Arithmetic 38

Summary 42

References 43

Glossary 44

Questions and Problems 45

CHAPTER 2 **Boolean Switching Algebra** 48

Introduction 48

2.1 Binary Logic Functions 48
2.1.1 IEEE Logic Symbols 54
2.1.2 Functions, Symbols, and Truth Tables 55

2.2 Switching Algebra 57
2.2.1 Equivalence 58
2.2.2 Closure 58
2.2.3 Identity 59
2.2.4 Associative Properties 59
2.2.5 Distributive Properties 61
2.2.6 Commutative Properties 62
2.2.7 Complement Property 62
2.2.8 Duality Property 62
2.2.9 Absorption Property 62
2.2.10 Idempotency Property 63
2.2.11 Binary Variables and Constants 63
2.2.12 DeMorgan's Theorems 64

2.3 Functionally Complete Operation Sets 68

2.4 Reduction of Switching Equations Using Boolean Algebra 70

2.5 Realization of Switching Functions 73
2.5.1 Conversion of Switching Functions to Logic Diagrams 73
2.5.2 Converting Logic Diagrams to Switching Equations 77

Summary 80

References 80

Glossary 81

Questions and Problems 82

CHAPTER 3 Principles of Combinational Logic 84

Introduction 84

3.1 Definition of Combinational Logic 84
3.1.1 Problem Statements to Truth Tables 85
3.1.2 Deriving Switching Equations 89

3.2 Canonical Forms 91

3.3 Generation of Switching Equations from Truth Tables 93

3.4 Karnaugh Maps 96
3.4.1 Three- and Four-Variable Karnaugh Maps 97
3.4.2 Five- and Six-Variable Karnaugh Maps 107
3.4.3 Simplification Using Five-Variable Karnaugh Maps 109
3.4.4 Simplification Using Six-Variable Karnaugh Maps 112
3.4.5 Incompletely Specified Functions (Don't Care Terms) 113
3.4.6 Simplifying Maxterm Equations 117

3.5 Quine-McClusky Minimization Technique 120
3.5.1 Quine-McClusky Using Don't Care Terms 123
3.5.2 Reduced Prime Implicant Tables 125

3.6 Map-Entered Variables 129

3.7 Mixed Logic Combinational Circuits 135
3.7.1 Logic Symbols 136
3.7.2 Conversion to Bubble Logic 140
3.7.3 Synthesizing Switching Functions Using Bubble Notation 142

3.8 Multiple Output Functions 146

Summary 149

References 150

Glossary 150

Questions and Problems 151

CHAPTER 4 Analysis and Design of Combinational Logic 156

Introduction 156

4.1 General Approach to Combinational Logic Design 156

4.2 Introduction to Digital Integrated Circuits 164

4.3 Decoders 171
4.3.1 BCD Decoders 181

4.4 Encoders 185

4.5 Digital Multiplexers 190
4.5.1 Using Multiplexers as Boolean Function Generators 194

4.6 Adders and Subtractors 201
4.6.1 Cascading Full-Adders 205
4.6.2 Look-Ahead Carry 206

此为试读,需要完整PDF请访问: www.ertongbook.com

4.6.3 MSI Adders 207
4.6.4 Using MSI Adders as Subtractors 208
4.6.5 Using an MSI Adder as a BCD to Excess-3 Code Converter 211
4.6.6 BCD Adder 212
4.7 Binary Comparators 215
4.8 Arithmetic Logic Units 222
4.9 Array Multipliers 225
4.10 Tristate Buffers 228
4.11 Combinational Logic Hazards 231
4.11.1 Static Hazards 232
4.11.2 Dynamic Hazards 236
Summary 237
References 237
Glossary 238
Questions and Problems 239

CHAPTER 5 Flip-Flops, Simple Counters, and Registers 243

Introduction 243
5.1 Sequential Circuit Models 243
5.2 Flip-Flops 248
5.2.1 Flip-Flop Logic Symbols, Function, and Triggering 248
5.3 Flip-Flop Timing Specifications 265
5.3.1 Clock Parameters, Pulse Width, and Skew 265
5.3.2 Flip-Flop Timing, Setup, Hold, and Delay 267
5.3.3 Flip-Flop Metastability 268
5.4 Simple Counters 271
5.4.1 Divide by 2, 4, and 8 Counters (Asynchronous) 272
5.4.2 Johnson Counter (Synchronous) 274
5.4.3 Ring Counter (Synchronous) 275
5.5 MSI Integrated Circuit Counters 276
5.5.1 MSI Asynchronous Counters 277
5.5.2 MSI Synchronous Counters 279
5.5.3 Control Signal Generation by Decoding Counter Outputs 283
5.5.4 A Counter Application: Digital Clock 287
5.5.5 IEEE Standard Symbols for MSI Counters 291
5.6 Registers 295
5.6.1 Registers Data Input and Output 295
5.6.2 Tristate Registers 300
5.6.3 Registers Connected to a Common Data Bus 305
5.6.4 Register Transfer Timing Considerations 310
Summary 312
References 313

Glossary 314
Questions and Problems 316

CHAPTER 6 Introduction to Sequential Circuits 322

Introduction 322
6.1 Mealy and Moore Models 322
6.2 State Machine Notation 323
6.2.1 Present State, Next State 324
6.2.2 State Diagram 324
6.2.3 State Table 328
6.2.4 Transition Table 329
6.2.5 Excitation Table and Equations 330
6.2.6 Excitation Realization Cost 336
6.3 Synchronous Sequential Circuit Analysis 339
6.3.1 Analysis Principles 340
6.3.2 Analysis Examples 340
6.4 Construction of State Diagrams 348
6.4.1 Up-Down Decade Counter 349
6.4.2 Sequence Detectors 349
6.4.3 Serial EX-3 to BCD Code Converter 354
6.5 Counter Design 356
6.5.1 Modulo-8 Synchronous Counter 357
6.5.2 Up-Down Decade Counter Design 360
Summary 369
References 370
Glossary 370
Questions and Problems 371

CHAPTER 7 Sequential Circuit Design 375

Introduction 375
7.1 State Equivalence 375
7.2 State Reduction 376
7.2.1 Equivalence Classes 376
7.2.2 Implication Charts 378
7.3 State Reduction of Incompletely Specified State Tables 384
7.3.1 Merger Graphs 386
7.4 State Assignment Techniques 389
7.4.1 State Assignment Permutations 390
7.4.2 State Assignment Algorithm 392
7.4.3 Implication Graph 396

7.5 Algorithm State Machines 399
7.5.1 ASM Symbols 399
7.5.2 Elapsed Time Measurement, an ASM Design Example 404
7.6 Linked Sequential Machines 413
7.6.1 Computer Simulator and Graphic Plotter Interface, a Linked Sequential Machine Design Example 415
Summary 432
References 433
Glossary 433
Questions and Problems 434

CHAPTER 8 Asynchronous Sequential Circuits 441
Introduction to Asynchronous Sequential Machines 441
8.1 Fundamental and Pulse Mode Asynchronous Sequential Machines 442
8.2 Analysis of Asynchronous Sequential Machines 444
8.3 Deriving Flow Tables 452
8.4 State Assignment 456
8.4.1 Races and Cycles 456
8.4.2 Shared Row State Assignment 458
8.4.3 Multiple Row State Assignment 460
8.4.4 One Hot State Assignment 461
8.5 Asynchronous Design Problems 462
8.5.1 Asynchronous Design Problem 1 463
8.5.2 Asynchronous Design Problem 2 465
8.6 Data Synchronizers 470
8.6.1 Interface Protocol Asynchronous Cell 472
8.7 Mixed Operating Mode Asynchronous Circuits 474
Summary 477
References 478
Glossary 479
Questions and Problems 480

CHAPTER 9 Programmable Logic and Memory 485
Introduction 485
9.1 Memory 486
9.1.1 ROM, PROM, and EPROM 486
9.2 Using an EPROM to Realize a Sequential Circuit 491
9.3 Programmable Logic Devices 495
9.3.1 Programmable Logic Array (PLA) 496
9.3.2 Programmable Array Logic 498

9.3.3 Designing an Up-Down Decade Counter Using a PAL 502
9.3.4 Generic Array Logic 507
9.3.5 Designing a Synchronous Sequential Circuit Using a GAL 509

9.4 Erasable Programmable Logic Devices 514
9.4.1 Altera EP600 EPLD 517
9.4.2 Sequential Circuit Realization Using an EP600 519

9.5 PLD Computer-Aided Design 523
9.5.1 PLD Realization of Combinational Logic 525
9.5.2 Realizing Truth Tables Using a PLD Language 529
9.5.3 Realizing Flip-Flops Using a PLD Language 530
9.5.4 Realizing State Machines Using a PLD Language 530

9.6 Field Programmable Gate Arrays 534
9.6.1 Xilinx FPGA 535
9.6.2 System Development Tools for the Xilinx FPGA 542
9.6.3 Xilinx Macro Library 543
9.6.4 Actel FPGA 543

Summary 549

References 550

Glossary 551

Questions and Problems 552

CHAPTER 10 Digital Integrated Circuits 556

Introduction 556

10.1 Diodes as Switches 556
10.1.1 Diode Gates 558

10.2 Bipolar Transistor Switch 561

10.3 Diode Transistor Logic 562

10.4 Evolution from DTL to TTL 563

10.5 Transistor–Transistor Logic 565
10.5.1 TTL Circuit Operation 567
10.5.2 TTL Specifications 569
10.5.3 TTL Subfamilies 573
10.5.4 Schottky Junctions 578
10.5.5 Comparison of TTL Subfamily Specifications 579
10.5.6 Open Collector TTL Circuits 580
10.5.7 Tristate TTL Devices 584
10.5.8 Mixed TTL Subfamily Fan-Out 585
10.5.9 Other TTL Circuits 587

10.6 Emitter-Coupled Logic 590
10.6.1 Emitter-Coupled Logic Circuit 590
10.6.2 ECL Specifications 594
10.6.3 ECL to TTL and TTL to ECL Interfacing 595

10.7 Complementary Metal Oxide Semiconductor 597
10.7.1 Field Effect Transistors 597
10.7.2 MOSFETs 603
10.7.3 MOSFET Logic Gates 606
10.7.4 CMOS Logic Gates 609
10.7.5 Power Dissipation for High-Speed CMOS 612
10.7.6 Propagation Delay for High-Speed CMOS 613
10.7.7 CMOS Noise Margins 613
10.7.8 CMOS Subfamilies 613
Summary 615
References 615
Glossary 616
Questions and Problems 617

Appendix 1 TTL Analysis Spice Exercise 623
Appendix 2 Answers to Odd-Numbered Questions and Problems 627
Index 689

CHAPTER

1

DIGITAL CONCEPTS AND NUMBER SYSTEMS

INTRODUCTION

Digital circuits, processing discrete information, are found in an astonishingly wide range of electronic systems. Consumer microwave ovens and television, advanced industrial controls, space communications, traffic control radar and hospital critical care units are all designed using digital techniques. Advances in digital circuit manufacturing have provided systems designers with more functions contained in less space, thereby improving system reliability and increasing speed. Digital systems operate on numbers that represent some real logical or arithmetic function. It is the designer's task to establish the relationship between the numbers and the task being performed. This chapter develops the use of numbers and how they represent real world conditions in digital systems.

The use of the words *digit* and *digital* was probably in association with counting; it can apply to decimal numbers (0, 1, 2, 3, . . . , 9) or numbers in another base such as binary (0, 1). Counting in decimal numbers and the number of fingers or toes are probably directly related in human history. More appropriate for students of electronics and computers, digital systems are systems that process discrete information. A digital system is simply one that receives input, processes or controls activity, and outputs information in a discrete or noncontinuous manner. The information may be encoded or represented in a variety of ways; it can be in the familiar base 10 number system or in some other number system, such as base 2 (binary).

Digital design is the *application of a set of rules and techniques for developing digital circuits and subsystems to create a solution for some problem.*

1.1 DIGITAL AND ANALOG: BASIC CONCEPTS

The difference between analog and digital lies in the method of encoding information. Digital electronics uses discrete quantities to represent information while analog electronics deals with continuous signals. **Discrete** means distinct or separated as opposed to **continuous** or connected. Analog systems process information that varies continuously. Examples of analog represented variables include temperature as measured with a mercury thermometer, clocks with hands, sine wave voltages indicated on a galva-

此为试读,需要完整PDF请访问: www.ertongbook.com