

David A. Hodges
Horace G. Jackson
Resve A. Saleh

Third Edition

Analysis and Design of Digital Integrated Circuits

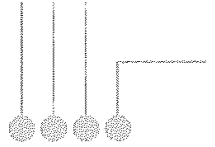
In Deep Submicron Technology

USEFUL DESIGN PARAMETERS (simplified)

Name	Symbol	0.18 μm		0.13 μm		Units
		NMOS	PMOS	NMOS	PMOS	
Channel Length (rounded for convenience)	L	200	200	100	100	nm
Supply Voltage	V_{DD}	1.8	1.8	1.2	1.2	V
Oxide Thickness	t_{ox}	35	35	22	22	Å
Oxide Capacitance	C_{ox}	1.0	1.0	1.6	1.6	$\mu\text{F}/\text{cm}^2$
Threshold Voltage	V_{T0}	0.5	-0.5	0.4	-0.4	V
Body-Effect Term	γ	0.3	0.3	0.2	0.2	$\text{V}^{1/2}$
Fermi Potential	$2 \phi_F $	0.84	0.84	0.88	0.88	V
Junction Capacitance Coefficient	C_{j0}	1.6	1.6	1.6	1.6	$\text{fF}/\mu\text{m}^2$
Built-In Junction Potential	ϕ_B	0.9	0.9	1.0	1.0	V
Grading Coefficient	m	0.5	0.5	0.5	0.5	—
Nominal Mobility (low vertical field)	μ_0	540	180	540	180	$\text{cm}^2/\text{V}\cdot\text{s}$
Effective Mobility (high vertical field)	μ_e	270	70	270	70	$\text{cm}^2/\text{V}\cdot\text{s}$
Critical Field	E_c	6×10^4	24×10^4	6×10^4	24×10^4	V/cm
Critical Field $\times L$	$E_c L$	1.2	4.8	0.6	2.4	V
Effective Resistance	R_{eff}	12.5	30	12.5	30	$\text{k}\Omega/\square$

Name	Symbol	Value	Units
Gate Capacitance Coefficient	C_g	2	$\text{fF}/\mu\text{m}$
Self Capacitance Coefficient	C_{eff}	1	$\text{fF}/\mu\text{m}$
Wire Capacitance Coefficient	C_w	0.1–0.25	$\text{fF}/\mu\text{m}$
Al Wire Resistance	$R_{\square}$	25–60	$\text{m}\Omega/\square$
Cu Wire Resistance	$R_{\square}$	20–40	$\text{m}\Omega/\square$
Wire Inductance	L_{eff}	40–50	$\text{pH}/\mu\text{m}$

UNIVERSITY OF MICHIGAN LIBRARIES



Analysis and Design of Digital Integrated Circuits

Electronics and VLSI Circuits Related Titles

Senior Consulting Editor

Stephen W. Director, *University of Michigan, Ann Arbor*

Consulting Editor

Richard C. Jaeger, *Auburn University*

DeMicheli: *Synthesis and Optimization of Digital Circuits*

Franco: *Design with Operational Amplifiers and Analog Integrated Circuits*

Hodges, Jackson, and Saleh: *Analysis and Design of Digital Integrated Circuits*

Kang and Leblebici: *CMOS Digital Integrated Circuits Analysis and Design*

Kasap: *Principles of Electrical Engineering Materials and Devices*

Kovacs: *Micromachined Transducers Sourcebook*

Jaeger and Blalock: *Microelectronic Circuit Design*

Neamen: *Electronic Circuit Analysis and Design*

Neamen: *Semiconductor Physics and Devices*

Razavi: *Design of Integrated Circuits for Optical Communications*

Razavi: *Design of Analog CMOS Integrated Circuits*



Analysis and Design of Digital Integrated Circuits

In Deep Submicron Technology

T H I R D E D I T I O N

David A. Hodges

University of California, Berkeley

Horace G. Jackson

University of California, Berkeley

Resve A. Saleh

University of British Columbia



Higher Education

Boston Burr Ridge, IL Dubuque, IA Madison, WI New York San Francisco St. Louis
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

The McGraw-Hill Companies,



Higher Education

ANALYSIS AND DESIGN OF DIGITAL INTEGRATED CIRCUITS: IN DEEP SUBMICRON TECHNOLOGY,
THIRD EDITION

Published by McGraw-Hill, a business unit of The McGraw-Hill Companies, Inc., 1221 Avenue of the Americas, New York, NY 10020. Copyright © 2004, 1988 by The McGraw-Hill Companies, Inc. All rights reserved. No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written consent of The McGraw-Hill Companies, Inc., including, but not limited to, in any network or other electronic storage or transmission, or broadcast for distance learning.

Some ancillaries, including electronic and print components, may not be available to customers outside the United States.

This book is printed on acid-free paper.

1 2 3 4 5 6 7 8 9 0 DOC/DOC 0 9 8 7 6 5 4 3

ISBN 0-07-228365-3

Publisher: *Elizabeth A. James*
Senior sponsoring editor: *Carlise Paulson*
Senior developmental editor: *Emily J. Lupash*
Marketing manager: *Dawn R. Bercier*
Project manager: *Joyce Watters*
Senior production supervisor: *Laura Fuller*
Lead media project manager: *Audrey A. Reiter*
Senior media technology producer: *Phillip Meek*
Senior coordinator of freelance design: *Michelle D. Whitaker*
Cover/interior designer: *Argosy Publishing*
Cover image: © *X-Architecture Routing Illustration Courtesy of Cadence Design Systems, Inc.*
Lead photo research coordinator: *Carrie K. Burger*
Compositor: *Lachina Publishing Services*
Typeface: *10.5/12 Times Roman*
Printer: *R. R. Donnelley Crawfordsville, IN*

Library of Congress Cataloging-in-Publication Data

Hodges, David A., 1937–

Analysis and design of digital integrated circuits : in deep submicron technology. David A. Hodges, Horace G. Jackson, Resve Saleh.— 3rd ed

p. cm.—(McGraw-Hill series in electrical engineering)

Includes index.

ISBN 0-07-228365-3

1. Digital integrated circuits—Design and construction.
 2. Integrated circuits—Ultra large scale integration.
 3. Electric circuit analysis.
- I. Jackson, Horace G. II. Saleh, Resve. III. Title. IV. Series.

TK7874.65.H64 2004
621.3815—dc21

2003051006
CIP

www.mhhe.com

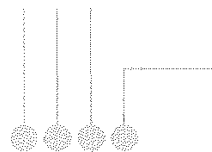
About the Authors

David A. Hodges is the Daniel M. Tellep Distinguished Professor of Engineering Emeritus at the University of California, Berkeley. He earned the B.E.E. degree at Cornell University and the M.S. and Ph.D. degrees at the University of California, Berkeley. In 1970 he joined the faculty in Electrical Engineering and Computer Sciences at UC Berkeley. Following a year as Chair of the EECS Department, he served as Dean of the College of Engineering from July 1990 through June 1996. Professor Hodges was the winner of the 1997 IEEE Education Medal and the 1999 ASEE Benjamin Garver Lamme Award. He was the founding editor of the *IEEE Transactions on Semiconductor Manufacturing*, and a past editor of the *IEEE Journal of Solid-State Circuits*. Professor Hodges is a Fellow of the IEEE and a Member of the National Academy of Engineering. He is a Director of Silicon Image, Inc. and a former Director of Mentor Graphics.

Horace G. Jackson was born and educated in England. In 1947 he went to Canada, where until 1956 he was engaged in nuclear physics research at the Chalk River Laboratories of Atomic Energy of Canada. In 1956 he joined the Lawrence Berkeley Laboratory of the University of California, where he worked as a Senior Staff Scientist, as well as serving as a Resident Lecturer in the Department of Electrical Engineering and Computer Sciences until his retirement. He has published widely in nuclear science and electronic engineering journals and is coauthor on two books. His research interests are in high frequency analog and high-speed digital integrated circuits.

Resve A. Saleh currently holds the NSERC/PMC-Sierra Chair in the Department of Electrical and Computer Engineering at the University of British Columbia. Dr. Saleh obtained his M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley. He received the prestigious Presidential Young Investigator Award in 1990 from the National Science Foundation in the United States. Dr. Saleh has published two books and over 60 journal articles and conference papers. He was an Associate Editor of IEEE Transactions on Computer-Aided Design and technical program chair of the Custom Integrated Circuits Conference. He has also served on numerous program committees. He is a founder and former

Chairman of Simplex Solutions (Sunnyvale, CA). Prior to starting Simplex, Dr. Saleh spent nine years as a Professor in the Department of Electrical and Computer Engineering at the University of Illinois in Urbana. He also spent one year at Stanford University on a sabbatical leave. Before embarking on his academic career, Dr. Saleh worked for Mitel Corporation in Ottawa, Canada, Toshiba Corporation in Japan, Tektronix in Beaverton, Oregon, and Nortel in Ottawa, Canada. At present, he is directing research at UBC in the System-on-Chip Research Laboratory.



Contents

Preface to Third Edition	xiii
1 Deep Submicron Digital IC Design	1
1.1 Introduction	1
1.2 Brief History of IC Industry	3
1.3 Review of Digital Logic Gate Design	6
1.3.1 Basic Logic Functions	6
1.3.2 Implementation of Logic Circuits	9
1.3.3 Definition of Noise Margin	11
1.3.4 Definition of Transient Characteristics	12
1.3.5 Power Estimation	14
1.4 Digital Integrated Circuit Design	15
1.4.1 MOS Transistor Structure and Operation	16
1.4.2 CMOS Versus NMOS	17
1.4.3 Deep Submicron Interconnect	19
1.5 Computer-Aided Design of Digital Circuits	24
1.5.1 Circuit Simulation and Analysis	24
*1.6 The Challenges Ahead	26
1.7 Summary	31
2 MOS Transistors	35
2.1 Introduction	35
2.2 Structure and Operation of the MOS Transistor	37
2.3 Threshold Voltage of the MOS Transistor	41
2.4 First-Order Current-Voltage Characteristics	52
2.5 Derivation of Velocity-Saturated Current Equations	57
2.5.1 Effect of High Fields	57
2.5.2 Current Equations for Velocity-Saturated Devices	61
*2.6 Alpha-Power Law Model	66
2.7 Subthreshold Conduction	68

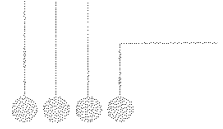
2.8	Capacitances of the MOS Transistor	70
2.8.1	Thin-Oxide Capacitance	71
2.8.2	<i>pn</i> Junction Capacitance	73
2.8.3	Overlap Capacitance	79
2.9	Summary	81
3	Fabrication, Layout, and Simulation	89
3.1	Introduction	89
3.2	IC Fabrication Technology	90
3.2.1	Overview of IC Fabrication Process	90
3.2.2	IC Photolithographic Process	92
3.2.3	Making Transistors	93
3.2.4	Making Wires	97
3.2.5	Wire Capacitance and Resistance	100
3.3	Layout Basics	104
3.4	Modeling the MOS Transistor for Circuit Simulation	107
3.4.1	MOS Models in SPICE	107
3.4.2	Specifying MOS Transistors	108
3.5	SPICE MOS LEVEL 1 Device Model	111
3.5.1	Extraction of Parameters for MOS LEVEL 1	113
*3.6	BSIM3 Model	115
3.6.1	Binning Process in BSIM3	115
3.6.2	Short-Channel Threshold Voltage	116
3.6.3	Mobility Model	119
3.6.4	Linear and Saturation Regions	120
3.6.5	Subthreshold Current	122
3.6.6	Capacitance Models	123
3.6.7	Source/Drain Resistance	124
*3.7	Additional Effects in MOS Transistors	125
3.7.1	Parameter Variations in Production	125
3.7.2	Temperature Effects	125
3.7.3	Supply Variations	127
3.7.4	Voltage Limitations	128
3.7.5	CMOS Latch-up	128
*3.8	Silicon-on-Insulator (SOI) Technology	130
*3.9	SPICE Model Summary	132
4	MOS Inverter Circuits	143
4.1	Introduction	143
4.2	Voltage Transfer Characteristics	144
4.3	Noise Margin Definitions	147
4.3.1	Single-Source Noise Margin (SSNM)	148
4.3.2	Multiple-Source Noise Margin (MSNM)	150
4.4	Resistive-Load Inverter Design	153

4.5	NMOS Transistors as Load Devices	162
4.5.1	Saturated Enhancement Load	162
4.5.2	Linear Enhancement Load	166
4.6	Complementary MOS (CMOS) Inverters	168
4.6.1	DC Analysis of CMOS Inverter	168
4.6.2	Layout Design of CMOS Inverter	176
4.7	Pseudo-NMOS Inverters	178
4.8	Sizing Inverters	181
4.9	Tristate Inverters	184
4.10	Summary	185
5	Static MOS Gate Circuits	195
5.1	Introduction	195
5.2	CMOS Gate Circuits	197
5.2.1	Basic CMOS Gate Sizing	198
5.2.2	Fanin and Fanout Considerations	202
5.2.3	Voltage Transfer Characteristics (VTC) of CMOS Gates	205
5.3	Complex CMOS Gates	209
5.4	XOR and XNOR Gates	212
5.5	Multiplexer Circuits	214
5.6	Flip-Flops and Latches	214
5.6.1	Basic Bistable Circuit	215
5.6.2	SR Latch	216
5.6.3	JK Flip-Flop	220
5.6.4	JK Master-Slave Flip-Flop	221
5.6.5	JK Edge-Triggered Flip-Flop	222
5.7	D Flip-Flops and Latches	223
5.8	Power Dissipation in CMOS Gates	227
5.8.1	Dynamic (Switching) Power	228
5.8.2	Static (Standby) Power	235
5.8.3	Complete Power Equation	237
5.9	Power and Delay Tradeoffs	238
5.10	Summary	241
6	High-Speed CMOS Logic Design	249
6.1	Introduction	249
6.2	Switching Time Analysis	251
6.2.1	Gate Sizing Revisited—Velocity Saturation Effects	255
6.3	Detailed Load Capacitance Calculation	257
6.3.1	Fanout Gate Capacitance	258
6.3.2	Self-Capacitance Calculation	260
6.3.3	Wire Capacitance	267

6.4	Improving Delay Calculation with Input Slope	267
6.5	Gate Sizing for Optimal Path Delay	276
6.5.1	Optimal Delay Problem	276
6.5.2	Inverter Chain Delay Optimization—FO4 Delay	277
6.5.3	Optimizing Paths with NANDs and NORs	283
6.6	Optimizing Paths with Logical Effort	286
6.6.1	Derivation of Logical Effort	286
6.6.2	Understanding Logical Effort	292
6.6.3	Branching Effort and Sideloads	297
6.7	Summary	301
7	Transfer Gate and Dynamic Logic Design	309
7.1	Introduction	309
7.2	Basic Concepts	310
7.2.1	Pass Transistors	310
7.2.2	Capacitive Feedthrough	313
7.2.3	Charge Sharing	316
7.2.4	Other Sources of Charge Loss	318
7.3	CMOS Transmission Gate Logic	318
7.3.1	Multiplexers Using CMOS Transfer Gates	320
7.3.2	CMOS Transmission Gate Delays	325
7.3.3	Logical Effort with CMOS Transmission Gates	331
7.4	Dynamic D-Latches and D Flip-Flops	333
7.5	Domino Logic	336
7.5.1	Logical Effort for Domino Gates	342
7.5.2	Limitations of Domino Logic	343
7.5.3	Dual-Rail (Differential) Domino Logic	346
7.5.4	Self-Resetting Circuits	349
7.6	Summary	349
8	Semiconductor Memory Design	359
8.1	Introduction	359
8.1.1	Memory Organization	360
8.1.2	Types of Memory	362
8.1.3	Memory Timing Parameters	363
8.2	MOS Decoders	364
8.3	Static RAM Cell Design	368
8.3.1	Static Memory Operation	368
8.3.2	Read Operation	371
8.3.3	Write Operation	374
8.3.4	SRAM Cell Layout	376
8.4	SRAM Column I/O Circuitry	377
8.4.1	Column Pull-Ups	378

8.4.2	Column Selection	380
8.4.3	Write Circuitry	382
8.4.4	Read Circuitry	382
8.5	Memory Architecture	390
8.6	Summary	393
9	Additional Topics in Memory Design	399
9.1	Introduction	399
*9.2	Content-Addressable Memories (CAMs)	400
*9.3	Field-Programmable Gate Array	407
9.4	Dynamic Read-Write Memories	413
9.4.1	Three-Transistor Dynamic Cell	414
9.4.2	One-Transistor Dynamic Cell	415
9.4.3	External Characteristics of Dynamic RAMs	419
9.5	Read-Only Memories	421
9.5.1	MOS ROM Cell Arrays	421
9.6	EPROMs and E ² PROMs	425
*9.7	Flash Memory	432
*9.8	FRAMs	435
9.9	Summary	436
10	Interconnect Design	441
10.1	Introduction	441
10.2	Interconnect RC Delays	444
10.2.1	Wire Resistance	444
10.2.2	Elmore Delay Calculation	446
10.2.3	RC Delay in Long Wires	449
10.3	Buffer Insertion for Very Long Wires	453
10.4	Interconnect Coupling Capacitance	457
10.4.1	Components of Coupling Capacitance	457
10.4.2	Coupling Effects on Delay	463
10.4.3	Capacitive Noise or Crosstalk	467
*10.5	Interconnect Inductance	468
*10.6	Antenna Effects	473
10.7	Summary	477
11	Power Grid and Clock Design	483
11.1	Introduction	483
11.2	Power Distribution Design	484
11.2.1	IR Drop and Ldi/dt	485
11.2.2	Electromigration	488
11.2.3	Power Routing Considerations	491

11.2.4	Decoupling Capacitance Design	493
11.2.5	Power Distribution Design Example	495
11.3	Clocking and Timing Issues	499
11.3.1	Clock Definitions and Metrics	499
11.3.2	Clock Skew	501
11.3.3	Effect of Noise on Clocks and FFs	504
11.3.4	Power Dissipation in Clocks	506
11.3.5	Clock Generation	507
11.3.6	Clock Distribution for High-Performance Designs	509
11.3.7	Example of a Clock Distribution Network	511
11.4	Phase-Locked Loops/Delay-Locked Loops	514
11.4.1	PLL Design Considerations	516
11.4.2	Clock Distribution Summary	522
Appendix A A Brief Introduction to SPICE		529
A.1	Introduction	529
A.2	Design Flow	530
A.3	Syntax	531
A.3.1	Title	532
A.3.2	Settings of Various Global Parameters	532
A.3.3	Listing of Sources and Active and Passive Elements	534
A.3.4	Analysis Statements	541
A.4	Complete SPICE Examples	545
Appendix B Bipolar Transistors and Circuits		547
B.1	The Bipolar Junction Transistor	547
B.2	The Schottky-Barrier Diode	550
B.3	BJT Model for Circuit Simulation	552
B.4	Bipolar Transistor Inverter	553
B.5	Voltage Transfer Characteristics	554
B.6	Schottky-Clamped Inverter	556
B.7	BJT Inverter Switching Times	557
B.8	Bipolar Digital Gate Circuits	558
B.9	Voltage Transfer Characteristics	558
B.10	Propagation Delay Time	561
B.11	Input Clamp Diodes	561
Index		563



Preface to the Third Edition

Motivation for a Third Edition

Since the publication of the second edition over 15 years ago, the integrated circuit (IC) industry has undergone a tremendous amount of change. The most significant trend is the emergence of CMOS as the dominant IC technology. Other technologies such as NMOS, bipolar, and GaAs, have given way to CMOS due to considerations such as power, level of integration, and cost. The tools, technologies, and know-how in CMOS are so pervasive that it will continue to lead the industry for the next decade.

A second important trend is that scaling is continuing its relentless pace according to Moore's Law. The industry reached the $1\text{ }\mu\text{m}$ minimum line width around the time of publication of the second edition. This was once considered to be a physical and psychological limit of scaling. However, it was quickly realized that transistors could be fabricated with minimum dimensions well below $1\text{ }\mu\text{m}$ and this ushered in the submicron era. Since then, we have been witness to six generations of technology scaling: $0.8\text{ }\mu\text{m}$, $0.5\text{ }\mu\text{m}$, $0.35\text{ }\mu\text{m}$, $0.25\text{ }\mu\text{m}$, $0.18\text{ }\mu\text{m}$, and $0.13\text{ }\mu\text{m}$.

At roughly the $0.35\text{ }\mu\text{m}$ technology node, we entered the deep submicron era where a number of fundamental changes were encountered in behavior of transistors and wires. Devices experienced a number of short-channel effects, perhaps the most important of which is velocity saturation. Interconnect began to control many of the electrical properties of the design such as delay, noise, power, and reliability. As a result, the transition from $0.18\text{ }\mu\text{m}$ to $0.13\text{ }\mu\text{m}$ also triggered a wholesale switch from aluminum to copper to mitigate these new interconnect issues. With all of these fundamental changes in the IC industry, there was a need to completely revise this book.

What's in the New Edition?

- We have focused on CMOS technology and moved the bipolar material to an Appendix.
- The content is based on $0.18\text{ }\mu\text{m}$ and $0.13\text{ }\mu\text{m}$ CMOS technologies.

xiii

- Most recent CMOS fabrication processes are described, including shallow-trench isolation, copper interconnect, and low- k dielectrics.
- Two important chapters have been added on deep submicron interconnect.
- New material has been added on logical effort that is useful for back-of-the-envelope calculations of optimal gates sizes in high-speed CMOS design.
- Advanced material has also been added on Flash memories, field-programmable gate arrays, and content-addressable memories.
- A chapter has been added on advanced power grid design, clock design, and phase-locked loops.
- A new section has been added on BSIM3 modeling and simulation.
- This edition uses short-channel MOS device equations with velocity saturation throughout the book. New equations have been derived for noise margins and switching thresholds.
- Junction capacitance modeling has been updated to include shallow-trench isolation effects.
- A SPICE tutorial has been provided in Appendix A.
- Gallium arsenide has been removed from this edition.

With these changes, the new edition of the book will be valuable for many years to come.

Why Buy This Book?

This book will teach you how to think like a designer. It is the product of the design know-how of many of the leaders in the field. This edition also contains extremely valuable information for practicing engineers in the semiconductor industry. The real value of the book is that it starts with the fundamentals of semiconductor devices and sequences in a step-by-step fashion to the key issues of digital IC design in deep submicron technologies. Like the previous editions, it is intended to be used as a fourth-year university textbook, with some of the latter material suitable for first-year graduate courses.

This book retains the readability and accessibility of previous editions that led to its widespread use. While the goal is to provide the reader with the key concepts and equations for digital IC design, it provides much more detail than other books in the field. Concepts are introduced in a logical sequence and then reinforced throughout the book. Complex equations are reduced to simple design equations that can be used to carry out rapid hand calculations.

There are many derivations, worked examples, exercises, and SPICE simulations to allow the student and practicing engineer to build a solid understanding of the material in each chapter. There are problems at the back of the chapter to gain experience with the material and explore other topics not directly covered in the chapter. The chapter summaries are in the form of useful equations and parameters from each chapter. The inside front and back covers provide useful parameters, scale factors, and conversion factors that are used throughout the book.

Website for the Book

A website for the textbook is available through McGraw-Hill. The website contains solutions, supporting material, color diagrams, illustrations, design problems, and errata. The website is valuable in constructing lecture material for a course based on this book.

Advanced CMOS Technologies

The material in this book is built around the $0.18\text{ }\mu\text{m}$ and $0.13\text{ }\mu\text{m}$ technology nodes. Both technologies are covered in enough detail to select one or the other as the primary technology for a course based on this book. Since SPICE is the simulation tool used in this book, it is necessary to have access to device models for one of these two technologies for the course. All of the SPICE examples are based on $0.18\text{ }\mu\text{m}$, while the worked problems are based on both $0.18\text{ }\mu\text{m}$ and $0.13\text{ }\mu\text{m}$ technologies. The textbook is also suitable for technologies below $0.13\text{ }\mu\text{m}$, with appropriate changes in the examples and problems at the back of the chapters.

Advanced Topics

Advanced topics are included for students wishing to go beyond the basic material of this text and gain insight into the issues in industry. They are designated with an asterisk (*) and may be skipped without loss of continuity.

How to Use This Book

Chapters 1–3 comprise the foundational material upon which the rest of the book is based. Chapters 4–7 are the main body of the text where all of the transistor-level design issues are described in detail. Chapter 8 focuses on an application of all material covered in the book, specifically in static memory design. Chapter 9 describes advanced topics in semiconductor memory design. The book continues with a detailed treatment of deep submicron interconnect issues in Chapter 10 and concludes with power grid and clock design in Chapter 11.

For a 15-week semester course, with 3 hours of lecture time each week, the entire text may be covered at the rate of about one chapter each week. For a 12-week course, there is sufficient time to cover Chapters 1–8, Chapter 10, and selected topics in Chapters 9 and 11.

In a 10-week course, only Chapters 1–8 and Chapter 10 can be covered assuming that much of the advanced topics are skipped. Depending on the background of the students entering the course, instructors should use their own discretion in selecting the appropriate material and sequencing of this book.

A chapter-by-chapter outline of the topics follows.

Chapter 1**Deep Submicron Digital IC Design**

In the first chapter, we provide a perspective on digital design in the deep submicron era and motivate the topics in the rest of the book. We briefly review important concepts of logic gates primarily to provide notational context for the book, including the ideal logic element, static input-output characteristics, noise margin, and propagation delay time. Then we elaborate on the key issues in deep submicron such as power dissipation, velocity saturated transistors, interconnect resistance, coupling capacitance, and inductance. The role of computer-aided design tools such as SPICE is described. The chapter concludes with the challenges ahead as described in the technology roadmap.

Chapter 2**MOS Transistors**

The short-channel device models to be used in the rest of this book are described in this chapter. To begin the treatment, rudimentary device physics concepts are covered to explain the threshold voltage equation and the transistor current equations. Both long- and short-channel models are derived. Then the oxide and junction capacitance models are described to complete the chapter.

Chapter 3**Fabrication, Layout, and Simulation**

This chapter describes the relationship between fabrication, layout, and simulation in the integrated circuit design process. The topics of fabrication and layout are important to IC designers and should be well-understood. However, for this course, it serves more as a background for the rest of the text. The more important subject is simulation with SPICE. The model parameters for SPICE are detailed in this chapter. A brief user manual is provided in Appendix A for those unfamiliar with the basics of SPICE. We also provide some advanced material on MOS transistors and fabrication technologies of the future.

Chapter 4**MOS Inverter Circuits**

This is a core chapter on MOS digital inverters and introduces concepts of voltage transfer characteristics, noise margins, inverter configurations, and simple timing and power calculations. Analytical equations are derived for noise margin parameters and switching thresholds of inverters for a number of MOS inverters.

Chapter 5**Static MOS Gate Circuits**

This is another core chapter of the book. It examines the static design issues for NANDs, NORs, and complex gates. It develops extensions to inverter design to size the transistors in CMOS gates. Sequential elements, such as flip-flops and latches, are described in this chapter. The chapter concludes with a detailed treatment of the various components of power dissipation in logic gates.

Chapter 6

High-Speed CMOS Logic Design

This chapter describes the issues involved in high-speed logic design. It develops useful equations for switching delay calculation for step and ramp inputs. This involves the use of the large-signal on-resistance of gates, and the calculation of the loading capacitance, both of which are detailed. The total capacitance can be computed using two key parameters for input and output capacitance. Gate sizing for equal delay and minimal delay are described. The thrust of the latter half of this chapter is high-speed logic optimization using logical effort.

Chapter 7

Transfer Gate and Dynamic Logic Design

In this chapter, we explore dynamic design techniques using transmission gates and precharged logic. The important concepts of charge-sharing, bootstrapping, feedthrough, and charge leakage are elaborated in this chapter. Domino logic is described in detail as it is the most common form of dynamic logic in use today. These concepts lay the groundwork for the operation of many of the memory circuits discussed in the next two chapters.

Chapter 8

Semiconductor Memory Design

This chapter addresses the analysis and design of VLSI memories, commonly known as semiconductor memories. In this chapter, we classify the different types of memory, examine the major subsystems, and then focus on the static RAM (SRAM) design issues. This topic is particularly suitable for our study of CMOS digital design as it allows us to apply many of the basic concepts presented in earlier chapters. The entire design process for static RAMs is described in detail.

Chapter 9

Additional Topics in Memory Design

This chapter explores a variety of other semiconductor memories, their architectures, access mechanisms, and cell configurations. We begin by examining content-addressable memories, since they are a derivative of the SRAM architecture. We also cover an important application of SRAM cells in the growing market segment of programmable logic called field-programmable gate-arrays (FPGAs). The chapter sequences through dynamic RAMs, mask-programmable ROMs, erasable programmable ROMs, electrically-erasable ROMs and Flash memories, and concludes with a look at memory cells based on ferroelectric materials called FRAMs.

Chapter 10

Interconnect Design

This chapter is devoted to the study of interconnect issues that the IC designer faces when designing in deep submicron technologies. It addresses the issues associated with RLC aspects of wires in detail. We begin by re-examining the RC delay calculation using the Elmore delay, and address the issue of buffer insertion in long wires.

Then we examine the capacitances associated with 3D interconnect which leads to a discussion of the effect of coupling capacitance on delay and crosstalk in logic circuits. The chapter continues with a discussion of inductance effects, and concludes with a look at antenna effects.

Chapter 11

Power Grid and Clock Design

In this chapter, we address two chip-level design issues that are dominated by interconnect: power system distribution and clock distribution. This chapter addresses advanced issues such as power routing, IR drop, Ldi/dt , and the impact of voltage drop on timing. The chapter highlights the interaction of the power system with the clock design. It also addresses clock generation and phase-lock loop (PLL) circuits to conclude the chapter.

Acknowledgments

The authors acknowledge the many colleagues and students who aided us in the preparation of all editions of this book over two decades. This extensively revised and augmented third edition has been developed by Professor Resve Saleh over the past three years. Much of the new material was developed in the Spring of 2000 when Professor Saleh taught at Stanford University. Professor Mark Horowitz at Stanford provided the original blueprint for a new course on deep submicron digital IC design which ultimately found its way into this edition. It was indeed a pleasure to work with him during that year to gain his insights into the important issues in circuit design today. His approach to digital integrated circuit design and his methods of teaching served as inspiration for this book. We gratefully acknowledge his impact and thank him for his contributions. Professor Gu-Yeon Wei of Harvard University also participated as an instructor for the first course. He prepared early versions of materials that eventually wound up herein. In addition, Professor Bruce Wooley of Stanford University provided valuable course materials.

There were a number of faculty and students at the University of British Columbia who made significant contributions to the book with suggestions, corrections, and proofreading of various chapters. We appreciate the efforts of Professors Steve Wilton, Shahriar Mirabbasi, and David Pulfrey, and students Jess Chia, Payam Lajevardi, Gary Lim, Louis Hong, Laura Ishkintana, Roberto Rosales, Sean Safarpour, and Marwa Hamour. In addition, Sandy Scott assisted with many of the administrative duties associated with the book.

A number of industry professionals have also had an impact on the book, most notably those associated with Simplex Solutions. David Overhauser, Narain Arora, Steffen Rochel, Peter McCrorie, Sandy Taylor, and Michael Benoit were extremely helpful in many aspects of the material presented in this book. Pallab Chatterjee of SiliconMap provided useful technology insights for upcoming generations of CMOS.

Many people were involved in reviewing this project and offered valuable feedback. We would like to acknowledge the assistance of following people:

R. Jacob Baker, *University of Idaho*
Anantha Chandrakasan, *MIT*
Kyusun Choi, *Pennsylvania State University*
P. David Fisher, *Michigan State University*
Jim Frenzel, *University of Idaho*
Arvin Grabel, *Northeastern University*
Jeffrey L. Gray, *Purdue University*
Edwin Greeneich, *Arizona State University*
Paul Hurst, *University of California–Davis*
Shrinivas G. Joshi, *Marquette University*
Thottam S. Kalkur, *University of Colorado, Colorado Springs*
Hong Koo Kim, *University of Pittsburgh*
Ivan Kourtev, *University of Pittsburgh*
Hisham Z. Massoud, *Duke University*
Martin Margala, *University of Alberta*
Kelvin F. Poole, *Clemson University*
Ahok Srivastava, *Louisiana State University*
Dennis Sylvester, *University of Michigan*
John Uyemura, *Georgia Institute of Technology*
Chin-Long Wey, *Michigan State University*
Mona Elwakkad Zaghloul, *George Washington University*

Jan Willis and Janet Greene provided the cover material for this book and their contributions are gratefully acknowledged. We also acknowledge the support of PMC-Sierra for the writing of this book. The folks at McGraw-Hill were a pleasure to work with and we acknowledge the efforts of Emily Lupash and Joyce Watters, and countless others, who made this book possible.

We would especially like to thank Professors Richard Newton and Don Peder-son of the University of California, Berkeley, for their continuous support and encouragement over the years.

The authors dedicate this book to our families who have supported us through several editions so that we can continue to inspire the next generation of IC engineers.

Deep Submicron Digital IC Design

CHAPTER OUTLINE

- 1.1 Introduction
- 1.2 Brief History of IC Industry
- 1.3 Review of Digital Logic Gate Design
- 1.4 Digital Integrated Circuit Design
- 1.5 Computer-Aided Design of Digital Circuits
- *1.6 The Challenges Ahead
- 1.7 Summary

References
Problems

1.1 Introduction

Integrated circuits (ICs) are part of virtually every electronic component in the world today, such as cell phones, personal digital assistants, pagers, personal computers, printers, set-top boxes, automobiles, and so on. Most of these IC chips are “digital” in nature, with an increasing presence of “analog” circuitry. This textbook addresses the design of digital circuits in deep submicron (DSM) technologies. If you have taken a design course on very large-scale integration (VLSI), you will recognize many of the circuit issues covered in this book. However, the main objective of this book is to train you to “think like a circuit designer.” The goal is to give you the intuition and models that you need to be able to analyze existing circuits and create interesting new circuits. We aim to provide some real-world experience in how circuits are designed in industry, and how to make design trade-offs to achieve a good balance among speed, power consumption, and reliability. We strive to provide the understanding needed to anticipate the likely improvements and potential difficulties that may be encountered with future technologies.

Before we can talk about circuits, we need to understand the basic digital components and tools that we use to analyze them. We need to look at technology trends and what types of problems have been introduced due to technology scaling. This will provide the motivation for the topics that are described in the rest of this book.

The design of modern digital systems requires contributions from several engineering specialists. First, a *system designer*, or *system architect*, determines the desired characteristics for the overall system and prepares a detailed specification that defines all inputs, outputs, environmental conditions, operating speeds, etc. The system is then implemented using a register-transfer level (RTL) language such as VHDL or Verilog.¹ This RTL description is translated into a logic design that can meet the functional requirements. The output of this step is referred to as the *gate-level* design. Once the gate-level design is completed, it is converted to the circuit level where transistors are used to implement each logic gate. Finally, the transistor schematic is converted to an integrated circuit in the form of geometric layout usually represented in GDS-II stream format.²

Good system design requires that design decisions result in the appropriate balance among system characteristics, logic design, circuit design, layout design, and fabrication technology. Since compromises must usually be made and alternatives evaluated, it is important that the various specialists mentioned above have some knowledge of the related fields.

The task of the *circuit design engineer* is to design transistor circuits that implement the required logic functions. Whenever many copies of the desired system are to be manufactured, it is important to achieve high reliability of operation and a proper balance among cost and performance characteristics, such as timing, power, and area. The design must also operate properly in the presence of process variations, supply fluctuations, and changes in the environmental conditions. The chapters that follow address in depth the issues of microelectronic design that determine these characteristics. Of particular concern is the tradeoff between timing and power in ICs since designs today are either high speed or low power.

Today, virtually all digital systems are based on integrated circuit technology. Various design options and tradeoffs exist. Choices must be made among circuit families, level of integration (the number of circuits on a chip), and programmable versus fixed-function (“hard-wired”) circuits. A wide variety of IC technology is in use today. The most prevalent technology is the metal-oxide-semiconductor (MOS) process. Other technologies such as bipolar, GaAs, and SiGe are also available but represent a smaller part of the total market. The various integrated circuit technologies have widely differing characteristics. In the bipolar category, transistor-transistor logic (TTL) and emitter-coupled logic (ECL) have seen widespread use in the past. In the last 20 years, a dramatic transition has been made to MOS technology due to its high density of integration. In fact, integrated circuit *process and device engineers* continue to make major improvements in these technologies every 2 or 3 years. Through the 1970s, *n*-channel MOS (NMOS) technology was

¹ VHDL and Verilog are hardware description languages used to describe large digital systems.

² GDS-II is an industry standard format for describing the geometric objects in a layout.

commonplace. Since 1980, complementary MOS (CMOS) has become the dominant technology due to its low-power characteristics.

Some understanding of integrated circuit fabrication techniques is required to compare the relative characteristics of different circuit families, such as NMOS and CMOS. An appreciation for the direction and rate of change in fabrication technology is important if product designs are to provide good possibilities for evolutionary improvements. Furthermore, the statistical variation in the process must be well understood to design modern circuits. The fabrication process and transistor structure have changed over the last 10 years and we will examine these changes. While it is not the purpose of this book to describe the details of IC fabrication, it is necessary to describe the basic processing steps used today to understand both the layout of MOS circuits and the origins of the important parameters for simulation tools used for their analysis.

Computer-aided design (CAD) tools are essential in design and analysis of digital integrated circuits. We will examine the types of tools that are typically used in industry, but focus on circuit simulation that is important for cell library characterization and custom IC design. The primary workhorse tool for detailed circuit analysis is SPICE.³ We will spend time understanding MOS models for hand analysis and comparing their results against circuit simulation to validate the hand analysis. In effect, we want models that are suitable for hand analysis but accurate enough to provide insight into the actual circuit operation. Detailed circuit simulation can be used to examine the second-order issues, and address other first-order issues such as process, temperature, and supply variations.

1.2 Brief History of IC Industry

The IC industry started in the late 1960s and early 1970s with a *ten micron technology*. This technology node was identified by the minimum geometry that could be printed on the chip. We refer to designs of that era as small-scale integration (SSI) and medium-scale integration (MSI), meaning that only a few gates, or a few hundred gates, could be integrated on a single chip. Every 3 years or so, technology dimensions were scaled by a factor s , which has historically been found to be equal to 0.7. This meant that a chip that was $1\text{ mm} \times 1\text{ mm} = 1\text{ mm}^2$ could be reduced to $0.7\text{ mm} \times 0.7\text{ mm} \approx 0.5\text{ mm}^2$ in the next generation. Another way to look at it is that twice the number of transistors could be integrated on the same 1 mm^2 chip as compared to the previous technology node. Because of this, the cost per logic function decreased with each new generation. This trend came to be known as *Moore's law*, perhaps the most important observation in the history of ICs. Many have predicted the end of Moore's Law several times in the past 30 years and have been proved wrong to this day.

During the 1970s, the ability to integrate thousands of gates on a single chip became feasible. This was the era of large-scale integration (LSI). Designers began

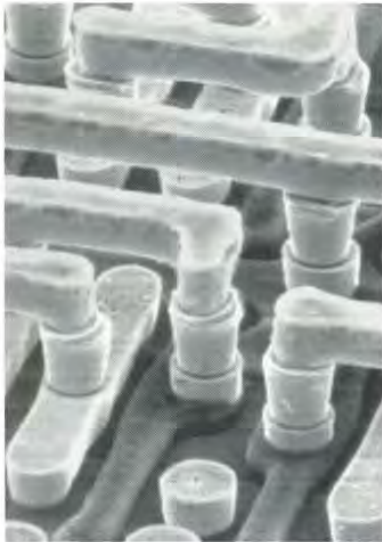
³ SPICE is an acronym for Simulation Program with Integrated Circuit Emphasis, originally developed at the University of California—Berkeley in the 1970s.

to develop microprocessors with significant processing capability, and memories with large storage capacities. Integrated circuits began to show up in pocket calculators, computers, and television sets. By the end of this decade, the enormous potential of integrated circuits was well understood. Around 1980, a revolution began in the microelectronics industry with major advances in IC processing technology, microprocessor design, memory design, and computer-aided design (CAD). This was driven by a whole host of consumer products including personal computers, printers, and VCRs. The name coined for this era was, of course, very large-scale integration (VLSI). The term *VLSI* captured everyone's imagination of the unlimited possibilities of this technology. The ability to place 1 million (1M) transistors on a single chip was within reach and the race began to develop chips that were capable of using this level of integration.

Through the 1980s, available technologies were mostly in the $5\text{ }\mu\text{m}$ to $1\text{ }\mu\text{m}$ range. This dimension nominally refers to the *channel length* of the transistor. It also refers to the minimum resolvable geometry on a given layer of metal in the integrated circuit, specifically, the metal line widths or metal-to-metal spacing (metal *pitch*). Advances in photolithography, the key process that defines the minimum dimension in a technology, eventually led to line widths that were below $1\text{ }\mu\text{m}$. This was referred to as the *submicron* era, at one time thought unreachable. However, scaling continued aggressively to the point where $0.5\text{ }\mu\text{m}$ and $0.35\text{ }\mu\text{m}$ line widths were achieved by the mid-1990s. At the same time, the number of layers of metal continued to increase. Metal layers composed of aluminum and tungsten were used to connect transistors. The industry began with only one layer of metal for all connections. As the number of transistors increased, there was a need to increase the number of levels of metal so that all required connections could be made. Figure 1.1 shows a four-layer metal process in $0.35\text{ }\mu\text{m}$ technology.

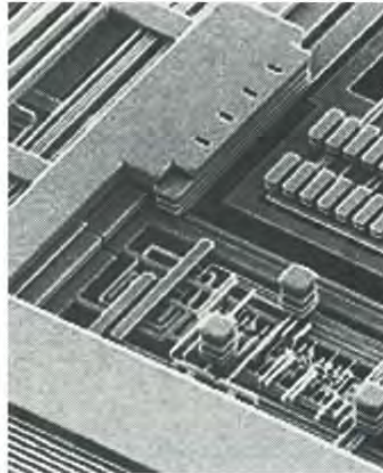
Many argued that $0.35\text{ }\mu\text{m}$ technology would be a physical limit for photolithography since the wavelength of light is approximately equal to this value. However, further advances allowed scaling below this barrier. At this point, the term *deep submicron* (DSM) was coined to emphasize that we had gone beyond another scaling limit, and well below the $1\text{ }\mu\text{m}$ barrier. The device behavior was largely controlled by a number of new *short-channel effects*. In addition, new problems began to arise concerning the metal interconnect. The wires connecting the transistor began to introduce additional RC delays in the circuit due to increased resistance. Furthermore, coupling between lines caused delay variations and noise injection. The increases in resistance in the power distribution system led to voltage drops in the power grid, commonly referred to as IR drop. The reliability of the aluminum metal lines began to degrade due to metal migration resulting from high current levels. Collectively, these issues were referred to as signal integrity problems and they characterize what we call the *deep submicron era*.

While these problems were not completely resolved, scaling continued its relentless pace to $0.25\text{ }\mu\text{m}$, $0.18\text{ }\mu\text{m}$, $0.15\text{ }\mu\text{m}$, and by the year 2001, to $0.13\text{ }\mu\text{m}$. We entered a new era where aluminum lines with tungsten vias gave way to a dual-Damascene *copper* process, as shown in Figure 1.2, to mitigate the interconnect issues mentioned above. New dielectric materials were introduced, the so-called *low-k dielectrics*, to reduce coupling effects between wires. This period goes by many names, for example, very deep submicron (VDSM) and ultra-deep submicron

**Figure 1.1**

Four-layer aluminum wires with tungsten vias in 0.35 μm technology.

[Courtesy: IBM Corporation. Unauthorized use not permitted.]

**Figure 1.2**

Copper interconnect.

[Courtesy: IBM Corporation. Unauthorized use not permitted.]

(UDSM). Because of the fine geometries, a change of units from microns to nanometers seemed appropriate. In this book, we will often refer to 0.13 μm as 130 nm technology, and vice versa. Initial developments are already underway for the 90 nm and 65 nm technology nodes.

DSM technologies have introduced new problems in both devices and interconnect. A partial list is provided in Table 1.1. While the issues that have been encountered with devices are similar in number to the list shown for interconnect, the interconnect issues have caused more failures in CMOS designs over the past few years. Many now believe that interconnect is more important than devices. We will give the two roughly equal billing in this book. The reader should note that many of the problems are not fully resolved, as yet. One purpose is to provide sufficient detail so that future designers are well-equipped to deal with these and other deep submicron issues.

This book addresses digital integrated circuit design primarily in 0.18 μm and 0.13 μm technologies. The 130 nm technology features copper wires with low- k dielectrics, twin-wells for the devices, and shallow trench isolation (STI). This technology is quite different from its predecessors, due to changes in materials, but similar to its successors in the next two technology generations, 90 nm and 65 nm. There are also many similarities with 0.18 μm technologies when examining the fundamental principles. In 0.18 μm technology, the materials are aluminum wires and

Table 1.1

A short list of major DSM device and interconnect issues

DSM Devices	DSM Wires
Short-channel effects on V_T	RC delays
Velocity saturation	IR drop
Thin-oxide (tunneling/breakdown)	Ldi/dt
Subthreshold current	Capacitive coupling
DIBL	Inductive coupling
Hot-carrier effects	Electromigration
	Antenna effects

tungsten vias. STI is also used at $0.18\ \mu\text{m}$ and below. Except for these differences, the design and analysis methods described in this book apply equally well. At the time of the writing of this book, $0.18\ \mu\text{m}$ is the dominant technology so coverage of this technology is appropriate. Furthermore, access to $0.18\ \mu\text{m}$ technology files is available through a number of websites. Therefore, many examples from $0.18\ \mu\text{m}$ are included in this book.

1.3 Review of Digital Logic Gate Design

1.3.1 Basic Logic Functions

As we embark on the design of digital integrated circuits, we start by reviewing the basic logic elements that are used in the design process⁴ and elaborate on some of the notation and symbols used in the book. Digital logic gates implement Boolean functions such as the inverter (INV), NAND, and NOR. Any logic function can be constructed from these basic gates. These *combinational* gates and their corresponding truth tables are shown in Figure 1.3.

Logic functions are represented in one of two canonical forms: sum-of-products or product-of-sums. For example, a sum-of-products representation takes the form:

$$F = AC + BC + AD + BD$$

whereas a product-of-sum form for the same function would be

$$F = (A + B)(C + D)$$

The sum function is always represented using the “+” symbol. The product function may be represented as “*” or “•”, or it may be left out altogether. Typically, we will employ the sum-of-products form, but both are useful. Other useful combinational gates such as the exclusive OR (XOR), exclusive NOR (XNOR), and multiplexers

⁴ Although the treatment is brief, those familiar with this subject may choose to skip this section.

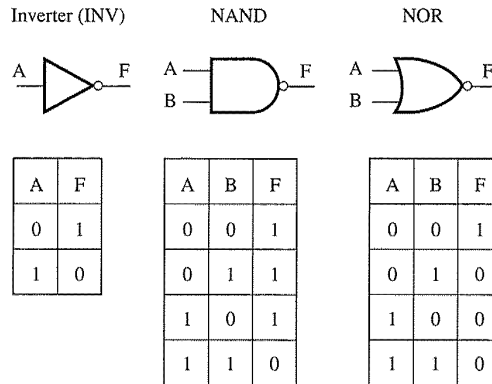


Figure 1.3
Basic logic gates.

(MUX) can be easily implemented using the basic gates of Figure 1.3. For example, the XOR gate has the function:

$$F = A \oplus B = A\bar{B} + \bar{A}B$$

A bar over a literal indicates the complement of a variable. Assuming that only A and B are available as inputs, the function can be implemented using two NAND gates, one NOR gate, and five inverters. A more efficient version would use three NANDs and only two inverters.

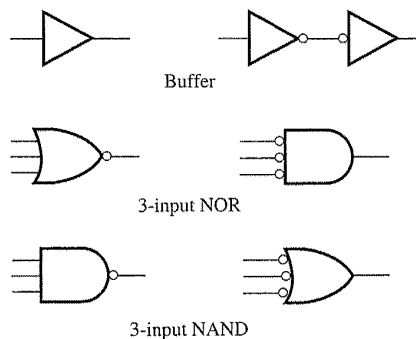
Also important is the multiplexer which is a selection of a particular input, A or B in this case, based on the value of select input S :

$$F = AS + B\bar{S}$$

Another common circuit function is the buffer, which is a gate that produces an output that is the same as its input. Using the gates in Figure 1.3, we would implement this function using two inverter stages. There are two possible symbols for a buffer, as shown in Figure 1.4. Since the intent of the gate is to perform a buffering operation, we would like to reflect this in the logic circuit somehow. We can either use a buffer symbol, or cascade two inverters and place a bubble in front of the second gate, instead of a bubble at the end (as shown in Figure 1.4). Simply having two inverters in series may lead to confusion later in the design process.

Similarly, NAND and NOR gates have two possible representations depending on the placement of the bubbles, as shown in Figure 1.4. These two representations can be constructed using DeMorgan's Laws. Recall that DeMorgan's Laws, in their most basic form, can be stated as:

$$\begin{aligned}\overline{(a + b)} &= \bar{a} \cdot \bar{b} \\ \overline{(a \cdot b)} &= \bar{a} + \bar{b}\end{aligned}\tag{1.1}$$

**Figure 1.4**

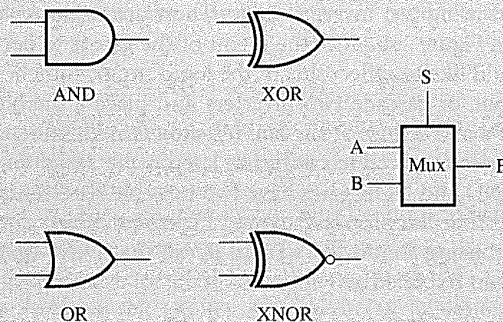
Alternative representations of buffer, NOR, and NAND.

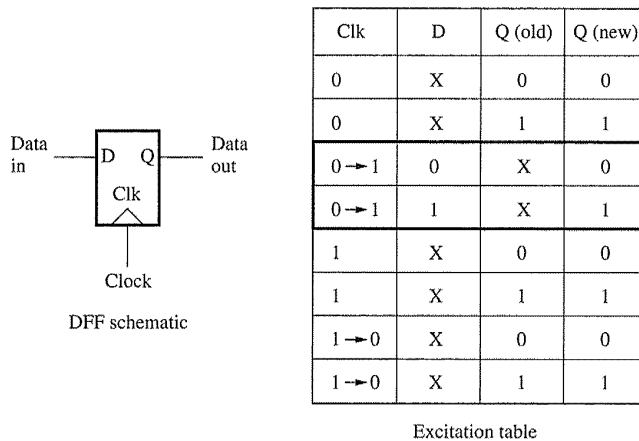
In general, DeMorgan states that the complement of a function can be obtained by replacing each variable or element with its complement, and then exchanging the AND and OR functions. It also tells us that we can represent NOR and NAND gates in two ways, as illustrated in Figure 1.4. Both sets of gates function identically.

Which representation is preferred? It depends on the intended function of the gate. If the intent of the function is to perform a logical “AND” operation, the designer should use the AND function with bubbles on the input (rather than the NOR function). If the intent of the function is to perform a logical “OR” operation, the designer should use the OR function with bubbles on the input (rather than the NAND function). Imagine the confusion when debugging if the intended logic function and the actual logic implementation look different. To avoid this, we always use the gate representation that captures design intent.

Exercise 1.1

Draw the detailed logic circuits for the following gates using only the gates of Figures 1.3 and 1.4: AND, OR, XOR, XNOR, 2-input multiplexer.



**Figure 1.5**

Positive-edge triggered D-type flip-flop.

Sequential logic elements, such as flip-flops and latches, can be constructed from the basic logic elements described above using feedback. A wide variety of flip-flops exist: D-flip-flop, JK-flip-flop, T-flip-flop, SR-flip-flop, etc. These elements act to store logic values until the input changes and a clock signal activates them to read new data. Figure 1.5 shows a simple D-type flip-flop (DFF) and its corresponding excitation table. When the clock switches from low to high, the value at the data input is read into the DFF, and then propagated and held at the output. Under all other conditions, the output value is held at its previous value. This is a so-called positive-edge triggered flop, meaning that the input is sampled only on the positive edge of the clock. The X state in the table refers to a “don’t care” state indicating that the output is not dependent on the value of the corresponding input.

1.3.2 Implementation of Logic Circuits

Electronic circuits are used to implement these and more complex gates. Figure 1.6a shows an ideal logic gate. It operates from a single power source, from which it draws a minimum amount of power (ideally zero, of course).

The two binary output levels are at zero (logic 0) and at the supply voltage V_{DD} (logic 1). The output impedance is low so that large currents may be driven into external resistive or capacitive loads without altering the output voltage level. The transition between states at the output occurs abruptly for an input level equal to one-half the supply voltage, as in Figure 1.6b. There is negligible time delay between the input and output transitions. Virtually any number of inputs are available; the input impedance is high so that the circuit imposes little loading on the driving signal. Of course, all practical logic elements fall short of the ideal performance

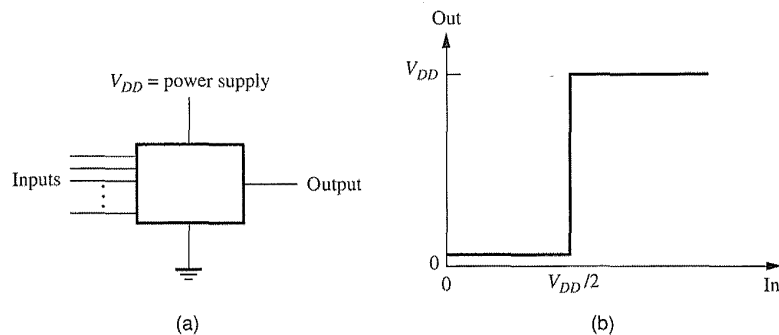
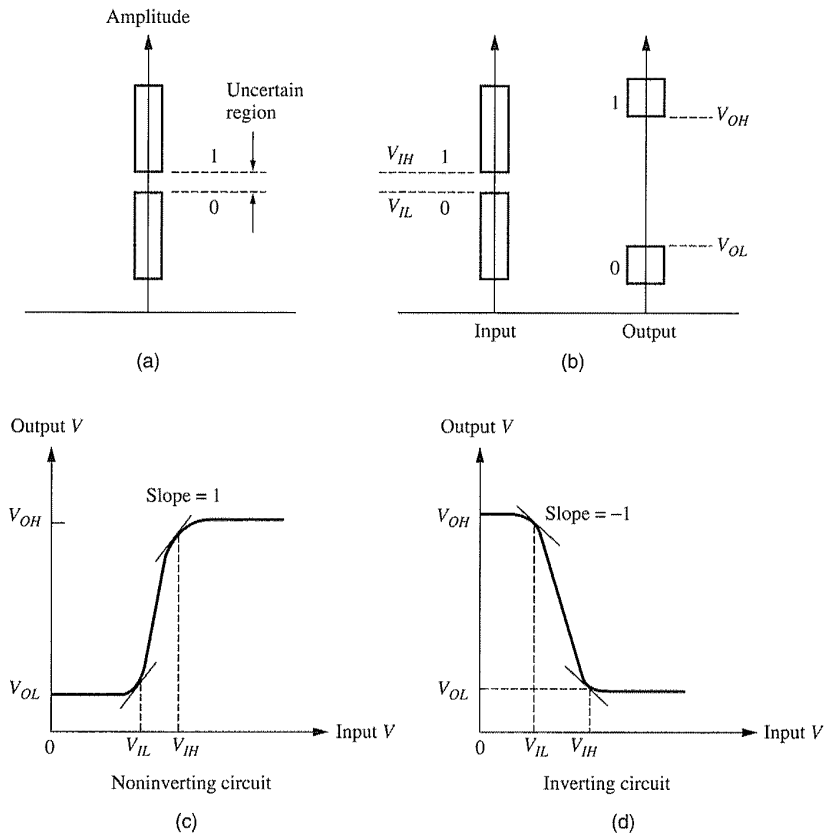


Figure 1.6
Ideal digital circuit.

defined here, but depending upon the application, some are better than others. Thus there are opportunities for good engineers to make useful tradeoffs as a part of circuit design.

The first shortcoming to realize is that the signals associated with actual implementations are continuous whereas the truth table values are discrete. Therefore, certain important characteristics are desired of electronic circuits for processing digital information.

1. The binary output signal must be a prescribed function of the binary input or inputs of the gate.
2. Quantization of amplitudes within the normal range of operating voltage is required, as illustrated in Figure 1.7a. This implies strong nonlinearities in circuit operation. Amplitudes within the boxed regions in Figure 1.7a represent each of the two binary states. At the circuit input, the uncertain region between the two boxed regions should be as small as possible.
3. Amplitude levels should be regenerated in passing from the input to the output of a digital circuit, as illustrated in Figure 1.7b. This requirement dictates a voltage transfer characteristic in the general shape shown in Figure 1.7c or Figure 1.7d. Voltage gain should be greater than unity somewhere between the logic states. The two nominal output levels are denoted V_{OH} and V_{OL} , as in Figure 1.7c or Figure 1.7d. The input voltages V_{IL} and V_{IH} are defined by the points at which the magnitude of the slope of the voltage transfer characteristic is unity.
4. Directivity is required for a useful logic circuit. Changes in an output level should not appear at any unchanging input of the same circuit; that is, there must be an explicit, unilateral cause-effect relationship between input(s) and output(s).

**Figure 1.7**

Logic abstraction of continuous signals.

5. The output of one circuit must be capable of driving more than one input of similar circuits. The number that can be driven is termed the *fanout* of the circuit. Similarly, for general use, digital circuits must be capable of accepting more than one input. The number of independent input nodes is termed the *fanin*.

1.3.3 Definition of Noise Margin

The word *noise*, in the context of digital circuits and systems, means unwanted variations of voltages or currents at logic nodes. Today, there are a wide variety of noise sources that may affect circuit performance. We want to define a *metric* that can be used to assess the effect of noise on a circuit. If the magnitude of noise is too great,

it will cause logic errors. However, if noise amplitude at the input of any logic circuit is smaller than a specified critical magnitude known as the *noise margin* of that circuit, the noise will be attenuated as the desired signal passes from input to output. In properly functioning digital systems, noise is attenuated in passing through a circuit while the desired logic signals are restored to full amplitude without error. Noise does not accumulate from one logic stage to the next as it does in analog systems. Digital systems have an important advantage in this respect and we depend on this advantage when designing digital circuits.

In the past, noise was due primarily to external off-chip sources. Today, noise also arises from internal on-chip sources, predominantly due to the nature of the interconnect structure of integrated circuits. The output levels may vary due to capacitive or inductive coupling, circuit manufacturing tolerances, temperature changes, power supply variations, and electrical loading at the output node. The worst combinations of all these effects are used to define the worst-case output voltage range.

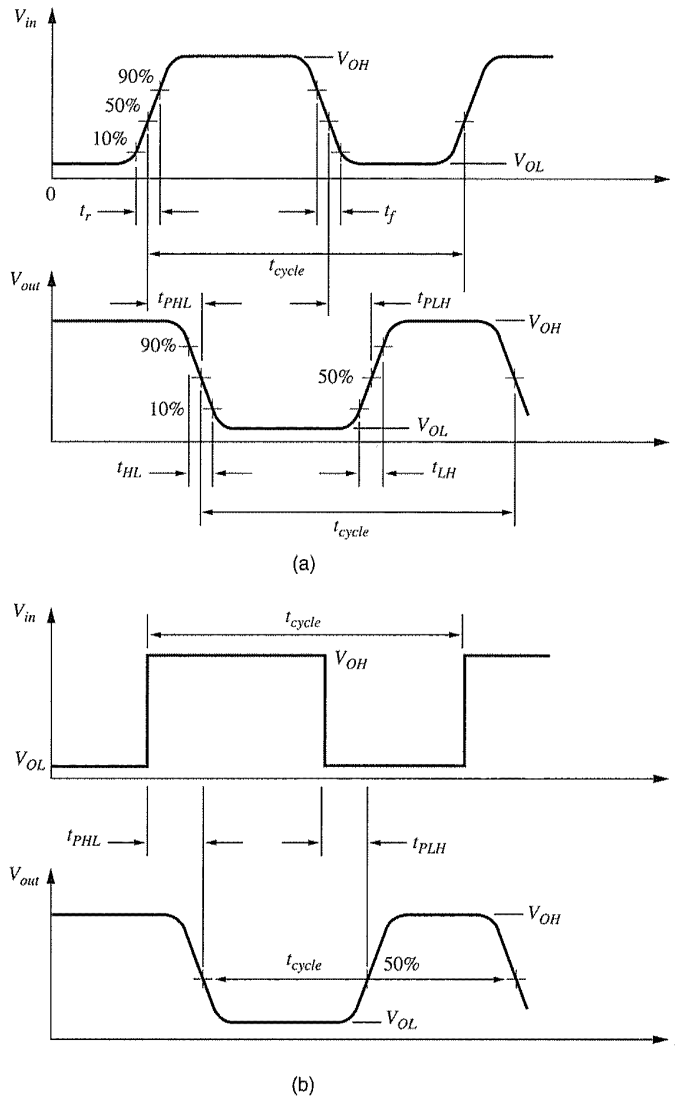
In general, noise margins are different in high and low logic states. They are denoted, respectively, as $NM_H = V_{OH} - V_{IH}$ and $NM_L = V_{IL} - V_{OL}$. As long as the input remains in these ranges, the circuit will work properly in the presence of noise. Further discussion of these terms and concepts will be found in subsequent chapters of this book.

1.3.4 Definition of Transient Characteristics

Digital logic implementations exhibit finite rise and fall delays, as well as finite propagation times due to capacitances and resistances associated with the gates. These delays limit the performance of logic circuits and we seek to minimize the overall delays along critical paths in the circuit. Variations in gates delays along different paths to a gate may cause *glitches* in its output, that is, an unexpected short pulse at the output due to differences in arrival times at the inputs. Specific definitions of transition times and propagation delay times are needed for a description of the switching characteristics of logic circuits. Once such definitions are established, calculations of these times can be made.

Standard definitions of digital circuit delay times are illustrated in Figure 1.8. Rise and fall times t_r and t_f are defined between the 10 and 90% points of the total voltage transition at the input of an inverter or gate. The total voltage range at both input and output is taken to be V_{OL} to V_{OH} because this is the nominal situation when identical inverters or gates are cascaded. Input transition voltages V_{IL} and V_{IH} are not normally used in specification of transient performance.

High-low and low-high transition times at the output of a gate are defined as t_{HL} and t_{LH} , again between the 10 and 90% points, as seen in Figure 1.8a. Propagation delay times from input to output, denoted t_{PHL} and t_{PLH} , are defined between the 50% points of the input and output pulse waveforms. Cycle time t_{cycle} is the time between identical points of successive cycles in the signal waveform as seen at any single node, as in Figure 1.8a. Often cycle time is specified in terms of its reciprocal, clock frequency f_{clk} . Practical digital systems operate with cycle times 20 to 50 times the propagation delay of a single gate circuit.

**Figure 1.8**

Definitions of transition and delay times. (a) Actual transient voltages. (b) Idealized transient voltages for hand calculations.

In hand calculations, it is difficult to take into account the finite rise and fall times of signals at inverter or gate inputs. Consequently, it is common to use an ideal pulse input with zero rise and fall times, as in Figure 1.8b. This ideal input signal is positioned with its edges at the 50% points of an actual input signal, as illustrated in

Figure 1.8b. Propagation delay times are then approximated by the time from the edge of the ideal input pulse to the 50% point of output voltage transitions. Techniques are available to estimate the effects of finite input slopes and these will be elaborated further in this book. Computer circuit simulation should be used to obtain more accurate results.

1.3.5 Power Estimation

Many of the issues in logic design today concern power dissipation. Consider Figure 1.9 which shows the power dissipation levels of microprocessor designs over a 25-year period starting in 1980, based on published results at ISSCC.⁵ Chips were increasing in power by a factor of 4 every 3 years for the first decade. In fact, the power level of some chips approached 100 W. High-speed logic typically implies high power. As gates switch during their logic operations, they consume power. The more gates that switch and the faster they switch, the more power they will consume. Because of the demand for higher and higher speeds, the power dissipation reached unacceptable levels that eventually exceeded 100 W. One of the key design problems today is to minimize the overall power consumed per logic operation. As a result, power is increasing at a rate of $1.4\times$ every 3 years over the last decade.

One difficult problem is to determine the actual power dissipation of a chip early in the design process. Power for a gate can be computed accurately, but power for an entire chip can only be estimated, since it depends in large part on the activity of the chip. Also, power dissipation varies for logic, memory, clock, analog

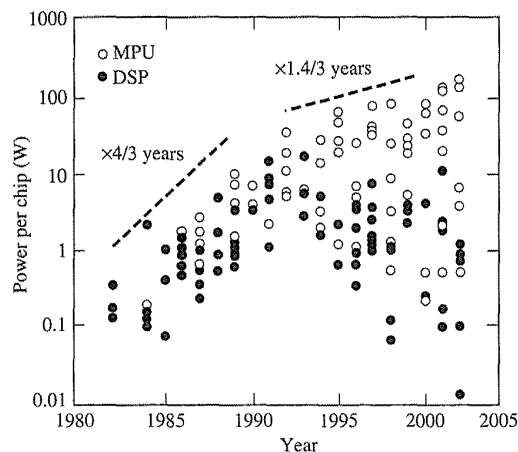


Figure 1.9

Power dissipation of processors over a 20-year period. [After Sakurai.]

⁵ International Solid-State Circuits Conference held annually by the Solid-State Circuits Society.

blocks, etc. Therefore, a variety of methods may be required to have a reasonable estimate of the total power.

The basic power equation is

$$P = I \times V_{DD} \quad (1.2)$$

where I is the current flowing from V_{DD} to ground. The power dissipation of a logic gate may be broadly categorized into static and dynamic power. Static power involves power dissipation when the gate is not switching. Dynamic power involves the power during switching operations. The total power is a combination of the static power and the dynamic power:

$$P_{total} = P_{static} + P_{dynamic} \quad (1.3)$$

For the static case, I is simply the steady-state current when there is no switching. This can be due to any dc current that may exist or leakage currents in the circuit:

$$P_{static} = (I_{DC} + I_{leakage}) V_{DD} \quad (1.4)$$

Dynamic power is associated with switching from high-to-low and low-to-high. The power dissipated is a function of the voltage swing, capacitance, and switching frequency. In the dynamic case, we will find that

$$P_{dynamic} = CV_{DD}^2 f \quad (1.5)$$

where f is the switching frequency of the gate and C is the output capacitance driven by the gate. In circuit design, we attempt to design high-performance circuits that dissipate low power. However, high-speed implies a large f and requires large transistors with a large C . This unfortunately increases the overall power dissipation. Clearly, a power-delay tradeoff exists and we can only hope to balance the desired speed with the maximum allowable power dissipation. Even though this is a simple example, it provides some insight into the type of compromises that are involved in circuit design.

1.4 Digital Integrated Circuit Design

Simply stated, design is the effective management of a large number of engineering tradeoffs. This is true in virtually all engineering disciplines, but it is particularly true in design of integrated circuits where a large number of tradeoffs are possible. The main tradeoffs relate to timing, power, and area. Today, timing and power are the two most important specifications for the design, with area being a lower priority due to the scaling of technology and the level of integration that is possible. Other important issues include noise tolerance, testability, yield, temperature, supply fluctuations, short-term and long-term reliability, time-to-market, cost, and packaging considerations. Each design may have a different set of priorities relative to these issues, but all of these factors will influence the design decisions in one way or another.

As mentioned earlier, the goal of this book is to teach you to think like a designer. While all of the above issues cannot be fully described in one book, the art of making engineering tradeoffs can be conveyed. This involves the use of simple models that allow you to carry out “back-of-the-envelope” calculations. This will also help to develop intuition about circuits and how they work. Other techniques will be described that allow you to quickly optimize a circuit for speed and assess different alternatives for a particular function. A popular misconception is that good circuit design involves new and innovative circuit topologies. In contrast, industrial circuit design is more concerned with getting a chip to work the first time in the presence of process and environmental variations, and at low cost. It must also be delivered within a market window and operate properly for the expected lifetime of the chip. While there is a place for new circuit configurations, this book addresses mainstream circuit design techniques.

1.4.1 MOS Transistor Structure and Operation

We now turn our attention to the MOS transistors that will be used to implement logic gates. At a very high level, the MOS transistor essentially behaves as a *switch*. It has two major states: *on* or *off*. When it is *on*, it is conducting current; in the *off* state, there is no current flow. This basic transistor action can be used to build up complex logic functions. For example, parallel combinations of these switches can be used to implement the OR function, while series combinations can be used to implement the AND function. We will illustrate, in later chapters, how a collection of transistors can be configured to operate as an inverter, buffer, NOR, NAND, MUX, D-flip-flop, etc. At this stage, we simply need to understand the structure and operation of the transistor, so that some of the design issues can be elaborated further.

The structure of an NMOS transistor is shown in Figure 1.10. The acronym MOS refers to the vertical layers of metal-oxide-semiconductor. The metal is usually a silicon-based material called polysilicon.⁶ The oxide is silicon-dioxide with a thickness of t_{ox} as shown in the figure. The transistor is comprised of two heavily doped n^+ regions diffused in a lightly doped p -type substrate. The n^+ regions have a large supply of mobile electrons available for current flow, while the p -type region has a large supply of mobile holes. The n^+ regions are separated by a distance, L , referred to as the channel length. The other important dimension of the transistor is the channel width, W , which goes into the page (not shown). This device is called an NMOS transistor since it has two n^+ regions, referred to as the *source* and *drain*. The other two terminals of the device are the *gate* at the top and the *substrate* at the bottom.

In the figure, the source and substrate terminals are both grounded. The gate terminal has voltage V_{GS} applied to it and the drain terminal has voltage V_{DS} applied to it. When V_{GS} is 0 V, the transistor is said to be in the “off” condition. This is because there is no current flow between the drain and source regions. In this condition, there is no path between the source and drain for electrons to flow. In fact, there are more holes than electrons in the region between the source and drain.

⁶ The use of the word metal is somewhat of a misnomer. In the early days, aluminum was used as the metal gate but the industry quickly switched to polysilicon due to yield problems. The acronym MOS has remained with us in spite of the change in material.

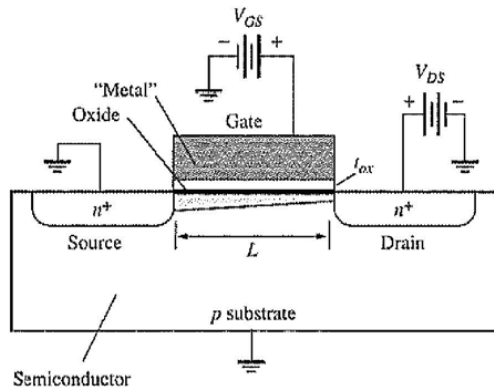
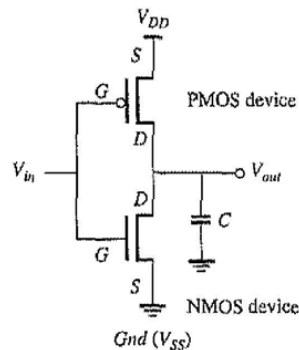


Figure 1.10
Basic transistor structure.

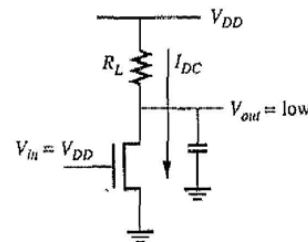
When a sufficiently high, positive value of V_{GS} is applied at the gate node, it will invert the surface of the semiconductor. The gate voltage at which this inversion occurs is called the *threshold voltage*, V_T , of the device. Under these conditions, a bridge of mobile electrons exists extending from source to drain. The transistor is considered to be “on” in this condition. Current will flow if a potential difference is created between the source and drain nodes using V_{DS} . This is an *n-channel* transistor since the polarity of the carriers in the channel is negative. A similar but opposite transistor, the *p-channel* transistor, can be created by diffusing two heavily doped p^+ regions into a lightly doped *n*-type substrate. For the PMOS device, a negative voltage V_{GS} of sufficient magnitude will invert the surface, and a negative potential difference for V_{DS} will initiate hole current flow.

1.4.2 CMOS Versus NMOS

With this basic understanding of the NMOS and PMOS devices, we can construct our first logic gate: the CMOS inverter. As shown in Figure 1.11, a CMOS inverter is comprised of an *n*-channel device and a *p*-channel device. The inverter is connected between the positive power supply voltage, V_{DD} , and ground, which we refer to as *Gnd* throughout the text (V_{SS} is also used for this reference node). The PMOS device is the *pull-up* device connected to V_{DD} . Its role is to pull the output to a high value. The NMOS *pull-down* device is connected to *Gnd*, and its role is to pull the output to a low value. The input, V_{in} , is connected to the gate terminal of both devices, while the output, V_{out} , is the drain terminal of both devices. When the input is at V_{DD} , the NMOS device is *on* while the PMOS device is *off*. Hence, the drain current of the NMOS device flows through a highly conductive channel and discharges the output capacitance. The result is that V_{out} is 0 V. When the input is at 0 V, the NMOS device is *off* while the PMOS device is conducting to pull the output to V_{DD} . Clearly, this configuration functions as an inverter.

**Figure 1.11**

CMOS inverter schematic.

**Figure 1.12**

Basic NMOS-style inverter.

In either the high or low state, CMOS inverters dissipate very little power because one of the transistors is always off. The tiny steady-state power consumption of CMOS is its most attractive feature.

In the 1970s and early 1980s, NMOS technology was dominant. In this technology, *p*-channel devices were not allowed, since *n*-channel devices were known to be superior to *p*-channel devices. All gates were constructed from *n*-channel transistors. An example of an NMOS inverter is shown in Figure 1.12. Here, the pull-down device is the same as in the CMOS inverter, but the pull-up device is shown as a resistance, R_L . The resistance was implemented with a special type of *n*-channel transistor that remained *on* all the time, referred to as a *depletion-mode device*. When the input to this inverter is low, the resistor pulls the output to a high value since the pull-down device is off. When the input is high, the pull-down device attempts to pull the output low, while the resistor tries to pull the node high. The sizing of the devices ensures that a valid low output is established. However, when the output is low, there is dc current flowing from V_{DD} to Gnd given by $(V_{DD} - V_{OL})/R_L$. As a result, the gate dissipates steady-state power when the output is low, as shown in Figure 1.12.

When the scaling of MOS technology permitted several hundred thousand gates on the chip, it began to signal the demise of NMOS technology. The reason is that if half the gates on the chip had a low output, then they would be dissipating power. In the standby condition where no logic switching takes place, high levels of power would be dissipated unnecessarily. As the number of gates increases in accordance with Moore's Law, the power continues to increase. In the case of CMOS gates, very little standby power is dissipated whether the output is high or low. In fact, even with a number of notable disadvantages of CMOS, its low standby power eventually won out and led to the extinction of NMOS-only technologies.

In keeping with its technology dominance, this book focuses on CMOS digital circuits. It covers many different types of CMOS logic gates, their implementation and their design tradeoffs. While there have been many types of interesting CMOS

circuits devised over the years, this book aims to provide coverage on the logic families that have emerged in mainstream design. However, there are new issues to address associated with deep submicron interconnect providing a second major focus for this book. This is highlighted in the next section. The hope is that, with a solid understanding of CMOS logic design and interconnect design concepts covered in the chapters to follow, the reader will be able to quickly grasp almost any new circuit topology or interconnect issue in the open literature.

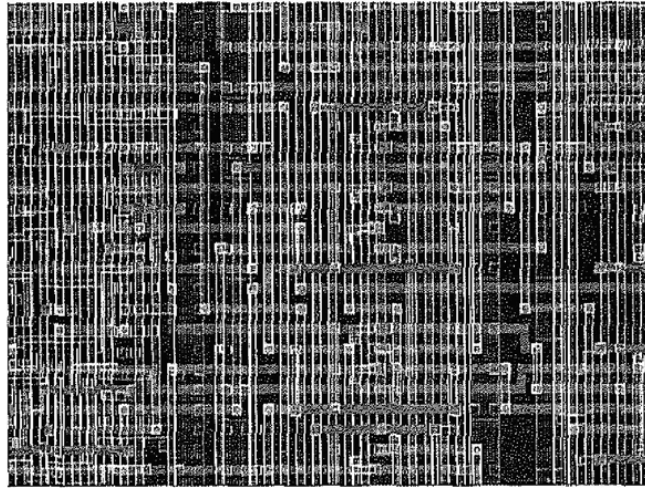
1.4.3 Deep Submicron Interconnect

We now turn our attention to the wires in deep submicron technology. These wires, commonly known as *interconnect*, are used to provide connections between gates and also to route the power supply and ground to all the gates on the chip. In the early days of integrated circuit design, the connection between the transistors that form the logic gates was a simple routing exercise. Only one or two metal layers were available to wire up the devices. Since there were only a few thousand gates on the chip, the wiring process was rather straightforward, and the capacitances and resistances associated with these wires could be safely ignored. With technology scaling, the transistor density increased tremendously following Moore's Law. A new problem arose due to routing congestion. As the routing capacity of each layer was exhausted, the addition of many more levels of interconnect was needed to complete the routing. Meanwhile, the length of the wires increased to the point where the capacitance-to-ground of the wires could no longer be neglected. For example, to compute the delay of the inverter shown in Figure 1.11, calculation of capacitance value at the output required the inclusion of wire capacitance for any wire with an appreciable length.

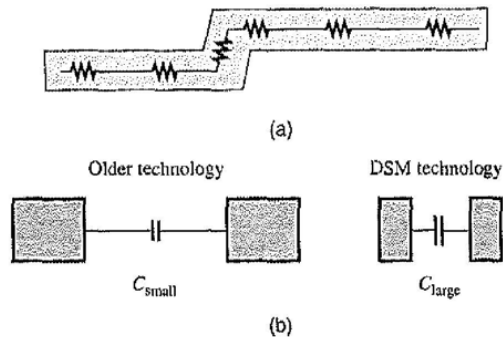
Figure 1.13 provides an indication of the sheer density of interconnect in ICs today. Wires on a particular layer run in one direction while wires on another layer run in an orthogonal direction. In the past few years, the number of layers of interconnect has grown to more than eight.

More importantly, the introduction of multilayer interconnect and scaled dimensions have caused numerous problems. Interconnect issues now dominate the performance, reliability, and power distribution of advanced integrated circuits and, in some sense, interconnect is more important than the devices carrying out the logic operations.

Two prominent effects of scaling are shown in Figure 1.14. First, narrow wires have a nonnegligible resistance associated with them as illustrated in Figure 1.14a. This was noticed in earlier technologies but it became a problem at the $0.35\text{ }\mu\text{m}$ technology node. Second, the wires are getting thinner but not decreasing as rapidly in height. This produces wires that look like tall thin conductors. When placed next to one another they form parallel plate capacitors that couple two signals together. This capacitance was present in earlier generations, but its value was negligible. In present day technology, this *coupling capacitance* is significant, as seen in Figure 1.14b. This is also reflected in Figure 1.13 by the number of parallel lines adjacent to one another. These lines are tightly coupled due to the large capacitances between them.

**Figure 1.13**

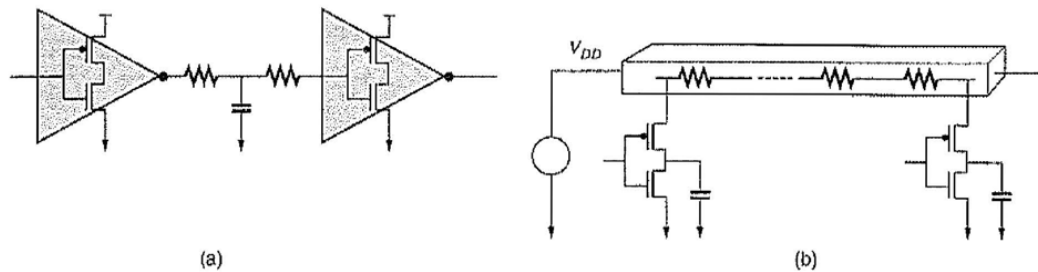
Example of integrated circuit layout dominated by routing.

**Figure 1.14**

Wire resistance and coupling capacitance.

In Figure 1.15, we show two effects of increased resistance. The delay of a signal can no longer be computed with the wire capacitance alone. Wire resistance must now be included (Figure 1.15a) to determine the delay accurately. The increased resistance in the power supply lines, as shown in Figure 1.15b, causes voltage drops along the wire, referred to as *IR drop*. This may affect the timing and functionality of gates connected to the power lines if the drop becomes too significant.

An increase in wire resistance can sharply increase the delay along the interconnect as the length increases. For long wires, the interconnect delay is propor-

**Figure 1.15**

Effect of interconnect resistance. (a) RC delay. (b) IR drop.

tional to the square of the length. At some point the interconnect delay overtakes the gate delay. This is not an acceptable situation when the wire delay is larger than the delay of the gate driving it.

A number of new design techniques were introduced to mitigate this problem. For example, copper was introduced as wire material due to its lower resistivity. Another effective approach for long wires is to insert large buffers at intervals along the wire to keep the delay from being larger than the gate delay. By inserting buffers, the interconnect lengths are effectively shortened, thereby reducing the delay.

Clock design has become a major challenge due to the size of chips, the number of flip-flops to be driven, and the interconnect effects. The clock is an important signal that must be routed to almost all points of the chip. Differences in the arrival times of the clock to their target flip-flops is called *clock skew*. To minimize delay and skew, the interconnect lengths can be shortened by inserting buffers at certain intervals along the wire and ensuring that the delay to all points is roughly the same.

Figure 1.16 illustrates the impact of interconnect on clock design. A buffered clock tree and its corresponding delay diagram is shown in the figure. On the left, the clock root node is situated in the lower portion of the design and the main spine is routed from this point to the various blocks on the chip. The clock signal is then delivered to the target flip-flops in each block. A three-dimensional view of the delay from the clock root to various points on the chip is shown on the right side. This delay plot is color-coded such that clocks that arrive early are shaded white and gray, and late-arriving signals are darker. In the figure, there are a number of late-arriving signals with a high skew as evidenced by the many dark peaks.

Increased resistance in the power grid leads to IR drop, as illustrated in Figure 1.17. This is a contour plot of the voltage drop across the chip. The ideal supply voltage exists only at the boundaries of the chip, with reduced values available to gates inside the chip. If the voltage drops below 10% of the ideal value of V_{DD} , the circuit may not operate as expected. A different shade is used for each voltage range starting with a lighter shade for voltages near V_{DD} . The dark shading near the center of the chip indicates a violation of the noise budget of 10% calling for a redesign of the power grid.

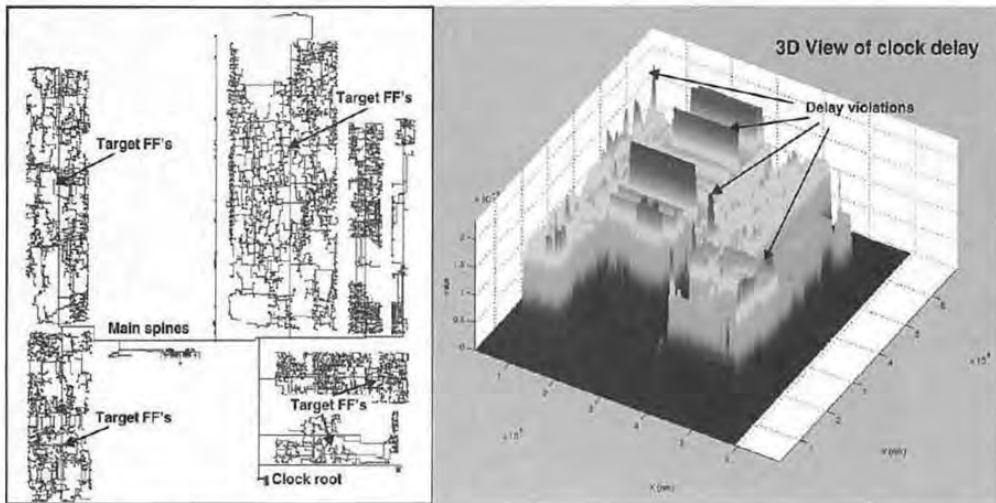


Figure 1.16
Clock tree and 3D delay plot. Plots courtesy of Simplex Solutions, Inc.



Figure 1.17
Contour plot of IR drop on a digital IC. Plots courtesy of Simplex Solutions, Inc.

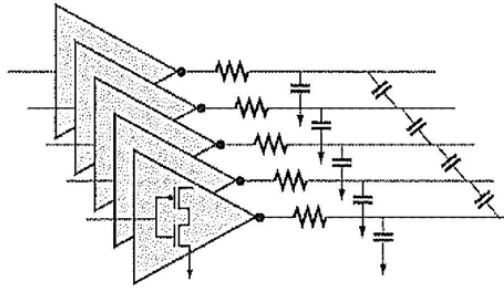


Figure 1.18
Effect of coupling capacitance.

The next issue to examine is the effect of coupling capacitance. A circuit representation of coupling capacitances along interconnect lines is shown in Figure 1.18. Problems occur when one or more gates switch. The switching of a given gate can affect the neighboring lines in two ways. First, the delay may be unpredictable since the effective coupling capacitance may be large or small depending on the switching characteristics of the neighboring lines. Second, noise may be injected into the neighboring lines and upset their logic values. These two problems became serious issues in $0.25\text{ }\mu\text{m}$ designs, and continue to plague designs at $0.18\text{ }\mu\text{m}$ and below. To combat this problem, designers will space out the wires to reduce capacitance. In addition, new low- k dielectrics are under development to replace oxide as the insulating material between metal lines.

If this is not enough, recently, the issues of self and mutual inductive effects have been encountered. In the power grid, an additional voltage drop of Ldi/dt is observed in addition to the aforementioned IR drop. This term is important due to the large rates of change of the current seen in advanced high-speed designs. There are also documented occurrences of mutual inductive coupling effects in multibit busses which have led to intermittent failures in the design. A realistic scenario for the power and signal lines today is illustrated in Figure 1.19. Not shown due to the

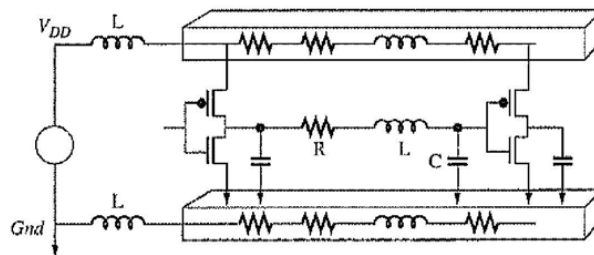


Figure 1.19
Power and signal lines in DSM logic circuits.

complexity of the diagram are the capacitive and inductive coupling between wires. Given all of these issues, it is no wonder that many feel that interconnect is truly more important than the devices in deep submicron technology.

1.5 Computer-Aided Design of Digital Circuits

Prior to 1970, electronic circuits were analyzed and designed almost exclusively by hand, a situation that is reflected in the content of textbooks from that era. Rapid growth in the feasible complexity of integrated circuits has made computer aids essential to the design process today.

In the 1980s and early 1990s, CAD tools revolutionized the way that circuits were designed. Broadly speaking, there are two categories of design tools: front-end and back-end tools. The front-end tools take a high-level description of the design and translate it to a gate-level design via logic synthesis. The back-end tools take the gate-level design and carry out the physical design to produce a layout containing the geometric artwork in GDS-II format. This flow which is intended for application-specific integrated circuits (ASIC) is a highly automated process. While this flow does not produce the highest performance ICs, it does allow designs to be implemented in a short time frame.

The other extreme relative to the ASIC flow is the custom IC design flow. Here, the designer is handcrafting almost every transistor. However, there is extensive use of repetitive structures for memory and datapath designs. The custom design process produces the highest performance designs, but it can be time-consuming since the circuits are handcrafted rather than being automatically generated. The purpose of this book is to describe the necessary details of circuit design to allow you to carry out custom design where required, and ASIC design wherever possible.

Figure 1.20 illustrates the complexity of a deep submicron physical design flow. Description of all the types of computer-aided design tools is beyond the scope of this book. However, we provide this generalized flow so that the reader can refer back to it when a particular tool is described in this book.

1.5.1 Circuit Simulation and Analysis

We will make extensive use of one type of software tool that is useful for cell library characterization and custom IC design: the circuit simulator. The program used for the examples in this book is SPICE, the most widely used circuit simulator. SPICE can perform nonlinear dc, large-signal time domain (transient), small-signal frequency domain, and other types of simulations. The dc and transient analysis capabilities are of greatest interest for digital circuit studies.

The internal numerical accuracy of programs such as SPICE is high; errors seldom exceed 1%. The accuracy of SPICE simulation in predicting circuit performance depends entirely upon how completely and correctly the input data describe the real physical circuit. There are two aspects to this issue. First, we must have a mathematical or numerical *model* that adequately represents each physical device. Usually the model for a device is stated as equations that relate voltages and cur-