PEARSON

国外电子与电气工程经典图书系列

Digital Fundamentals

(Tenth Edition)

数字基础

（第十版）

Thomas L. Floyd 著

（英文影印版）

科学出版社

PEARSON Education

国外电子与电气工程经典图书系列

数字基础

（第十版）

（英文影印版）

Digital Fundamentals

（Tenth Edition）

Thomas L. Floyd 著

科学出版社

北京

图字：01-2011-4031

内 容 简 介

本书全面论述了数字电路设计的基础知识，结合练习、实例和插图，对数字电路设计进行了高度精确的理论解释，故障诊断和应用都使用实际设备进行验证。内容包括数字电路概念、数字系统运作及编码、逻辑门、布尔数学运算和逻辑简化、组合逻辑电路分析及应用、可编程逻辑器件、触发器、计数器、移位记录器、记忆和存储、可编程逻辑及软件、信号接口和进程等。

本书可作为电类各专业数字系统课程的双语教材或参考书，也可供工程技术人员参考。

图书在版编目(CIP)数据

数字基础 = Digital Fundamentals: 英文/(美)弗洛伊德(Floyd, T. L.)著. —10 版. —北京：科学出版社，2011

(国外电子与电气工程经典图书系列)

ISBN 978-7-03-031853-4

Ⅰ.①数… Ⅱ.①弗… Ⅲ. ①数字电路-电路设计-高等学校-教材-英文 Ⅳ. ①TN79

中国版本图书馆 CIP 数据核字(2011)第 137949 号

责任编辑：王鑫光 匡 敏 / 责任印制：张克忠 / 封面设计：迷底书装

科学出版社出版
北京东黄城根北街16号
邮政编码：100717
http://www.sciencep.com
双青印刷厂印刷

科学出版社发行 各地新华书店经销

*

2011 年 8 月第 一 版 开本：787×1092 1/16
2011 年 8 月第一次印刷 印张：54 3/4
印数：1—3 000 字数：1 360 000

定价：108.00 元

（如有印装质量问题，我社负责调换）

PREFACE

This tenth edition of *Digital Fundamentals* continues a long tradition of presenting a strong foundation in the core fundamentals of digital technology. This text provides basic concepts reinforced by plentiful illustrations, examples, exercises, and applications. System application activities, troubleshooting sections, programmable logic, signal interfacing and processing, and computer technology are included in addition to the core fundamentals. New topics and features have been added to this edition, and many other topics have been strengthened or enhanced.

The approach used in *Digital Fundamentals* allows students to master the all-important fundamental concepts before they get into more advanced or optional topics. The range of topics provides the flexibility to accommodate a variety of program requirements. For example, some of the design-oriented or system application topics may not be appropriate in some courses. Some programs may not cover programmable logic, and others may not have time to include computers or signal interfacing and processing. Also, some programs may not cover the details of "inside-the-chip" circuitry. These and other areas can be omitted or lightly covered without affecting the coverage of the fundamental topics. A background in transistor circuits is not a prerequisite for this textbook, and the coverage of integrated circuit technology (inside-the-chip circuits) is optional.

The User's Guide beginning on p. ix provides help in customizing your course, by illustrating a variety of approaches that allow tailoring the table of contents to meet your unique requirements. The organization allows for inclusion or omission of various topics without affecting the other topics that are covered.

New in This Edition

- New page layout and design for better visual appearance and ease of use.
- Revised and improved topics.
- Obsolete devices have been deleted.
- The *System Application Activities* (formerly *Digital System Applications*) have been thoroughly revised and include a Multisim feature.
- More end-of-chapter problems.
- A True/False Quiz at the end of every chapter.
- Boolean simplification coverage now includes the Quine-McClusky method in an appendix as an option.
- Coverage of the cyclic redundancy code (CRC).
- Introduction to multi-core processors.

Standard Features

- Core fundamentals are presented without being intermingled with advanced or peripheral topics.
- *Margin notes* provide information in a condensed form.

- *Key terms* are listed on each chapter opener. Within the chapter, the key terms are highlighted in color boldface. Each key term is defined at the end of the chapter as well as in the comprehensive glossary at the end of the book. Glossary terms are indicated by black boldface in the text.
- Reminders inform students where to find the answers to the various exercises and problems throughout each chapter.
- Full-color format.
- Chapter 14 is designed as a "floating chapter" to provide optional coverage of 1C technology (inside-the-chip circuitry) at any point in the course.
- An outline of main headings, list of objectives, introduction, device list, and preview of the system application activity are on the chapter opener.
- Introduction and objectives are at the beginning of each section within a chapter.
- Checkup exercises conclude each section in a chapter.
- Each worked example has a *Related Problem.*
- *Computer Notes* throughout provide interesting information about computer technology as it relates to the text coverage.
- *Hands-On Tips* throughout provide useful and practical information.
- *System Application Activity*, a feature at the end of many chapters, provides interesting and practical use of digital concepts applied in a system environment.
- Multisim files on a CD provide circuits that are referenced in the text for optional simulation and troubleshooting.
- Chapter summary.
- Multiple-choice self-test at the end of each chapter.
- Extensive sectionalized problem sets at the end of each chapter include troubleshooting, system application activity, and special design problems.
- The operation and application of test instruments, including the oscilloscope, logic analyzer, function generator, and DMM are covered.
- Troubleshooting sections in many chapters.
- Introduction to programmable logic.
- Introduction to signal interfacing and processing.
- Introduction to computer concepts.
- Coverage of CMOS and bipolar 1C technologies.

Accompanying Student Resources

- *Experiments in Digital Fundamentals,* Tenth Edition. Lab manual by Dave Buchla.
- *Multisim CD-ROM.* Packaged with each text, this CD includes circuits from the text simulated in both versions 9 and 10.
- *Companion Website* (**www.prenhall.com/floyd**)

Instructor Resources

- *PowerPoint® CD-ROM* This CD-ROM includes a completely new set of unique PowerPoint slides developed by Dave Buchla. These innovative, interactive slides are coordinated with each chapter and are an excellent tool to supplement classroom presentations.

- *Online Course Support* If your program is offered in a distance learning format, please contact your local Prentice Hall sales representative for a list of product solutions.
- *Instructor's Resource Manual* Includes worked-out solutions to chapter problems, solutions to System Application Activities, an overview of IEEE Std. 19-1984, and worked-out lab results for the lab manual by Dave Buchla.
- *TestGen* This edition features over 900 questions.

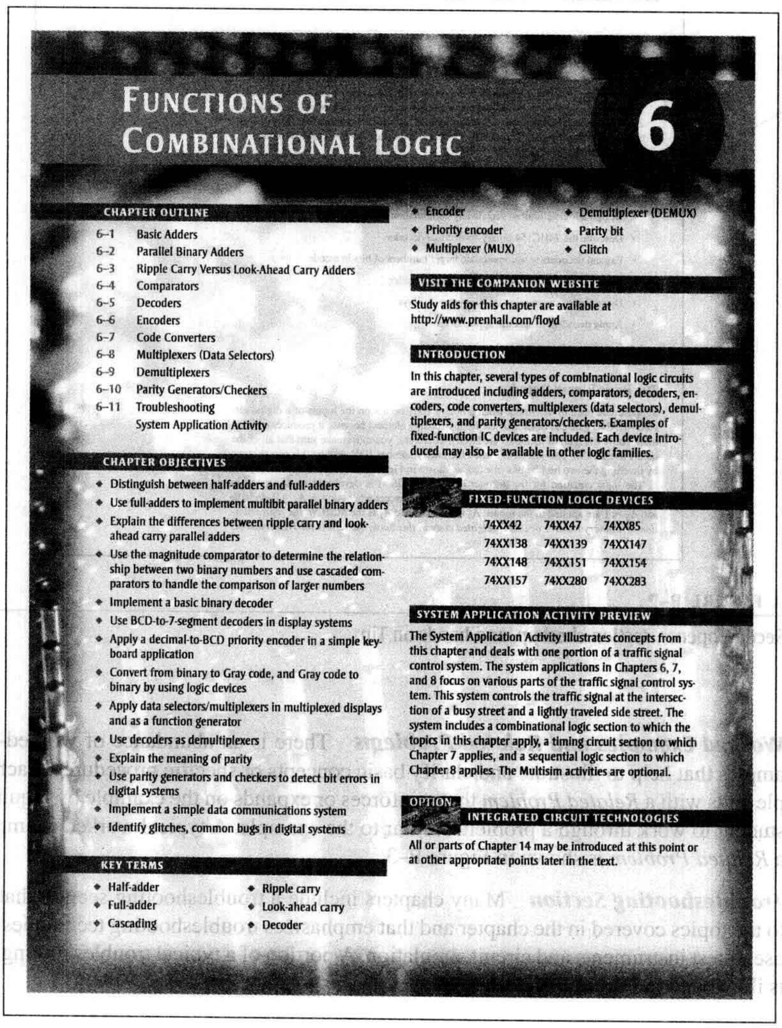
Instructor Resource Center

Register today at www.prenhall.com to access instructor resources digitally.

To access supplementary materials online, instructors need to request an instructor access code. Go to **www.pearsonhighered.com/irc,** where you can register for an instructor access code. Within 48 hours after registering, you will receive a confirming e-mail, including an instructor access code. Once you have received your code, go to the site and log on for full instructions on downloading the materials you wish to use.

Illustration of Book Features

Chapter Opener Each chapter begins with an opener, which includes a list of the sections in the chapter, chapter objectives, introduction, a list of specific devices, a list of key terms, a brief preview of the System Application Activity, and a website reference for chapter study aids. A typical chapter opener is shown in Figure P–1.



FUNCTIONS OF COMBINATIONAL LOGIC 6

CHAPTER OUTLINE

6–1 Basic Adders
6–2 Parallel Binary Adders
6–3 Ripple Carry Versus Look-Ahead Carry Adders
6–4 Comparators
6–5 Decoders
6–6 Encoders
6–7 Code Converters
6–8 Multiplexers (Data Selectors)
6–9 Demultiplexers
6–10 Parity Generators/Checkers
6–11 Troubleshooting
System Application Activity

CHAPTER OBJECTIVES

- Distinguish between half-adders and full-adders
- Use full-adders to implement multibit parallel binary adders
- Explain the differences between ripple carry and look-ahead carry parallel adders
- Use the magnitude comparator to determine the relationship between two binary numbers and use cascaded comparators to handle the comparison of larger numbers
- Implement a basic binary decoder
- Use BCD-to-7-segment decoders in display systems
- Apply a decimal-to-BCD priority encoder in a simple keyboard application
- Convert from binary to Gray code, and Gray code to binary by using logic devices
- Apply data selectors/multiplexers in multiplexed displays and as a function generator
- Use decoders as demultiplexers
- Explain the meaning of parity
- Use parity generators and checkers to detect bit errors in digital systems
- Implement a simple data communications system
- Identify glitches, common bugs in digital systems

KEY TERMS

- Half-adder
- Full-adder
- Cascading
- Ripple carry
- Look-ahead carry
- Decoder
- Encoder
- Priority encoder
- Multiplexer (MUX)
- Demultiplexer (DEMUX)
- Parity bit
- Glitch

VISIT THE COMPANION WEBSITE

Study aids for this chapter are available at http://www.prenhall.com/floyd

INTRODUCTION

In this chapter, several types of combinational logic circuits are introduced including adders, comparators, decoders, encoders, code converters, multiplexers (data selectors), demultiplexers, and parity generators/checkers. Examples of fixed-function IC devices are included. Each device introduced may also be available in other logic families.

FIXED-FUNCTION LOGIC DEVICES

74XX42	74XX47	74XX85
74XX138	74XX139	74XX147
74XX148	74XX151	74XX154
74XX157	74XX280	74XX283

SYSTEM APPLICATION ACTIVITY PREVIEW

The System Application Activity illustrates concepts from this chapter and deals with one portion of a traffic signal control system. The system applications in Chapters 6, 7, and 8 focus on various parts of the traffic signal control system. This system controls the traffic signal at the intersection of a busy street and a lightly traveled side street. The system includes a combinational logic section to which the topics in this chapter apply, a timing circuit section to which Chapter 7 applies, and a sequential logic section to which Chapter 8 applies. The Multisim activities are optional.

OPTION INTEGRATED CIRCUIT TECHNOLOGIES

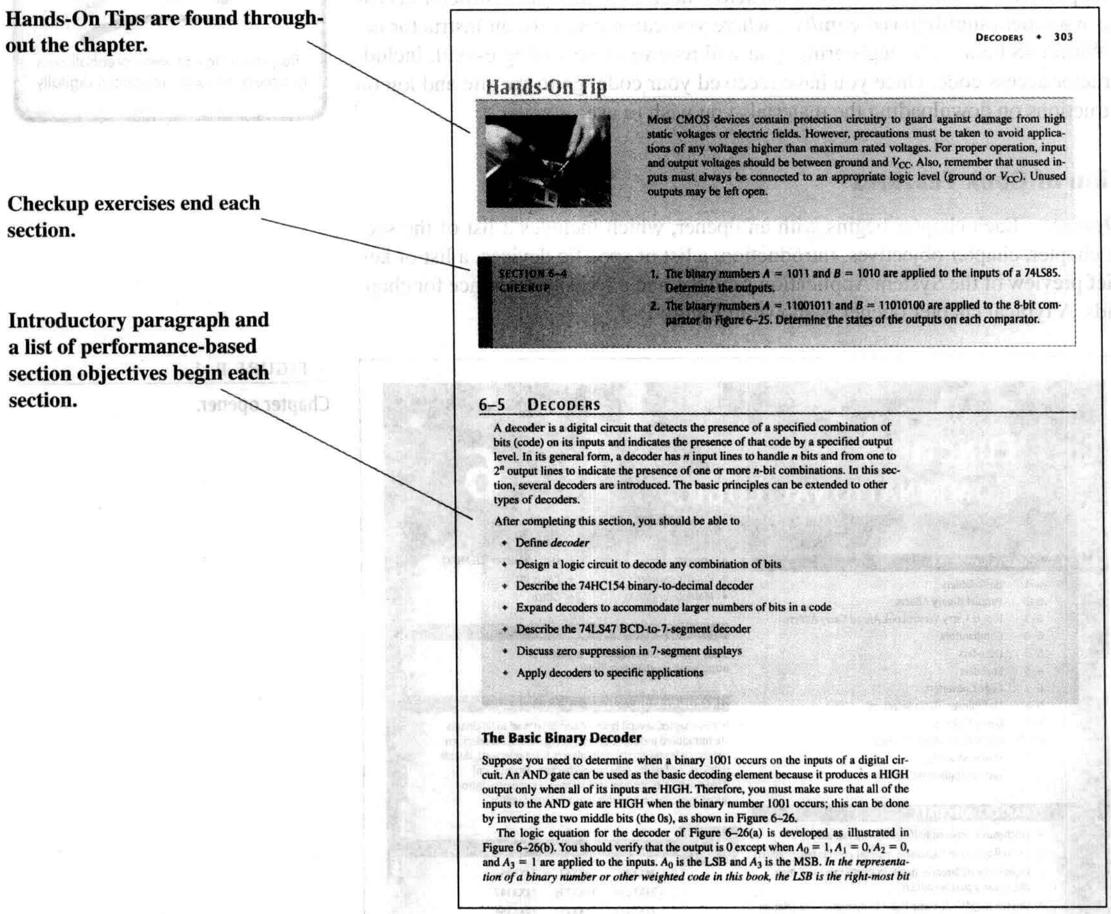
All or parts of Chapter 14 may be introduced at this point or at other appropriate points later in the text.

◄ **FIGURE P–1**

Chapter opener.

Section Opener Each section in a chapter begins with a brief introduction that includes a general overview and section objectives. An illustration is shown in Figure P–2.

Section Checkup Each section ends with a review consisting of questions or exercises that emphasize the main concepts presented in the section. This feature is shown in Figure P–2. Answers to the Section Checkups are at the end of the chapter.



DECODERS ♦ 303

Hands-On Tip

Most CMOS devices contain protection circuitry to guard against damage from high static voltages or electric fields. However, precautions must be taken to avoid applications of any voltages higher than maximum rated voltages. For proper operation, input and output voltages should be between ground and V_{CC}. Also, remember that unused inputs must always be connected to an appropriate logic level (ground or V_{CC}). Unused outputs may be left open.

SECTION 6–4 CHECKUP

1. The binary numbers $A = 1011$ and $B = 1010$ are applied to the inputs of a 74LS85. Determine the outputs.
2. The binary numbers $A = 11001011$ and $B = 11010100$ are applied to the 8-bit comparator in Figure 6–25. Determine the states of the outputs on each comparator.

6–5 DECODERS

A decoder is a digital circuit that detects the presence of a specified combination of bits (code) on its inputs and indicates the presence of that code by a specified output level. In its general form, a decoder has n input lines to handle n bits and from one to 2^n output lines to indicate the presence of one or more n-bit combinations. In this section, several decoders are introduced. The basic principles can be extended to other types of decoders.

After completing this section, you should be able to

- Define *decoder*
- Design a logic circuit to decode any combination of bits
- Describe the 74HC154 binary-to-decimal decoder
- Expand decoders to accommodate larger numbers of bits in a code
- Describe the 74LS47 BCD-to-7-segment decoder
- Discuss zero suppression in 7-segment displays
- Apply decoders to specific applications

The Basic Binary Decoder

Suppose you need to determine when a binary 1001 occurs on the inputs of a digital circuit. An AND gate can be used as the basic decoding element because it produces a HIGH output only when all of its inputs are HIGH. Therefore, you must make sure that all of the inputs to the AND gate are HIGH when the binary number 1001 occurs; this can be done by inverting the two middle bits (the 0s), as shown in Figure 6–26.

The logic equation for the decoder of Figure 6–26(a) is developed as illustrated in Figure 6–26(b). You should verify that the output is 0 except when $A_0 = 1, A_1 = 0, A_2 = 0$, and $A_3 = 1$ are applied to the inputs. A_0 is the LSB and A_3 is the MSB. *In the representation of a binary number or other weighted code in this book, the LSB is the right-most bit*

▲ **FIGURE P–2**

Section opener, section checkup, and hands-on Tip.

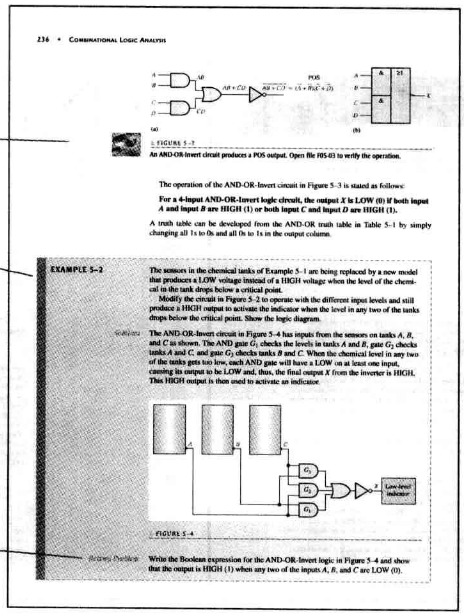
Worked Examples and Related Problems There is an abundance of worked-out examples that help to illustrate and clarify basic concepts or specific procedures. Each example ends with a *Related Problem* that reinforces or expands on the example by requiring the student to work through a problem similar to the example. A typical worked example with a *Related Problem* is shown in Figure P–3.

Troubleshooting Section Many chapters include a troubleshooting section that relates to the topics covered in the chapter and that emphasizes troubleshooting techniques and the use of test instruments and circuit simulation. A portion of a typical troubleshooting section is illustrated in Figure P–4.

A special icon indicates selected circuits that are on the CD-ROM packaged with the text.

Examples are set off from text.

Each example contains a problem related to the example.



236 • COMBINATIONAL LOGIC ANALYSIS

▲ FIGURE 5–3

An AND-OR-Invert circuit produces a POS output. Open file F05-03 to verify the operation.

The operation of the AND-OR-Invert circuit in Figure 5–3 is stated as follows:

For a 4-input AND-OR-Invert logic circuit, the output *X* is LOW (0) if both input *A* and input *B* are HIGH (1) or both input *C* and input *D* are HIGH (1).

A truth table can be developed from the AND-OR truth table in Table 5–1 by simply changing all 1s to 0s and all 0s to 1s in the output column.

EXAMPLE 5–2

The sensors in the chemical tanks of Example 5–1 are being replaced by a new model that produces a LOW voltage instead of a HIGH voltage when the level of the chemical in the tank drops below a critical point.

Modify the circuit in Figure 5–2 to operate with the different input levels and still produce a HIGH output to activate the indicator when the level in any two of the tanks drops below the critical point. Show the logic diagram.

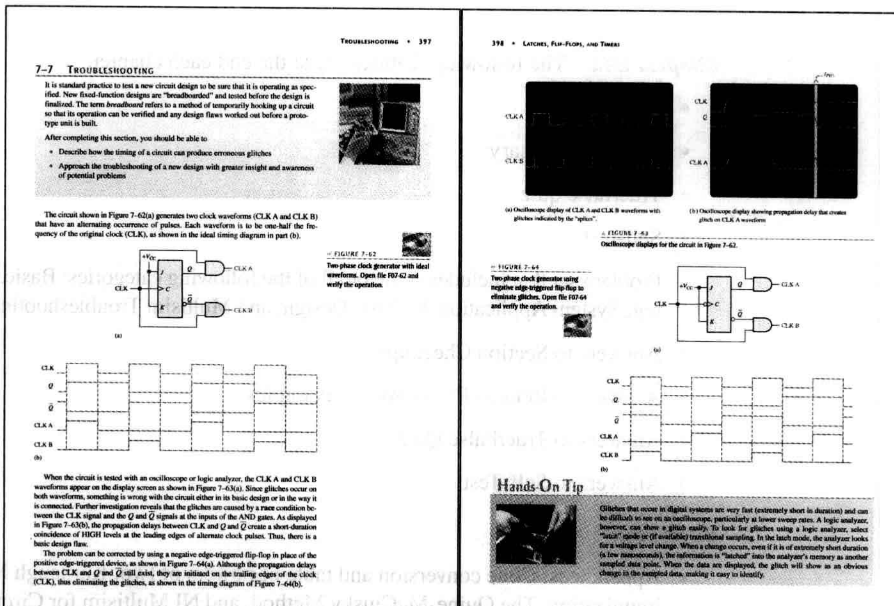
Solution The AND-OR-Invert circuit in Figure 5–4 has inputs from the sensors on tanks *A*, *B*, and *C* as shown. The AND gate G_1 checks the levels in tanks *A* and *B*, gate G_2 checks tanks *A* and *C*, and gate G_3 checks tanks *B* and *C*. When the chemical level in any two of the tanks gets too low, each AND gate will have a LOW on at least one input, causing its output to be LOW and, thus, the final output *X* from the inverter is HIGH. This HIGH output is then used to activate an indicator.

▲ FIGURE 5–4

Related Problem Write the Boolean expression for the AND-OR-Invert logic in Figure 5–4 and show that the output is HIGH (1) when any two of the inputs *A*, *B*, and *C* are LOW (0).

◄ **FIGURE P–3**

An example and related problem.



TROUBLESHOOTING • 397

7–7 TROUBLESHOOTING

It is standard practice to test a new circuit design to be sure that it is operating as specified. New fixed-function designs are "breadboarded" and tested before the design is finalized. The term *breadboard* refers to a method of temporarily hooking up a circuit so that its operation can be verified and any design flaws worked out before a prototype unit is built.

After completing this section, you should be able to

- Describe how the timing of a circuit can produce erroneous glitches
- Approach the troubleshooting of a new design with greater insight and awareness of potential problems

The circuit shown in Figure 7–62(a) generates two clock waveforms (CLK A and CLK B) that have an alternating occurrence of pulses. Each waveform is to be one-half the frequency of the original clock (CLK), as shown in the ideal timing diagram in part (b).

◄ FIGURE 7–62

Two-phase clock generator with ideal waveforms. Open file F07-62 and verify the operation.

When the circuit is tested with an oscilloscope or logic analyzer, the CLK A and CLK B waveforms appear on the display screen as shown in Figure 7–63(a). Since glitches occur on both waveforms, something is wrong with the circuit either in its basic design or in the way it is connected. Further investigation reveals that the glitches are caused by a **race** condition between the CLK signal and the *Q* and $\bar{Q}$ signals at the inputs of the AND gates. As displayed in Figure 7–63(b), the propagation delays between CLK and *Q* and $\bar{Q}$ create a short-duration coincidence of HIGH levels at the leading edges of alternate clock pulses. Thus, there is a basic design flaw.

The problem can be corrected by using a negative edge-triggered flip-flop in place of the positive edge-triggered device, as shown in Figure 7–64(a). Although the propagation delays between CLK and *Q* and $\bar{Q}$ still exist, they are initiated on the trailing edges of the clock (CLK), thus eliminating the glitches, as shown in the timing diagram of Figure 7–64(b).

398 • LATCHES, FLIP-FLOPS, AND TIMERS

(a) Oscilloscope display of CLK A and CLK B waveforms with glitches indicated by the "spikes".

(b) Oscilloscope display showing propagation delay that creates glitch on CLK A waveform

▲ FIGURE 7–63

Oscilloscope displays for the circuit in Figure 7–62.

◄ FIGURE 7–64

Two-phase clock generator using negative edge-triggered flip-flop to eliminate glitches. Open file F07-64 and verify the operation.

Hands-On Tip

Glitches that occur in digital systems are very fast (extremely short in duration) and can be difficult to see on an oscilloscope, particularly at lower sweep rates. A logic analyzer, however, can show a glitch easily. To look for glitches using a logic analyzer, select "latch" mode or (if available) transitional sampling. In the latch mode, the analyzer looks for a voltage level change. When a change occurs, even if it is of extremely short duration (a few nanoseconds), the information is "latched" into the analyzer's memory as another sampled data point. When the data are displayed, the glitch will show as an obvious change in the sampled data, making it easy to identify.

▲ **FIGURE P–4**

Representative pages from a portion of a typical Troubleshooting section.

System Application Activity Appearing at the end of many chapters, this feature presents a practical application of the concepts and procedures covered in the chapter. This feature presents a "real-world" system in which analysis, troubleshooting, design, and simulation are included. Some System Application Activities are limited to a single chapter, whereas others extend over two or more chapters. A reference to a related laboratory experiment is provided. Figure P–5 shows a portion of a typical System Application Activity.

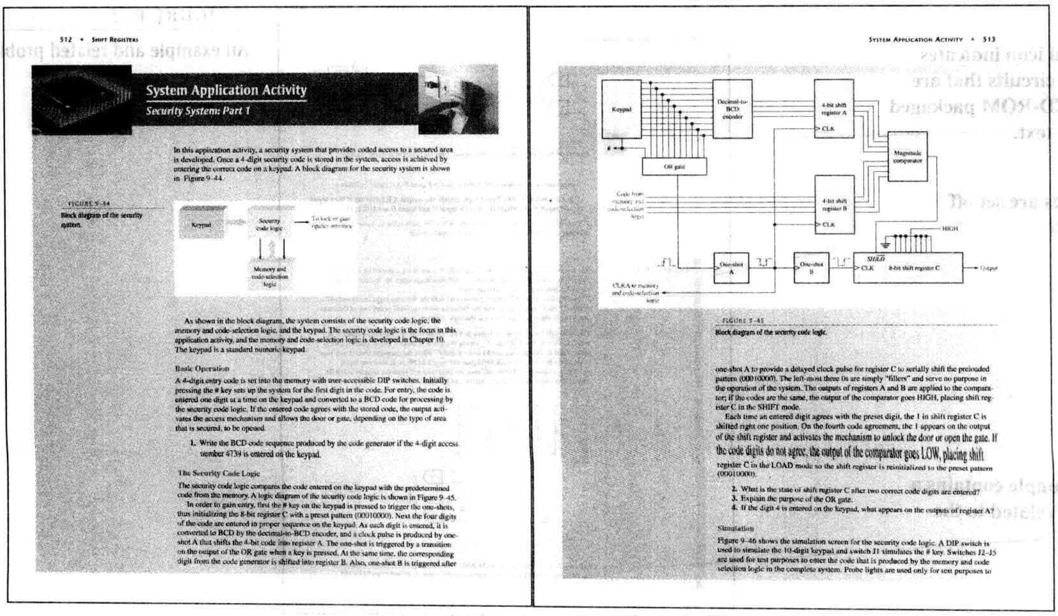


FIGURE P–5

Representative pages from a typical System Application Activity.

Chapter End The following features are at the end each chapter.

- Summary
- Key term glossary
- True/false quiz
- Self-test
- Problem set that includes some or all of the following categories: Basic, Troubleshooting, System Application Activity, Design, and Multisim Troubleshooting Practice.
- Answers to Section Checkups
- Answers to Related Problems for Examples
- Answers to True/False Quiz
- Answers to Self-Test

Book End

- Appendices: Code conversion and table of powers of two, Karnaugh Map POS Minimunization, The Quine-McClusky Method, and NI Multisim for Circuit Simulation.
- Comprehensive glossary
- Answers to selected odd-numbered problems
- Index

To the Student

Digital technology pervades almost everything in our daily lives. For example, cell phones and other types of wireless communications, television, radio, process controls, automotive electronics, consumer electronics, aircraft navigation—to name only a few applications—depend heavily on digital electronics.

A strong grounding in the fundamentals of digital technology will prepare you for the highly skilled jobs of the future. The single most important thing you can do is to understand the core fundamentals. From there you can go anywhere.

In addition, programmable logic is important in many applications and that topic is introduced in this book. Of course, efficient troubleshooting is a skill that is also widely sought after. Troubleshooting and testing methods from traditional prototype testing to manufacturing techniques such as bed-of-nails, flying probe, and boundary scan are covered.

User's Guide for Instructors

Generally, time limitations or program emphasis determines the topics to be covered in a course. It is not uncommon to omit or condense topics or to alter the sequence of certain topics in order to customize the material for a particular course. The author recognizes this and has designed this textbook specifically to provide great flexibility in topic coverage.

Certain topics are organized in separate sections or features such that if they are omitted, the rest of the coverage is not affected. Also, if these topics are included, they flow seamlessly with the rest of the coverage. The book is organized around a core of fundamental topics that are, for the most part, essential in any digital course. Around this core, there are other topics that can be included or omitted, depending on the course emphasis or other factors. Figure P–6 illustrates this concept.

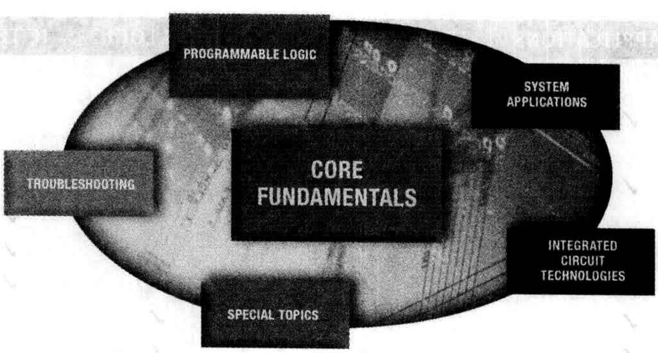


FIGURE P–6

- ◆ ***Core Fundamentals*** The fundamental topics of digital technology should, for the most part, be covered in all programs. Linked to the core are several "satellite" topics that may be considered for inclusion or omission, depending on your course goals. Any block surrounding the core can be omitted without affecting the core fundamentals.
- ◆ ***Programmable Logic*** Although it is an important topic, programmable logic can be omitted, although it is recommended that you cover this topic if at all possible. You can cover as little or as much as you consider appropriate for your program.
- ◆ ***Troubleshooting*** Troubleshooting sections appear in many chapters.
- ◆ ***System Application Activities*** System applications appear in many chapters.
- ◆ ***Integrated Circuit Technologies*** Some or all of the topics in Chapter 14 can be covered at selected points if you wish to discuss details of the circuitry that make up

digital integrated circuits. Chapter 14 can be omitted without any impact on the rest of the book.

◆ ***Special Topics*** These topics are *Signal Interfacing and Processing* and *Computer Concepts* in Chapters 12 and 13, respectively. These are special topics that may not be essential for your course. They can omitted without any impact on the rest of the book.

Also, within each block in Figure P–6 you can choose to omit or deemphasize some topics because of time constraints or other priorities. For example, in the core fundamentals, cyclic redundancy code, carry look-ahead adders, or sequential logic design could possibly be omitted. Additionally, any or all of Multisim features throughout the book can be treated as optional.

Customizing the Table of Contents You can take any one of several paths through *Digital Fundamentals*, Tenth Edition, depending on the goals of your particular program. Whether you choose a minimal coverage of only core fundamentals, a full-blown coverage of all the topics, or anything in between, this book can be adapted to your needs. The Table of Contents following this preface is color-coded to match the blocks in Figure P–6. This allows you to identify topics for omission or inclusion when customizing your course.

Examples of some of the options for use of *Digital Fundamentals*, Tenth Edition, are shown in Table P–1 in terms of the color-coded topics in Figure P–6. Many other combinations of topics are possible, too, including partial coverage of some topics.

▼ TABLE P–1

Examples of topic combinations.

TOPICS	CORE	SYSTEM APPLICATIONS	TROUBLESHOOTING	PROG LOGIC	IC TECH	SPECIAL
Option 1	✓					
Option 2	✓	✓				
Option 3	✓		✓			
Option 4	✓	✓	✓			
Option 5	✓				✓	
Option 6	✓	✓			✓	
Option 7	✓		✓		✓	
Option 8	✓	✓		✓		
Option 9	✓	✓	✓	✓		
Option 10	✓	✓	✓	✓	✓	✓

Acknowledgments

This revision of *Digital Fundamentals* has been made possible by the work and skills of many people. I think that we have accomplished what we set out to do, and that was to further improve an already very successful textbook and make it even more useful to the student and instructor by presenting not only basics but also up-to-date and leading-edge technology.

Those at Prentice Hall who have, as always, contributed a great amount of time, talent, and effort to move this project through its many phases in order to produce the book as you see it include, but are not limited to, Rex Davidson, Chris Reed, and Wyatt Morris. Lois Porter has done another fantastic job of manuscript editing. Dave Buchla read the entire manuscript and provided many useful ideas for improving this edition and revised the lab manual for close coordination with the text. Dave also developed an excellent series of

PowerPoint slides closely linked to the text on a chapter-by-chapter basis and intented to be used in conjunction with the text. Gary Snyder revised and updated the Multisim circuit files to versions 9 and 10. My thanks and appreciation go to all of these people and others who were directly involved in the project.

In the revision of this and all textbooks, I depend on expert input from many users as well as nonusers. My sincere thanks to the following reviewers who submitted many valuable suggestions and provided lots of constructive criticism: Bill Diong, Texas Christian University; Bruce Hesher, Brevard Community College; William H. Murray, Broome Community College; and Tristan Tayag, Texas Christian University.

I also want to thank all of the members of the Prentice Hall sales force whose efforts have helped make this text available to a large number of users. In addition, I am grateful to all of you who have adopted this text for your classes or for your own use. Without you we would not be in business. I hope that you find this edition of *Digital Fundamentals* to be even better than earlier editions and that it will continue to be a valuable learning tool and reference for the student.

Tom Floyd

CONTENTS

INTRODUCTORY CONCEPTS

1

CHAPTER OUTLINE

CHAPTER OBJECTIVES

- Explain the basic differences between digital and analog quantities
- Show how voltage levels are used to represent digital quantities
- Describe various parameters of a pulse waveform such as rise time, fall time, pulse width, frequency, period, and duty cycle
- Explain the basic logic operations of NOT, AND, and OR
- Describe several types of logic functions and explain their application in an example system
- Identify fixed-function digital integrated circuits according to their complexity and the type of circuit packaging
- Identify pin numbers on integrated circuit packages
- Recognize various instruments and understand how they are used in measurement and troubleshooting digital circuits and systems
- Describe programmable logic, discuss the various types, and describe how PLDs are programmed

KEY TERMS

Key terms are in order of appearance in the chapter.

- Analog
- Digital
- Binary
- Bit
- Pulse
- Duty cycle
- Clock
- Timing diagram
- Data
- Serial
- Parallel
- Logic
- Input
- Output
- Gate
- NOT
- Inverter
- AND
- OR
- Integrated circuit (IC)
- Fixed-function logic
- Troubleshooting
- Programmable logic
- SPLD
- CPLD
- FPGA
- Compiler

VISIT THE COMPANION WEBSITE

Study aids for this chapter are available at http://www.prenhall.com/floyd

INTRODUCTION

The term *digital* is derived from the way operations are performed, by counting digits. For many years, applications of digital electronics were confined to computer systems. Today, digital technology is applied in a wide range of areas in addition to computers. Such applications as television, communications systems, radar, navigation and guidance systems, military systems, medical instrumentation, industrial process control, and consumer electronics use digital techniques. Over the years digital technology has progressed from vacuum-tube circuits to discrete transistors to complex integrated circuits, some of which contain millions of transistors.

This chapter introduces you to digital electronics and provides a broad overview of many important concepts, components, and tools.

1–1 DIGITAL AND ANALOG QUANTITIES

Electronic circuits can be divided into two broad categories, digital and analog. Digital electronics involves quantities with discrete values, and analog electronics involves quantities with continuous values. Although you will be studying digital fundamentals in this book, you should also know something about analog because many applications require both; and interfacing between analog and digital is important.

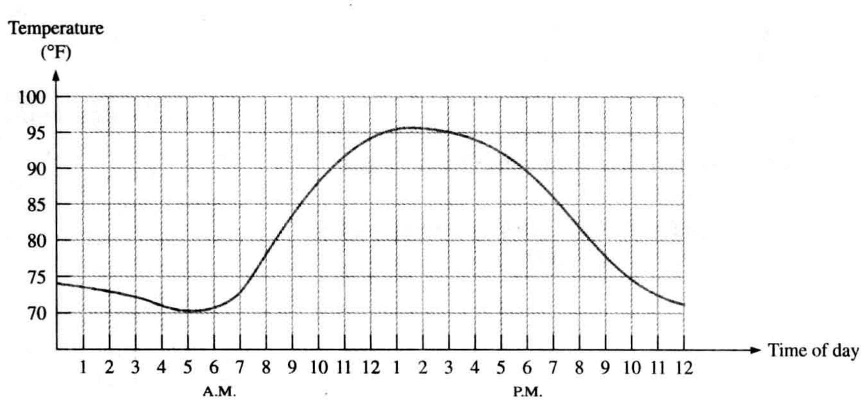
After completing this section, you should be able to

- Define *analog*
- Define *digital*
- Explain the difference between digital and analog quantities
- State the advantages of digital over analog
- Give examples of how digital and analog quantities are used in electronics

An **analog*** quantity is one having continuous values. A **digital** quantity is one having a discrete set of values. Most things that can be measured quantitatively occur in nature in analog form. For example, the air temperature changes over a continuous range of values. During a given day, the temperature does not go from, say, 70° to 71° instantaneously; it takes on all the infinite values in between. If you graphed the temperature on a typical summer day, you would have a smooth, continuous curve similar to the curve in Figure 1–1. Other examples of analog quantities are time, pressure, distance, and sound.

FIGURE 1–1

Graph of an analog quantity (temperature versus time).



Rather than graphing the temperature on a continuous basis, suppose you just take a temperature reading every hour. Now you have sampled values representing the temperature at discrete points in time (every hour) over a 24-hour period, as indicated in Figure 1–2. You have effectively converted an analog quantity to a form that can now be digitized by representing each sampled value by a digital code. It is important to realize that Figure 1–2 itself is not the digital representation of the analog quantity.

*All bold terms are important and are defined in the end-of-book glossary. The blue bold terms are key terms and are included in a Key Term glossary at the end of each chapter.

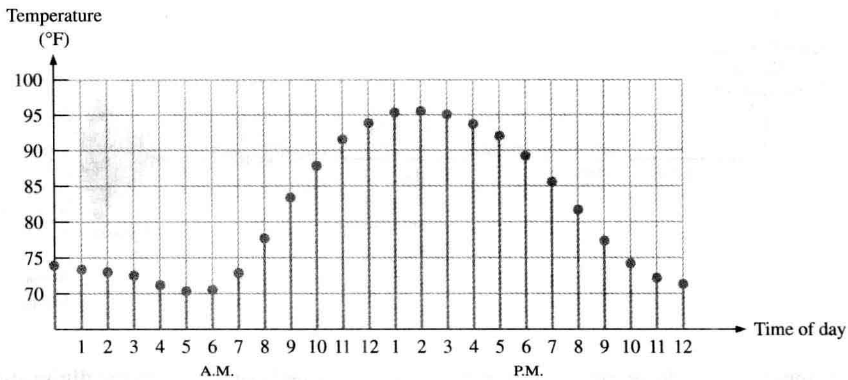


FIGURE 1–2

Sampled-value representation (quantization) of the analog quantity in Figure 1–1. Each value represented by a dot can be digitized by representing it as a digital code that consists of a series of 1s and 0s.

The Digital Advantage Digital representation has certain advantages over analog representation in electronics applications. For one thing, digital data can be processed and transmitted more efficiently and reliably than analog data. Also, digital data has a great advantage when storage is necessary. For example, music when converted to digital form can be stored more compactly and reproduced with greater accuracy and clarity than is possible when it is in analog form. Noise (unwanted voltage fluctuations) does not affect digital data nearly as much as it does analog signals.

An Analog Electronic System

A public address system, used to amplify sound so that it can be heard by a large audience, is one simple example of an application of analog electronics. The basic diagram in Figure 1–3 illustrates that sound waves, which are analog in nature, are picked up by a microphone and converted to a small analog voltage called the audio signal. This voltage varies continuously as the volume and frequency of the sound changes and is applied to the input of a linear amplifier. The output of the amplifier, which is an increased reproduction of input voltage, goes to the speaker(s). The speaker changes the amplified audio signal back to sound waves that have a much greater volume than the original sound waves picked up by the microphone.

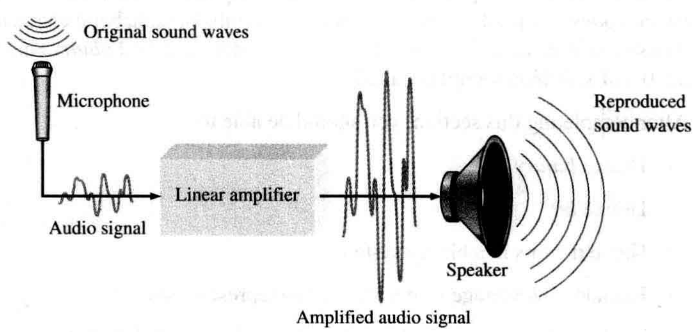


FIGURE 1–3

A basic audio public address system.

A System Using Digital and Analog Methods

The compact disk (CD) player is an example of a system in which both digital and analog circuits are used. The simplified block diagram in Figure 1–4 illustrates the basic principle. Music in digital form is stored on the compact disk. A laser diode optical