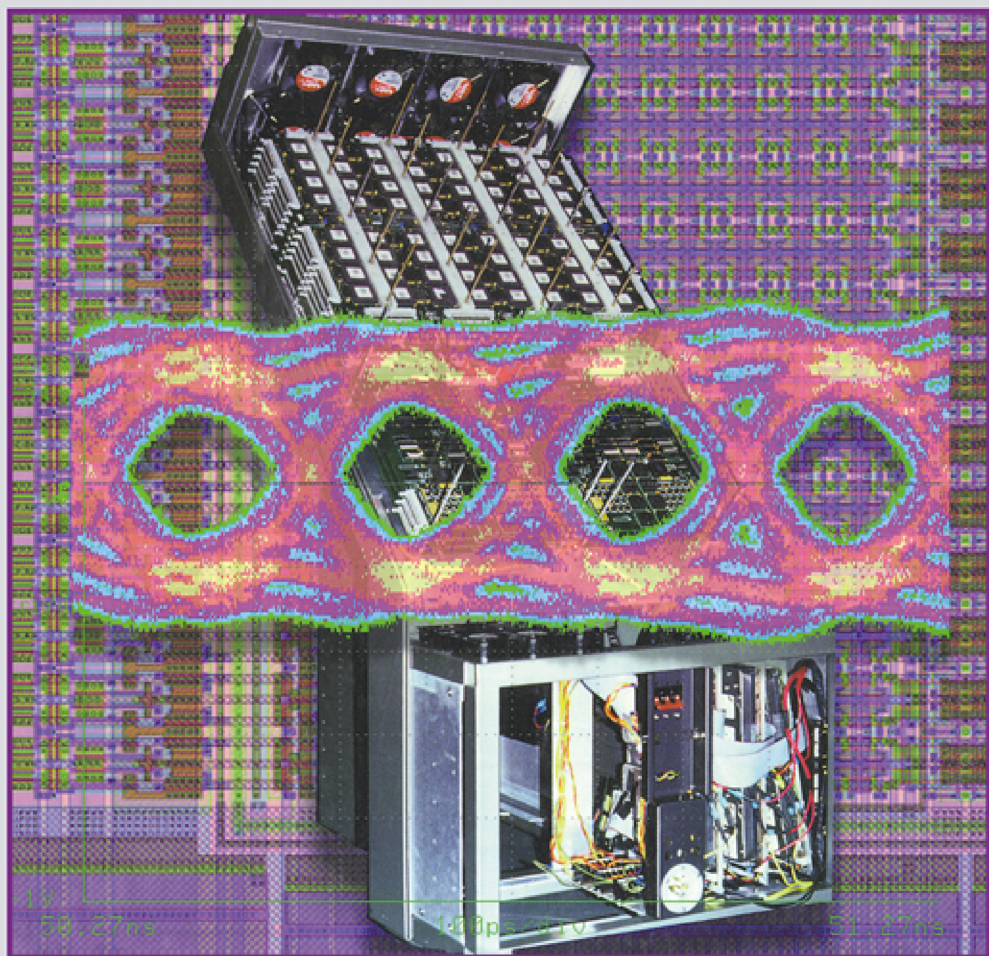
DIGITAL SYSTEMS ENGINEERING

WILLIAM J. DALLY   JOHN W. POULTON



Physical Constants

Name	Symbol	Value	Units
Speed of light in vacuum	c	3.00×10^{8}	m/s
Boltzman's constant	k_B	1.38×10^{-23}	J/K
Charge on electron	q	1.60×10^{-19}	C
Thermal voltage (@ 300 K)	$\phi_T = k_B T/q$	25.9	mV
Permittivity of vacuum	ε_0	8.85×10^{-12}	F/m
Permeability of vacuum	μ_0	$4\pi \times 10^{-7}$	H/m

Silicon Constants

Name	Symbol or Formula	Value	Units	Alternate Units
Permittivity of Silicon	$\varepsilon_{Si} = \varepsilon_{rSi} \bullet \varepsilon_0$	1.04×10^{-10}	F/m	104 aF/μm
Permittivity of SiO_2	$\varepsilon_{Si} = \varepsilon_{rSiO2} \bullet \varepsilon_0$	3.45×10^{-11}	F/m	34.5 aF/μm
Breakdown Voltage, SiO_2		7×10^{8}	V/m	0.7 V/nm
Bulk Mobility, Electrons	μ_n	0.135	$m^2/V \bullet s$	
Bulk Mobility, Holes	μ_p	0.048	$m^2/V \bullet s$	
Intrinsic Carrier Concentration in Silicon	n_i	1.45×10^{16}	$1/m^3$	

Bulk Resistivity of Various Pure Metals (ρ in $\Omega \bullet m$)

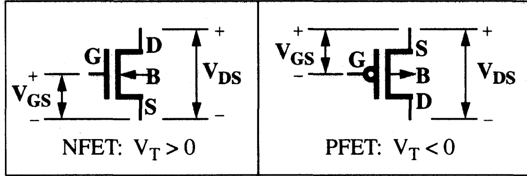
Material	Value
Silver (Ag)	1.6×10^{-8}
Copper (Cu)	1.7×10^{-8}
Gold (Au)	2.2×10^{-8}
Aluminum (Al)	2.8×10^{-8}
Tungsten (W)	5.3×10^{-8}

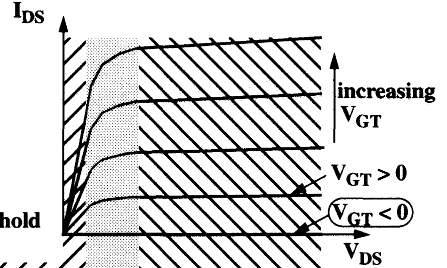
Relative Permittivity (Dielectric Constant, ε_r) of Various Materials

Material	Value
Silicon Dioxide (SiO_2)	3.9
Silicon Nitride (Si_3N_4)	7.5
Beryllia (BeO)	7.3
Alumina (Al_2O_3)	9.5
Epoxy Glass (FR-4)	4.5–5.0
Polyimide	3.5–4.5
Teflon	2.2–2.8

MOSFET Equations

Drain Current





$$I_{DS} \begin{cases} \approx 0 & V_{GT} \equiv (V_{GS} - V_T) < 0 & \text{Subthreshold Region} \\ \approx \beta V_{GT} V_{DS} & V_{GT} \geq 0,\ V_{DS} \text{ small} & \text{Linear Region} \\ = \beta\left[V_{GT}V_{DS} + \frac{V_{DS}^2}{2}\right](1 + \lambda V_{DS}) & V_{GT} > V_{DS} \geq 0 & \text{Transition Region} \\ = \beta\frac{V_{GT}^2}{2}(1 + \lambda V_{DS}) & V_{DS} \geq V_{GT} \geq 0 & \text{Saturation Region} \end{cases}$$

Device Transconductance $\beta = k\frac{W}{L}$ — W, L = width, length of device

Process Transconductance $k = \frac{\mu\varepsilon_{SiO2}}{T_{OX}}$ — $\mu = \mu_n$, electron mobility (NFETs); $= \mu_p$, hole mobility (PFETs)

Subthreshold Conduction

$I_{DS} \neq 0$ when $V_{GT} < 0$

Subthreshold Drain Current $I_{DS} = I_{ST}\exp\left[\frac{V_{GS} + \sigma V_{DS} - V_T}{n\phi_T}\right]$ — I_{ST}, σ, and n are process dependent

Body Effect

Body-Effect Threshold Voltage

$$V'_T = V_T\big|_{V_{SB} > 0} = V_{TO} + \gamma[(2|\phi_F| + V_{SB})^{1/2} - (2|\phi_F|)^{1/2}]$$

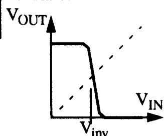
$V_{TO} = V_T\big|_{V_{SB} = 0}$ and $|\phi_F| = \frac{kT}{q}\ln\frac{N}{n_i}$ — N is the dopant atom concentration

Typical Parameters for 0.35μm Process

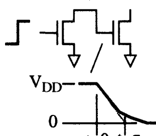
Parameter	Description	NFET	PFET
V_T	Threshold voltage	+0.5 V	–0.5 V
k	Process transconductance	200 $\mu A/V^2$	50 $\mu A/V^2$
T_{OX}	Gate oxide thickness	7 nm	
λ	Channel length modulation	0.1 V^{-1}	
γ	Body effect parameter	0.3 $V^{1/2}$	

Inverters

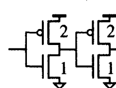
IN—▷o—OUT



$$V_{inv} = \frac{V_{TN} + \sqrt{\beta_P/\beta_N}(V_{DD} + V_{TP})}{1 + \sqrt{\beta_P/\beta_N}}$$



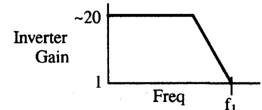
$$\tau_n = \frac{V_{DD}C_{gate}}{I_{DSSN}}$$



Delay through FO1 inverter:

$$\tau_{inv} = 3\tau_n$$

(τ_{inv}~30psec for 0.35μm process)

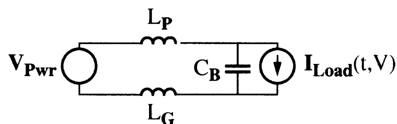


Gain-bandwidth product of inverter:

$$\omega_1 = \frac{g_m}{C} = \frac{I_{DSS}}{V_{DD}C} = \frac{1}{\tau_{inv}}$$

Power and Noise

Bypassing



Resonant Frequency of LC Circuit $\omega_C = \frac{1}{\sqrt{LC}}$

Bypass Capacitor Sizing $C_B > \frac{I_{avg}}{\Delta V}\left(k_i t_{ck} + \frac{L I_{avg}}{\Delta V}\right)$

$$k_i = \max_t \frac{\left|\int_0 (I - I_{avg})dt\right|}{I_{avg} t_{ck}}$$

t_{ck} is the clock period, I_{avg} is the time-averaged current consumption, and ΔV is the maximum allowed ripple voltage. Typical values for k_i are 0.25–0.5

Noise

Noise in a Signaling System $V_{Noise} = K_N V_{Signal} + V_{NFixed}$

Crosstalk coefficients between coupled transmission lines

$$k_{NearEnd} = \frac{k_C + k_L}{4}$$

$$k_{FarEnd} = \frac{(k_C - k_L)}{2}$$

where k_C and k_L are the capacitive and inductive coupling constants between the two lines.

For **homogeneous** coupled lines, k_C=k_L, and:

$$k_{NearEnd} = \frac{1}{4}\left(\frac{C_M}{C} + \frac{L_M}{L}\right) = \frac{C_M}{2C}$$

$$k_{FarEnd} = \frac{1}{2}\left(\frac{C_M}{C} - \frac{L_M}{L}\right) = 0$$

Johnson (Thermal) Noise in Resistors $V_{jR} = (4k_B TRB)^{1/2}$

where T is temperature in K, R is the resistance in Ω, and B is the bandwidth in Hz; V_{jR} is in volts.

Shot Noise $I_{sR} = (2qIB)^{1/2}$

where q is the electron charge, I is the current in the circuit, and B is the bandwidth in Hz; I_{sR} is in amperes.

Bit Error Rate (BER) $P(error) = \exp\left(\frac{VSNR^2}{2}\right)$

P is the probability of an error in a signaling system with $VSNR$ voltage signal-to-noise ratio. In typical signaling systems P is sampled once per bit, and $P = BER$.

Thermal Conductivity of Various Materials (κ in W/m•K)

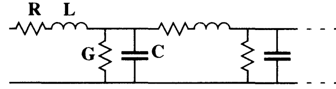
Material	Value
Silicon (Si)	150
Silicon Dioxide (SiO_2)	1.5
Copper (Cu)	420
Aluminum (Al)	240
Berylia (BeO)	220
Alumina (Al_2O_3)	35
Epoxy Glass (FR-4)	0.2
Polyimide/Glass	0.3
Teflon/Glass	0.3

Transmission-Line Equations

Slightly Lossy Line

Characteristic Impedance $Z_0 = \sqrt{\left(\frac{R+Ls}{G+Cs}\right)} = \sqrt{\frac{L}{C}}\Bigg|_{R,\,G\approx 0}$

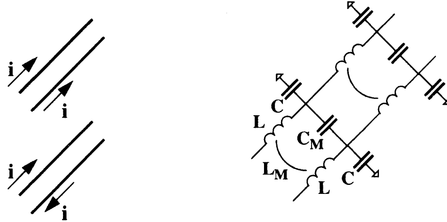
Propagation Velocity $v = \frac{1}{\sqrt{LC}}\Bigg|_{R,\,G\approx 0}$



Coupled Lines

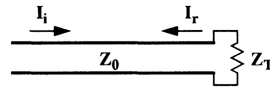
Even-Mode Impedance $Z_{Evn} = \sqrt{\frac{L+L_M}{C-C_M}}$

Odd-Mode Impedance $Z_{Odd} = \sqrt{\frac{L-L_M}{C+C_M}}$



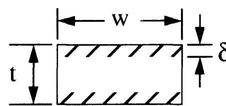
Telegrapher's Equation

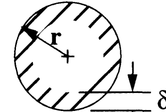
Reflection Coefficient $k_r = \frac{V_r}{V_i} = \frac{I_r}{I_i} = \frac{Z_T - Z_0}{Z_T + Z_0}$



Skin Effect

Skin Depth $\delta = \frac{1}{\sqrt{\pi f \mu \sigma}}$



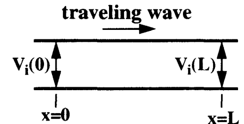


Skin Depth Frequency $f_s = \frac{\rho}{\pi\mu(t/2)^2}$ $\qquad f_s = \frac{\rho}{\pi\mu r^2}$

Skin Depth Resistance $R(f) = \frac{\sqrt{\pi f \mu \rho}}{2w} = R_{DC}\left(\frac{f}{f_s}\right)^{1/2}$ $\qquad R(f) = \frac{\sqrt{f\mu\rho/\pi}}{2r} = \frac{R_{DC}}{2}\left(\frac{f}{f_s}\right)^{1/2}$

Attenuation in Lossy Line

Attenuation $\frac{V_i(L)}{V_i(0)} = \exp[-(\alpha_R + \alpha_D)L] = \exp\left[-\left(\frac{R}{2Z_0} + \frac{GZ_0}{2}\right)L\right]$

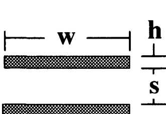


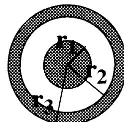
Conductor Loss $\alpha_R(f) = \frac{R_{DC}}{4Z_0}\left(\frac{f}{f_s}\right)^{1/2}$ **(Round)** $\qquad \alpha_R(f) = \frac{R_{DC}}{2Z_0}\left(\frac{f}{f_s}\right)^{1/2}$ **(Strip)**

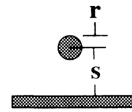
Dielectric Loss (Homogeneous) $\alpha_D(f) = \frac{\pi\sqrt{\varepsilon_r}\tan\delta}{c} f$

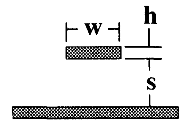
Dielectric Loss Tangent $\tan\delta = \frac{G}{\omega C} = \frac{\sigma_{Diel}}{\omega\varepsilon_r}$

R,C,Z_0 for Various Geometries (Homogeneous Dielectric, L = εμ/C)









Parallel strips: $R_{DC} = 2\frac{\rho}{wh}$, $C = \frac{\varepsilon w}{s}$, $Z_0 = \sqrt{\frac{\mu}{\varepsilon}}\frac{s}{W}$

Coaxial: $R_{DC} = \frac{\rho}{\pi r_1^2} + \frac{\rho}{\pi(r_3^2 - r_2^2)}$, $C = \frac{2\pi\varepsilon}{\log(r_2/r_1)}$, $Z_0 = \sqrt{\frac{\mu}{\varepsilon}}\frac{\log(r_2/r_1)}{2\pi}$

Two wires: $R_{DC} = \frac{2\rho}{\pi r^2}$, $C = \frac{\pi\varepsilon}{\log(s/r)}$, $Z_0 = \sqrt{\frac{\mu}{\varepsilon}}\frac{\log(s/r)}{\pi}$

Wire over plane: $R_{DC} = \frac{\rho}{\pi r^2}$, $C = \frac{\pi\varepsilon}{\log(2s/r)}$, $Z_0 = \sqrt{\frac{\mu}{\varepsilon}}\frac{\log(2s/r)}{\pi}$

Strip over plane: $R_{DC} = \frac{\rho}{wh}$, $C \approx \frac{\varepsilon w}{s} + \frac{2\pi\varepsilon}{\log(s/w)}$, $Z_0 = \frac{\sqrt{\varepsilon\mu}}{C}$

DIGITAL SYSTEMS ENGINEERING

What makes some computers fast and others slow? What makes some digital systems operate reliably for years while others fail mysteriously every few hours? Why do some systems dissipate kilowatts whereas others operate from batteries? The answers to these questions of speed, reliability, and power are all determined by the system-level electrical design of a digital system: issues of power distribution, noise management, signaling, timing, and synchronization.

Digital Systems Engineering presents a comprehensive treatment of these topics. It combines a rigorous development of the fundamental principles in each area with down-to-earth examples of circuits and methods that work in practice. The book not only can serve as an undergraduate textbook, filling the gap between circuit and logic design, but also can help practicing digital designers keep up with the speed and power of modern integrated circuits. The techniques described in this book, which were once used only in supercomputers, are now essential to the correct and efficient operation of any type of digital system.

Overall, the book describes the fundamental engineering problems of digital systems: power, noise, signaling, and timing. The best known techniques are presented for dealing with these problems, and working circuits that implement these techniques are discussed. At all levels the material is developed rigorously, and equations are derived from first principles. The book avoids the handbook approach of describing how things are done and focuses on why things are done, pointing out the limitations of current approaches. Emerging techniques are introduced that are likely to be how things will be done in the future. The book has been used in an undergraduate Digital Systems Engineering course at MIT. Materials from this course, including lecture notes, simulations, and laboratory assignments, will be available via the Web.

William J. Dally is Professor of Electrical Engineering and Computer Science at Stanford University. He has held positions at Bell Telephone Laboratories, the California Institute of Technology, and the Massachusetts Institute of Technology. He regularly consults for companies in the area of high-speed digital design. His clients include SGI/Cray Research, Digital Equipment Corp., and Intel Corp. His research interests include computer architecture, parallel computing, computer graphics, computer-aided design, and VLSI design.

John W. Poulton has been a faculty member in the Computer Science Department at the University of North Carolina at Chapel Hill since 1981 and currently holds the position of Research Professor. He was the co-inventor (with Henry Fuchs) and lead engineer on the development of several generations of Pixel-Planes graphics engines. These machines were among the first graphics systems to use custom logic-enhanced memory chips for pixel processing, which today are widely employed in graphics hardware architectures. His research interests include parallel computing, computer graphics, and circuit design of memory and communications systems in VLSI.

DIGITAL SYSTEMS ENGINEERING

WILLIAM J. DALLY   JOHN W. POULTON

CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo

Cambridge University Press
The Edinburgh Building, Cambridge CB2 8RU, UK

Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521592925

© William J. Dally and John W. Poulton 1998

This publication is in copyright. Subject to statutory exception
and to the provisions of relevant collective licensing agreements,
no reproduction of any part may take place without the written
permission of Cambridge University Press.

First published 1998
Reprinted 1999, 2000, 2001
This digitally printed version (with corrections) 2008

A catalogue record for this publication is available from the British Library

Library of Congress Cataloguing in Publication data

Dally, William J.

Digital systems engineering / William J. Dally, John W. Poulton.

p. cm.

Includes bibliographical references

ISBN 0-521-59292-5 (hb)

1. Electronic digital computers – Design and construction. 2. Digital Integrated circuits.
I. Poulton, John W. II. Title.

TK7888.3.D2934 1998
621.39 – dc21 97–43730
CIP

ISBN 978-0-521-59292-5 hardback
ISBN 978-0-521-06175-9 paperback

Cambridge University Press has no responsibility for the persistence or
accuracy of URLs for external or third-party Internet websites referred to in
this publication, and does not guarantee that any content on such websites is,
or will remain, accurate or appropriate.

About the cover: The photo on the cover shows three of the authors' projects. In the foreground is an oscilloscope trace showing an eye diagram of an equalized 4Gb/s signaling system jointly developed by the authors. Behind this trace is a 512-processor J-Machine, an experimental parallel computer developed by Dally and his group at MIT. In the background is a plot of the layout of a PixelFlow EMC chip, the heart of a high-performance graphics system developed by Poulton and his colleagues at UNC.

CONTENTS

Preface xxv
Acknowledgment xxix

1 INTRODUCTION TO DIGITAL SYSTEMS ENGINEERING *page* 1

1.1 Why Study Digital Systems Engineering? 2
1.2 An Engineering View of a Digital System 4
1.2.1 Feeds and Speeds 5
1.2.2 Signaling Conventions 6
Signaling Speed, 7 • Signaling Power, 7 • Signal Integrity, 8 • Other Signaling Conventions, 8
1.2.3 Timing and Synchronization 8
Synchronous Timing, 8 • Pipelined Timing, 9 • Closed-Loop Timing, 9 • Clock Distribution, 9 • Synchronization, 10
1.2.4 Power Distribution 10
1.2.5 Noise 10
1.2.6 A Systems View of Circuits 12
1.3 Technology Trends and Digital Systems Engineering 12
1.3.1 Moore's Law 12
1.3.2 Scaling of Chip Parameters 16
1.3.3 Scaling of Wires 17
Scaling of Power Distribution, 18 • Scaling of On-Chip Communication, 18 • Scaling of Off-Chip Communication, 19
1.3.4 High Levels of Integration Permit New Approaches 21
1.3.5 Digital Systems Problems and Solutions Continue to Change 21
1.4 Organization of this Book 22
1.5 Bibliographic Notes 22
1.6 Exercises 22

2 PACKAGING OF DIGITAL SYSTEMS 25

2.1 A Typical Digital System 27

2.2 Digital Integrated Circuits – On-Chip Wiring 29

2.3 Integrated Circuit Packages 30

2.3.1 Wire Bonds and Solder Balls 30

2.3.2 Package Types 32

2.3.3 Package Manufacturing Processes 35

2.3.4 Multichip Modules 36

2.3.5 A Typical Package Model 36
Physical Construction, 36 • Package Electrical Model, 37

2.4 Printed Circuit Boards 39

2.4.1 PC Board Construction 40

2.4.2 Electrical Properties 41

2.4.3 Manufacturing Process 42

2.4.4 Vias 42

2.4.5 Dimensional Constraints 44

2.4.6 Mounting Components: Surface-Mount and Through-Hole 44

2.4.7 Sockets 46

2.5 Chassis and Cabinets 47

2.6 Backplanes and Mother Boards 48

2.6.1 Daughter Cards 48

2.6.2 Backplanes 48

2.7 Wire and Cable 49

2.7.1 Wires 49

2.7.2 Signaling Cables 51
Coaxial Cable, 51 • Ribbon Cable, 52 • Twisted Pair, 53 • Flex-Circuit Cable, 53

2.7.3 Bus Bars 53

2.8 Connectors 54

2.8.1 PC Board Connectors 55

2.8.2 Interposers 56

2.8.3 Elastomeric Connectors 57

2.8.4 Power Connectors 58

2.8.5 Wire and Cable Connectors 58
Wire Harness Connectors, 58 • Coaxial Connectors, 60 • Ribbon-Cable Connectors, 60 • Methods of Attachment, 61

2.9 Optical Communication 62
2.9.1 Optical Transmitters 63
LEDs, 63 Laser Diodes, 63
2.9.2 Optical Fiber 64
Multimode Fiber, 65 • Single-Mode Fiber, 66 • Optical Connectors, 66
2.9.3 Optical Receivers 67
2.9.4 Multiplexing 68
Wavelength-Division Multiplexing, 68 • Time-Division Multiplexing, 69
2.9.5 Optical Amplifiers 70
2.9.6 Free-Space Optical Interconnect 70
2.10 Radio Communication 71
2.10.1 A Typical Digital Radio 71
2.10.2 The Power Equation 72
2.10.3 Modulation 74
Amplitude Modulation, 74 • Phase Modulation (PM), 74 • Frequency Modulation, 75 • Code-Division Multiple Access (CDMA), 75
2.10.4 Multipath 76
2.11 Bibliographic Notes 77
2.12 Exercises 77

3 MODELING AND ANALYSIS OF WIRES 79
3.1 Geometry and Electrical Properties 81
3.1.1 Resistance 81
3.1.2 Capacitance 82
3.1.3 Inductance 84
3.2 Electrical Models of Wires 84
3.2.1 The Ideal Wire 84
3.2.2 The Transmission Line 85
Partial Differential Equation, 85 • Impedance of an Infinite Line, 86 • Frequency-Domain Solution, 87 • Signal Returns, 87 • Lumped Models of Transmission Lines, 88
3.3 Simple Transmission Lines 88
3.3.1 Lumped Wires 88
Lumped Capacitive Loads, 88 • Lumped Resistive Lines, 89 • Lumped Inductive Lines, 90 • Lumped Models of Impedance Discontinuities, 90
3.3.2 RC Transmission Lines 90

Step Response of an RC Line, 91 • Low-Frequency RC Lines, 92

3.3.3 Lossless LC Transmission Lines 92

Traveling Waves, 92 • Impedance, 93 • Driving LC Transmission Lines, 93 • Reflections and the Telegrapher's Equation, 95 • Some Common Terminations, 96 • Source Termination and Multiple Reflections, 97 • Arbitrary Termination, 97 • Standing Waves, 99 • Summary, 100

3.3.4 Lossy LRC Transmission Lines 100

Wave Attenuation, 100 • DC Attenuation, 101 • Combined Traveling Wave and Diffusive Response, 102 • The Skin Effect, 103

3.3.5 Dielectric Absorption 105

3.4 Special Transmission Lines 106

3.4.1 Multidrop Buses 106

3.4.2 Balanced Transmission Lines 108

3.4.3 Common- and Differential-Mode Impedance 110

3.4.4 Isolated Lines 111

AC Coupling, 112 • Optical Isolation, 113

3.5 Wire Cost Models 113

3.5.1 Wire Area Costs 114

3.5.2 Terminal Costs 116

3.6 Measurement Techniques 117

3.6.1 Time-Domain Measurements 117

The Time-Domain Reflectometer, 117 • Rise Time and Resolution, 119 • Lumped Discontinuities, 120 • Transmission Measurements, 121 • Cross Talk Measurements, 121

3.6.2 Network Analysis 122

3.6.3 CAD Tools for Characterizing Wires 124

Spreadsheets, 124 • Two-Dimensional Electromagnetic Field Solvers, 124 • Signal Integrity Software Packages, 125 • 3D Electromagnetic Field Solvers, 126

3.7 Some Experimental Measurements 126

3.7.1 Frequency-Dependent Attenuation in a PC Board Trace 127

DC Resistance and Attenuation Calculations, 127 • High-Frequency Attenuation Factors, 128

3.7.2 Cross Talk in Coupled Lines 131

Coupled Embedded Striplines, 131 • Coupled

Inhomogeneous Lines, 133 • Coupling Between Lines at Right Angles, 134

3.7.3 Inductive and Capacitive Discontinuities

3.7.4 Measurement of IC Package Parasitics 138

3.7.5 Measurement Practice 142

3.8 Bibliographic Notes 142

3.9 Exercises 143

4 CIRCUITS 148

4.1 MOS Transistors 149

4.1.1 MOS Device Structure 150

4.1.2 Current–Voltage Characteristics 151

Threshold Voltage, 151 • Resistive Region, 152 • Saturation Region, 152 • p-Channel FETs, 153 • Channel-Length Modulation, 154 • Body Effect, 155 • Velocity Saturation, 156 • Subthreshold Conduction, 156 • Typical I–V Curves, 157 • Enhancement and Depletion Devices, 158

4.1.3 Parameters for a Typical 0.35-μm CMOS Process 158

4.2 Parasitic Circuit Elements 159

4.2.1 Parasitic Capacitors 160

Gate Capacitance, 160 • Source and Drain Diodes, 162

4.2.2 Parasitic Resistance 162

4.2.3 A Typical Device 163

4.2.4 SPICE Models 163

4.3 Basic Circuit Forms 165

4.3.1 Switch Networks 166

Pass Gates, 166 • Logic with Switches, 167 • Circuits Using Switches, 167 • Transient Analysis of Switch Networks, 168

4.3.2 The Static CMOS Gate 170

Inverter DC Transfer Characteristics, 170 • Inverter Gain, 172 • Transient Response, 173 • Propagation Delay and Nonzero Rise Time, 176 • The Effect of Input Rise Time on Delay, 177 • Asymmetrical Sizing, 178 • Miller-Effect Capacitance, 179 • Gain–Bandwidth Product, 180 • The Exponential Horn, 181 • SPICE Simulations of Gates, 182

4.3.3 Dynamic Circuits 184

The Dynamic Latch, 184 • Precharged Gates, 186 • Domino Logic, 187 • Dual-Rail Domino, 189 • Bootstrap Circuits, 190

4.3.4 Source Followers and Cascodes 191
Source Follower, 192 • Cascode, 194

4.3.5 Current Mirrors 195
The Basic Current Mirror, 196 • The Cascode Current Mirror, 197

4.3.6 The Source-Coupled Pair
V–I Characteristics of the Source-Coupled Pair, 199 • Differential Circuit Analysis, 200 • Differential Loads, 201 • Mode Coupling, 202 • FET Resistors, 203 • A Simple Differential Amplifier, 204

4.3.7 Regenerative Circuits and Clocked Amplifiers 207

4.4 Circuit Analysis 210

4.4.1 Qualitative Circuit Analysis 210
Qualitative Analysis of a Differential Amplifier, 210 • Qualitative Analysis of a Voltage-Controlled Oscillator, 211

4.4.2 Power Dissipation 212
Power Dissipation of a Static CMOS Gate, 212 • Energy-Delay Product of a CMOS Gate, 213 • AC Versus DC Power, 214 • Power Dissipation of Source-Coupled FET Logic, 215

4.5 Bibliographic Notes 215

4.6 Exercises 216

5 POWER DISTRIBUTION 221

5.1 The Power Supply Network 222

5.1.1 Local Loads and Signal Loads 224
Local Loads, 224 • Signal Loads, 224

5.1.2 Inductive Power Supply Noise 225

5.2 Local Regulation 228

5.2.1 Clamps and Shunt Regulators 228

5.2.2 Series Regulators 230
Linear Regulator, 231 • Switching Regulator, 233

5.3 Logic Loads and On-Chip Power Supply Distribution 237

5.3.1 Logic Current Profile 237

5.3.2 IR Drops 240

5.3.3 Area Bonding 242

5.3.4 Metal Migration 243

5.3.5 On-Chip Bypass Capacitors 243

5.3.6 Symbiotic Bypass Capacitance 244

5.4 Power Supply Isolation 245
5.4.1 Supply-Supply Isolation 245
5.4.2 Signal-Supply Isolation 246
5.5 Bypass Capacitors 247
5.6 Example Power Distribution System 249
5.7 Bibliographic Notes 256
5.8 Exercises 256

6 NOISE IN DIGITAL SYSTEMS 260
6.1 Noise Sources in a Digital System 261
6.2 Power Supply Noise 263
6.2.1 Single Supply Noise 264
6.2.2 Differential Supply Noise 266
6.2.3 Internal and External Supply Noise 267
6.3 Cross Talk 267
6.3.1 Cross Talk to Capacitive Lines 268
Coupling to a Floating Line, 268 • Coupling to a Driven Line, 269 • Typical Capacitance Values, 270 • Capacitive Cross Talk Countermeasures, 271
6.3.2 Cross Talk to Transmission Lines 272
Capacitive and Inductive Coupling of Transmission Lines, 272 • Lumped Inductive Coupling, 274 • Near- and Far-End Cross Talk, 274 • Typical Coupling Coefficients, 276 • Transmission Line Cross Talk Countermeasures, 277
6.3.3 Signal Return Cross Talk 278
6.3.4 Power Supply Cross Talk 280
6.4 Intersymbol Interference 280
6.4.1 Impedance Mismatch and Reflections 281
6.4.2 Resonant Transmitter Circuits 282
6.4.3 Inertial Delay and Hidden State 282
6.5 Other Noise Sources 285
6.5.1 Alpha Particles 286
6.5.2 Electromagnetic Interference 288
6.5.3 Process Variation 288
Typical Process Variations, 289 • Inverter Offset, 289 • Inverter Compensation, 290 • Differential Pair Offset, 290
6.5.4 Thermal (Johnson) Noise 291

6.5.5 Shot Noise 291
6.5.6 Flicker or $1/f$ Noise 292
6.6 Managing Noise 292
6.6.1 Bounded Noise and Noise Budgets 292
Proportional Noise Sources, 293 • Fixed Noise Sources, 294 • Overall Noise Budgets, 295
6.6.2 Gaussian Noise and Bit Error Rates 296
6.7 Bibliographic Notes 298
6.8 Exercises 298

7 SIGNALING CONVENTIONS 304
7.1 A Comparison of Two Transmission Systems 306
7.1.1 Signal Energy and System Power 308
7.1.2 Noise Immunity Versus Noise Margin 308
7.1.3 Delay 311
7.1.4 Discussion 312
7.2 Considerations in Transmission System Design 313
7.3 Signaling Modes for Transmission Lines 314
7.3.1 Transmitter Signaling Methods 315
Current-Mode Transmission, 315 • Voltage-Mode Transmission, 316 • Transmitter Signal-Return Cross Talk, 316 • Bipolar Versus Unipolar Signaling, 318 • Transmitter-Generated References, 319
7.3.2 Receiver Signal Detection 320
Generating the Receiver Reference, 320 • Receiver Return Cross Talk, 321 • Power Supply Noise, 322
7.3.3 Source Termination 323
Noise Considerations, 325 • Power Dissipation, 325 • Current-Mode Source Termination, 326
7.3.4 Underterminated Drivers 327
7.3.5 Differential Signaling 328
Symmetric Transmission Lines, 330
7.4 Signaling Over Lumped Transmission Media 331
7.4.1 Signaling Over a Capacitive Transmission Medium 331
Voltage-Mode Signaling, 332 • Current-Mode Signaling, 333 • Resistive Voltage Divider, 333 • Pulsed Signaling, 334 • Return-to-Zero (Precharged) Pulsed Signaling, 335 • Band-Limited Pulsed Signaling, 336 • References, 336
7.4.2 Signaling over Lumped LRC Interconnect 337
Rise-Time Control, 339 • Adding Parallel Termination, 342 • Reducing Power Supply Noise 343

7.5 Signal Encoding 344
7.5.1 Number of Signal Levels 345
7.5.2 Signal Magnitude 347
Hysteresis, 347
7.5.3 Signal Transfer Function 348
7.5.4 Error Correcting Codes 349
7.5.5 Pulsed Signaling 350
7.5.6 Signal Level and Delay 351
7.6 Bibliographic Notes 352
7.7 Exercises 353

8 ADVANCED SIGNALING TECHNIQUES 356
8.1 Signaling over RC Interconnect 357
8.1.1 Circuit Model 357
8.1.2 Repeaters 359
8.1.3 Increasing Wire Width and Spacing 360
8.1.4 Overdrive of Low-Swing RC Lines 361
8.2 Driving Lossy LC Lines 362
8.2.1 The Lone Pulse 363
8.2.2 Equalization of LRC Lines 364
8.3 Simultaneous Bidirectional Signaling 366
8.3.1 Current-Mode Bidirectional Signaling 367
8.3.2 Bidirectional Signaling Waveforms 368
8.3.3 Differential Simultaneous Bidirectional Signaling 368
8.3.4 Voltage-Mode Simultaneous Bidirectional Signaling 370
8.3.5 Reverse-Channel Cross Talk 371
8.4 AC and N of M Balanced Signaling 372
8.4.1 Terminology 372
DC Offset, 372 • Run Length, 373 • Disparity or Digital-Sum Variation (DSV), 374
8.4.2 Codes for DC-Balancing Signals 374
Nonoverlapping Block Codes, 374 • Running-Disparity Encoding, 375 • Framing, 377 • Burst-Error Length, 377 • The 8b/10b Code, 378 • DC Restoration, 379
8.4.3 Spatial N of M Balanced Signaling 379
Coding Efficiency, 380 • Systematic Encoding, 380
8.5 Examples 381
8.5.1 Logic Signaling 382
8.5.2 SRAM Bit Lines 383
8.5.3 Signaling Over Long On-Chip Wires 385

8.5.4 Signaling Chip-to-Chip on a Board 387
8.5.5 Signaling across a Cable 389
8.6 Bibliographic Notes 392
8.7 Exercises 392

9 TIMING CONVENTIONS 394
9.1 A Comparison of Two Timing Conventions 396
9.1.1 Skew and Jitter Analysis 398
9.1.2 Allowable Clock Rates 399
9.1.3 Discussion 400
9.2 Considerations in Timing Design 400
9.3 Timing Fundamentals 401
9.3.1 Timing Nomenclature 401
Delay and Transition Times, 401 • Periodic Signals, 401 • Maximum Absolute Value, Peak-to-Peak, and RMS, 403
9.3.2 Timing Properties of Delay Elements 403
9.3.3 Timing Properties of Combinational Logic 405
9.3.4 Timing Properties of Clocked Storage Elements 406
Edge-Triggered Flip-Flop, 406 • Level-Sensitive Latch, 407 • Double-Edge-Triggered Flip-Flop, 408
9.3.5 The Eye Diagram 409
9.4 Encoding Timing: Signals and Events 411
9.4.1 Encoding Aperiodic Events 411
Dual-Rail Signaling, 411 • Return-to-Zero (RZ)/Nonreturn-to-Zero (NRZ) Signaling, 412 • Clocked Signaling and Bundling, 412 • Ternary Signaling, 413
9.4.2 Encoding Periodic Signals 413
Required Transition Frequency, 414 • Bit Stuffing, 414 • Phase-Encoding, 414
9.5 Open-Loop Synchronous Timing 415
9.5.1 Global Clock, Edge-Triggered Timing 416
Minimum Delay Constraint, 416 • Maximum Delay Constraint, 417
9.5.2 Level-Sensitive Clocking 418
Basic Two-Phase Clocking, 418 • Borrowing Time, 419 • Effect of Skew, 420 • Qualified Clocks, 420 • Signal Labeling for Two-Phase Clocking, 421 • Single-Phase or Zero Nonoverlap Clocking, 422
9.5.3 Pipeline Timing 423
Optimum Clock Delay, 425 • Level-Sensitive Pipeline Timing, 426 • Pipelines With Feedback, 427

9.6 Closed-Loop Timing 428

9.6.1 A Simple Timing Loop 428
Residual Error, 429 • Loop Dynamics, 429

9.6.2 Phase Comparators 430
Flip-Flop Phase Comparator, 431 • Exclusive–OR (XOR) Phase Comparator, 433 • Sequential Phase and Frequency Comparator, 434

9.6.3 Variable Delay Line 436

9.6.4 Bundled Closed-Loop Timing 436
Canceled and Uncanceled Sources of Timing Uncertainty, 438 • Integrating Receivers, 438

9.6.5 Per-Line Closed-Loop Timing 439

9.6.6 Phase-Locked Loops 441
Voltage-Controlled Oscillators, 442 • Frequency Comparator, 443 • Loop Dynamics and Loop Filter, 444 • Reducing Jitter with a Phase-Locked Loop, 446

9.6.7 Oversampling Clock Recovery 447

9.7 Clock Distribution 449

9.7.1 Off-Chip Clock Distribution 449
Clock Distribution Trees, 450 • Phase-Locked Clock Distribution Networks, 451 • Salphasic Clock Distribution, 452 • Round-Trip Distribution, 453

9.7.2 On-Chip Clock Distribution 454
On-Chip Clock Trees, 454 • Mesh Distribution, 456 • Jitter in On-Chip Clock Distribution, 457

9.8 Bibliographic Notes 458

9.9 Exercises 458

10 SYNCHRONIZATION 462

10.1 A Comparison of Three Synchronization Strategies 463

10.2 Synchronization Fundamentals 465

10.2.1 Uses of Synchronization 466
Arbitration of Asynchronous Signals, 466 • Sampling Asynchronous Signals, 466 • Crossing Clock Domains, 467

10.2.2 Synchronization Failure and Metastability 468
Synchronizer Dynamics and Synchronization Time, 468 • Metastability, 469 • Probability of Synchronization Failure, 469 • Example Synchronizer Calculation, 470 • Completion Detection, 470 • Common Synchronizer Mistakes, 471

10.2.3 Clock Domains 472

Independent Clock Rates, 473 • Simplified Clock Distribution, 473 • Pipelined Signal Timing Eliminates Cable Delay Constraints, 473 • Aperiodic Clocks, 473

10.2.4 Classification of Signal–Clock Synchronization 473

10.3 Synchronizer Design 475

10.3.1 Mesochronous Synchronizers 475
Delay-Line Synchronizer, 475 • Two-Register Synchronizer, 476 • FIFO Synchronizer, 477 • Brute-Force Synchronization, 480

10.3.2 Plesiochronous Synchronizers 480
A Plesiochronous FIFO Synchronizer, 480 • Data Rate Mismatch, 481 • Detecting Phase Slip, 482 • Null Symbols and Flow Control, 483

10.3.3 Periodic Asynchronous Synchronizers 483
Clock Predictor Circuit, 484 • Periodic Synchronizer, 484

10.3.4 General Purpose Asynchronous Synchronizers 485
Waiting Synchronizer, 485 • Asynchronous FIFO Synchronizer, 485

10.4 Asynchronous Design 486

10.4.1 Stoppable Clocks 487

10.4.2 Asynchronous Signaling Protocols 489
Four-Phase Asynchronous Signaling, 489 • Two-Phase Asynchronous Signaling, 489 • The Weak Conditions, 490

10.4.3 Asynchronous Module Design Methods 492
State Diagrams, 492 • Concurrency and Choice, 494 • Trajectory Maps for Designing Asynchronous Sequential Logic, 495 • Set–Reset Excitation Equations, 497 • Arbitration and Circuits with Choice, 498 • Delay-Insensitive versus Matched-Delay Modules, 500

10.4.4 Composition of Asynchronous Circuits 502
Asynchronous Combinational Blocks, 502 • Align Blocks and Self-Timed Pipelines, 503 • Cyclic Asynchronous Circuits, 508

10.5 Bibliographic Notes 510

10.6 Exercises 511

11 SIGNALING CIRCUITS 514

11.1 Terminations 515

11.1.1 On-Chip Versus Off-Chip Termination 515

11.1.2 FET Terminations 516

11.1.3 Adjustable Terminators 517
Digital Versus Analog Adjustment, 518 • Binary Versus Thermometer Digital Adjustment Codes, 519

11.1.4 Automatic Terminator Adjustment 519
Automated Adjustment Controllers, 520 • Thermometer-Coded Controllers, 521 • Self-Series Termination Control, 521

11.2 Transmitter Circuits 522

11.2.1 Voltage-Mode Driver 523
Break-Before-Make Action, 524 • Pulse-Generating Driver, 525 • Tristate Driver, 525 • Open-Drain Outputs, 526

11.2.2 Self-Series-Terminating Drivers 528

11.2.3 Current-Mode Drivers 529
Saturated FET Driver, 529 • Current-Mirror Drivers, 530 • Differential Current-Steering Driver, 530 • Bipolar Current-Mode Drivers, 532

11.2.4 Rise-Time Control 533
Segmented Current Driver, 533 • The Problem with RC Rise-Time Control, 533 • Segmented Voltage Driver, 534 • Segmented Self-Series Terminated Driver, 535

11.2.5 Drivers for Lumped Loads 536
On-Chip Drivers for Capacitive Loads, 536 • Off-Chip Drivers for LRC Loads, 537

11.2.6 Multiplexing Transmitters 537

11.3 Receiver Circuits 540

11.3.1 Receivers Using Static Amplifiers 542
The Inverter As a Receiver, 542 • Source-Coupled FET Receivers, 543

11.3.2 Receivers Using Clocked Differential Amplifiers 544

11.3.3 Integrating Amplifiers 545
An Integrating Amplifier, 545 • Receiver Impulse Response, 545 • A Matched-Filter Receive Amplifier, 546

11.3.4 Demultiplexing Receivers 547

11.4 Electrostatic Discharge (ESD) Protection 548

11.4.1 ESD Failure Mechanisms 550
Field-Induced Failures, 550 • Thermally Induced Failures, 551

11.4.2 ESD Protection Devices 552
Primary Shunt, 552 • Series Resistor, 555 • Secondary Shunt 556 • Protecting Output Drivers, 556 • Guard Rings, 557 • Wiring and Contacting, 558

11.5 An Example Signaling System 559

11.5.1 Transmitter 559
Multiphase Clock Generator, 559 • Output Driver, 560 • Bias Generator, 561 • Predriver, 562 • Latches and Pass-Gate Clocking Network, 563 • Package Model, 563 • Transmission Line Model, 564 • Simulation Results for Package and Transmission-Line Models, 564 • Termination Schemes, 565 • Effectiveness of Slew-Rate Control, 566 • Noise Modeling, 566

11.5.2 Receiver 567
Phase Shifter and Multiphase Clock Generator, 568 • Samplers, 569 • Retiming Latches, 569 • Clock Adjuster, 569

11.6 Bibliographic Notes 571

11.7 Exercises 571

12 TIMING CIRCUITS 573

12.1 Latches and Flip-Flops 574

12.1.1 Level-Sensitive Latches 574
Dynamic Latches, 574 • CMOS Static Storage Element, 576 • CMOS Static Latches, 577

12.1.2 Edge-Triggered Flip-Flops 579
Auxiliary Control Inputs, 580 • True Single-Phase-Clocked (TSPC) Flip-Flops, 581 • Differential Edge-Triggered Flip-Flop, 582 • Double-Edge-Triggered Flip-Flops, 582

12.1.3 Failure Mechanisms in Flip-Flops and Latches 583
Race-Through, 583 • Dynamic Node Discharge, 584 • Power Supply Noise, 586 • Clock Slope, 586 • Charge Sharing, 588

12.2 Delay Line Circuits 589

12.2.1 Inverter Delay Lines 589
Delay Adjustment Range, 591 • Power-Supply Rejection in Inverter Delay Elements, 592 • Inverters with Regulated Supply Voltage, 593

12.2.2 Differential Delay Elements 593
Adjustable PFET Resistor, 594 • Replica-Biased Delay Line, 595 • Adjustment Range for Replica-Bias

Delay Lines, 595 • Static Supply Sensitivity for the Replica-Biased Delay Stage, 596 • Dynamic Supply Sensitivity, 597

12.2.3 Circuit and Layout Details 598
Replica Control Loop Stability, 598 • Power Routing and Bypassing, 600 • Matching and Balancing, 601 • Substrate Noise, 602

12.2.4 Other Differential Timing Components 603
Small-Swing to Full-Swing Buffers, 603 • Interpolators, 604 • Duty-Cycle Correctors, 606 • Clock Input Conditioning, 607

12.3 Voltage-Controlled Oscillators 608

12.3.1 First-Order Oscillators 608
Array Oscillators, 609 •

12.3.2 Second-Order Oscillators 610
Crystal Oscillators, 610 • Frequency Multiplication, 613 • Lumped-Element Oscillators, 613

12.4 Phase Comparators 615

12.4.1 XOR Comparator 615

12.4.2 Edge-Triggered Flip-Flop Phase Detector 617

12.4.3 Sequential Phase Detectors 617

12.5 Loop Filters 620

12.5.1 RC Loop Filters 621

12.5.2 Charge Pump Filters 624
Charge Pump Control Voltage Ripple, 626 • Self-Biased Loop Filters, 627

12.5.3 Delay-Locked Loop Filters 629
Self-Biased DLL Loop Filter, 630 • Switched-Capacitor Loop Filters, 630 • Loop Initialization, 631 • "Turbo" Mode, 632 • "Bang-Bang" Controllers, 633 • Digital Loop Controllers, 633

12.6 Clock Aligners 635

12.6.1 PLL Versus DLL Implementations 635

12.6.2 Simple DLL-Based Aligners 636

12.6.3 Phase-Based Aligners 638

12.6.4 A Hybrid Phase/Delay-Based Clock Aligner 639

12.7 Bibliographic Notes 640

12.8 Problems 641

Index 653

PREFACE

Today we are on the brink of a major change in the way digital systems are designed. In the past, many digital designers successfully ignored electrical issues, designing their systems in terms of discrete logic. Only the designers of supercomputers and high-speed communication systems needed to look past the digital abstraction. This is no longer the case. As technology has advanced, the systems-level engineering problems of digital systems have become critical to the success of all digital systems.

As technology continues to advance, issues of signaling, timing, power, and noise become increasingly important. At high signaling frequencies, wires can no longer be considered equipotential regions and must be modeled as transmission lines. The number and speed of gates on a chip have increased faster than the number and speed of pins, making inter-chip communication a system bottleneck and placing a premium on efficient signaling and timing conventions. With reduced supply voltages and higher currents, power distribution becomes a more challenging engineering problem. At high frequencies, timing conventions must be carefully designed to prevent skew and jitter of clock signals from degrading system performance. Because of these trends, the techniques described in this book, once used only in supercomputers, are now essential to the correct and efficient operation of any type of digital system.

We were motivated to write this book for two reasons. First, in our research building high-speed digital systems (parallel computers, network switches, and high-performance graphics systems) we developed a number of signaling, timing, and power distribution techniques that overcame the limitations of conventional methods. We were eager to share these methods and the engineering science behind them with other practicing engineers. Our second motivation was to avoid repetitions of many disasters we encountered in our interactions with industry. Over a year was spent at one company chasing noise problems before a system would operate reliably. Another system went through several iterations of ASICs due to timing problems. A third system failed periodically due to on-chip power supply fluctuations. A fourth system product was delayed by six months because of a subtle failure in the flip-flops used throughout a custom chip design. These problems delayed system delivery by months to years and in some cases directly contributed to the failure of companies. Band-aid fixes rarely exist for these types of problems; however, they could have been easily avoided by proper design if the engineers involved had been knowledgeable about noise, signaling, timing, and

power. By writing this book, we hope to help eradicate the widespread ignorance, and often misinformation, in these areas and in doing so, help avoid disasters of this kind in the future.

Organization

The book begins by describing the major engineering problems of digital systems: power, noise, signaling, and timing. It presents the best known techniques for dealing with these problems and describes working circuits that implement these techniques. At all levels material is developed rigorously, deriving equations from first principles. The book avoids the handbook approach of describing how things are usually done. While it gives a good description of current industrial practice, it emphasizes why things are done, points out the limitations of current approaches, and describes emerging techniques, such as low-voltage signaling and closed-loop timing, that are likely to be how things are done in the future.

The book introduces the topic of digital systems engineering by describing the major engineering problems associated with digital systems and the technology trends affecting them. The book then devotes three chapters laying the groundwork for the treatment of these problems by describing computer elements: wires and circuits. These chapters examine the components and interconnections used to build modern digital systems and develop engineering models of these components to facilitate the study of systems engineering problems in later chapters.

After this preamble, the book deals with the problem of power distribution in Chapter 5. Digital logic requires a stable, quiet, DC supply voltage while drawing a large AC current with frequency components comparable to signal rise times. This chapter develops circuit models for both on-chip and off-chip power distribution networks and loads. Off-chip distribution methods including bypass capacitors, local regulation, shunt regulators, and clamps are discussed, as are on-chip distribution methods including power grid layout, on-chip regulation, and symbiotic bypass capacitors. The chapter closes with the presentation of an example distribution system. Since power supply noise is one of the largest noise sources in digital systems, this chapter lays the groundwork for the treatment of noise in Chapter 6, which in turn lays the groundwork for the discussion of signaling conventions in Chapter 7.

Noise, the topic of Chapter 6, is one of the most important factors in the engineering of digital systems, yet it is also one of the least understood. Most noise in digital systems is created by the system itself, and hence is truly interference and not noise. This chapter gives an in-depth treatment of major noise sources including power supply noise, signal return coupling, crosstalk, inter-symbol interference, and parameter offsets. Lower-level noise sources including alpha particles, thermal noise, shot noise, and flicker noise are briefly treated. The engineering models of these noise sources are then used to introduce the use of noise budgets, the concept of noise immunity, and the calculation of error rate based on statistical noise models. This treatment of noise prepares the reader to understand the major problem in the design of signaling systems presented in Chapters 7 and 8.

Signaling, the method used to transmit a bit of information from one location to another, is central to the design of digital systems. A signaling convention involves encoding information into a physical quantity (typically current or voltage), providing a reference against which this quantity is measured, the design of transmitters to couple signal energy into the transmission medium and terminators to absorb energy and prevent unwanted reflections, and a method for controlling signal transitions to limit the spectrum of transmitted energy. The signaling convention determines to a large extent the reliability, speed, and power consumption of a system. A good signaling convention isolates a signal from noise, providing noise immunity, rather than attempting to overpower the noise with noise margin. Most signaling conventions in common use are quite poor, based on standards that are accidents of history rather than on careful design to isolate noise sources. For this reason, many modern systems define their own signaling conventions rather than employ these standards.

Chapter 7 deals with the basics of signaling over LC transmission lines and lumped LRC interconnect. This chapter introduces the concepts of voltage-mode and current-mode transmission, unipolar and bipolar signaling, series and parallel termination, references, and differential signaling. Undertermination, rise-time control, pulsed signaling, and multi-level signaling are also described. We develop methods for calculating the signal magnitude required to transmit a signal reliably in the presence of noise and show that, with adequate noise isolation, very low voltage signal swings (100 mV) are sufficient. Chapter 8 describes more advanced concepts including techniques for dealing with lossy RC and LRC transmission lines, simultaneous bidirectional signaling, and AC signaling.

A digital system uses a timing convention, along with a signaling convention to govern when to place a new symbol on a signal line and when to sample the line to detect the symbol. A good timing convention is one that maximizes performance by reducing and compensating for sources of timing noise, skew and jitter. It is the uncertainty in the timing of a signal, not the delay, that limits the rate at which a system can operate. Chapter 9 discusses the fundamentals of timing conventions. The methods used to encode timing on signals are described and the concepts of timing noise and timing budgets are introduced. The chapter goes on to develop methods for minimizing skew and jitter using both open-loop and closed-loop timing. A discussion of clock distribution techniques closes the chapter.

Closely related to timing is the problem of synchronization, determining the relative order of two events, which is discussed in Chapter 10. Synchronization is required when sampling asynchronous inputs into a synchronous system, or when a signal traverses the boundary between two clock domains. When a signal is synchronized there is some probability that the synchronizer will be driven into a metastable state, unable to resolve which event came first, and stay in that state long enough to cause a synchronization failure. This chapter introduces the problem of synchronization, describes metastability, synchronization failure, and gives methods to calculate the probability of synchronization failure. A section on synchronizer design recognizes several special cases of synchronization and describes how to build fast synchronizers that exploit these special cases to

avoid the normal synchronizer delays. Finally, the chapter describes methods for asynchronous design that avoid the problem of synchronization.

Circuits for signaling and timing are described in Chapters 11 and 12 respectively. Chapter 11 presents working circuits for the transmitters, receivers, and terminations needed to build the signaling systems described in Chapters 7 and 8. Particular attention is paid to techniques for managing noise and compensating for process, voltage, and temperature variations. Timing circuits are described in Chapter 12 including flip-flops, delay lines, VCOs, phase comparators, and clock drivers. The emphasis here is on minimizing skew and jitter. Each of these chapters closes by describing an example system that ties together the concepts of Chapters 7 through 10 with the circuit details of Chapters 11 and 12.

Teaching Digital Systems Engineering

The best way to empower engineers with mastery of digital systems engineering is to teach this material as part of the undergraduate curriculum in electrical engineering and computer engineering. There is currently a gap in the curriculum between circuit design, which covers the electrical design of individual logic and memory elements, and logic design and architecture, which deal with the logical organization of digital systems. System-level electrical issues, while critically important, are absent from the curriculum.

This book is intended to fill this gap in the electrical engineering and computer engineering curriculum. It is written as a textbook with the material presented in an order suitable for teaching and with exercises at the end of each chapter. The material is at a level appropriate for seniors or first-year graduate students in electrical engineering. Drafts of the book have been used to teach digital systems engineering courses at MIT (by Dally) and Washington University (by our colleague Fred Rosenberger). Starting with Autumn Quarter 1998, a course on digital systems engineering based on this book, EE273, will be offered at Stanford University.

Supplementary teaching materials, including course schedules, lecture slides, simulation models, problem sets and solutions, will be available via the worldwide web at http://www.cambridge.org/9780521592925

We learned a great deal and had a great deal of fun in the process of creating this book. We hope that you get at least a bit of the insight and enjoyment out of reading it that we got out of writing it.

William J. Dally, *Stanford, California*
John W. Poulton, *Chapel Hill, North Carolina*

ACKNOWLEDGMENTS

We would like to thank all of the people who have contributed to the creation of this book. Fred Rosenberger at Washington University carefully reviewed the draft manuscript and was the first brave soul (other than the authors) to teach a course from this book. His insightful comments greatly improved the quality of the resulting book.

Many people reviewed one or more draft chapters. Of these people, Tom Knight (MIT), Gill Pratt (MIT), and Rich Lethin (Equator Technologies) deserve special thanks for suggesting significant improvements.

The students who took 6.891 at MIT during Fall Semester 1996 helped greatly to debug the material and an early draft of the text.

Steve Tell and Robert Palmer at UNC performed many of the measurements that are presented in Chapter 3.

Our view of this material, as presented here, has been greatly influenced by interactions with our colleagues including (in no particular order) Mark Horowitz (Stanford), Kunle Olukotun (Stanford), Tom Lee (Stanford), Ken Yang (Stanford), Vernon Chi (UNC), John Eyles (UNC), Rich Brown (Michigan), Tom Knight (MIT), Gill Pratt (MIT), Steve Ward (MIT), Anant Agarwal (MIT), Chuck Seitz (Myricom), Al Barber (HP), Dan Dobberpuhl (Digital), Steve Oberlin (Cray), Mark Birrittella (Cray), Dan Mansour (Cray), Doug Kervin (Cray), Dave Dunning (Intel), Craig Peterson (Intel), Randy Mooney (Intel), Greg Fyler (Intel), Greg Papadopoulos (Sun), Lance Glasser (KLA), Phil Carvey (Avici), Steve Molnar (IDT, Inc.), Trey Greer (PixelFusion, Ltd), and Fred Heaton (Secant Network Technologies).

We learned much from the students and staff that worked with us on research projects related to this work including Larry Dennison, Whay Lee, Duke Xanthopoulos, Mike Noakes, Jeff Bowers, Steve Keckler, Andrew Chang, Ed Lee, and Richard An.

Alan Harvey at Cambridge University Press helped us throughout the project. Our project manager Andrew Wilson and our copy editor John Joswick took a very rough manuscript and turned it into a polished book.

Gill Weigand, Bob Parker, and Bob Lucas at DARPA had the foresight to fund research in these areas before it became a pressing problem.

Finally, our families, Sharon, Jenny, Katie, and Liza Dally and Susan, Sarah, and Winona Poulton made significant sacrifices so that we could have the time to devote to writing.

1 INTRODUCTION TO DIGITAL SYSTEMS ENGINEERING

1.1 Why Study Digital Systems Engineering? 2
1.2 An Engineering View of a Digital System 4
- **1.2.1** Feeds and Speeds 5
- **1.2.2** Signaling Conventions 6
- **1.2.3** Timing and Synchronization 8
- **1.2.4** Power Distribution 10
- **1.2.5** Noise 10
- **1.2.6** A Systems View of Circuits 12

1.3 Technology Trends and Digital Systems Engineering 12
- **1.3.1** Moore's Law 12
- **1.3.2** Scaling of Chip Parameters 16
- **1.3.3** Scaling of Wires 17
- **1.3.4** High Levels of Integration Permit New Approaches 21
- **1.3.5** Digital Systems Problems and Solutions Continue to Change 21

1.4 Organization of this Book 22
1.5 Bibliographic Notes 22
1.6 Exercises 22

Digital systems are pervasive in modern society. We use computers for bookkeeping, engineering, publishing, and entertainment. Digital communications systems handle our telephone calls and enable our Web browsing sessions. Other uses of digital systems are less visible. Most consumer electronics products are largely digital and becoming more so. Music today is distributed digitally on compact optical disks, and video production is rapidly becoming a digital process. A typical appliance is controlled digitally by a microcomputer. As many as ten microcomputers can be found in the average car for controlling functions ranging from the sound system to the antilock brakes.

A *digital* system represents information with discrete symbols (of which digits are a special case) rather than with a continuously varying quantity, as in an *analog* system. Most systems use just two symbols, often denoted by the binary digits (or *bits*) 0 and 1, to represent all information. Simple truth propositions are

represented directly with a single bit, whereas strings of bits are used to represent more complex data.

In a digital system, noise below a given level can be completely rejected. Symbols are encoded into ranges of voltage or current level. If we add a small amount of voltage, V_N, to the nominal voltage for the 0 symbol, V_0, the resulting voltage, $V_0 + V_N$, will still be in the range for a 0 symbol and, more importantly, can be restored to the nominal level, V_0. This property allows us to process information through many noisy stages of logic with no accumulation of noise. In an analog system, in contrast, disturbing an information voltage, V_x, with noise gives a voltage, $V_y = V_x + V_N$, that represents a different piece of information. The analog signal cannot be restored and will be severely degraded after many stages of noisy processing.

All digital systems, whether used explicitly for computation or as part of an entertainment system, are constructed from three basic components: logic, memory, and communication channels. Logic operates on symbols, for example adding two numbers together or comparing two characters. Memory stores symbols, moving them in time so that information computed at one point in time can be recalled later. Communication channels, usually wires, move information in space so that values generated by logic in one location can be stored in memory, used by logic in a different location, or both.

The development of a digital system involves circuit and logic design, architecture, and systems engineering. Individual components such as logic gates and memory arrays are designed at the circuit level. At the logic level we compose these individual components into subsystems such as adders and finite-state machines. At a higher level the instruction-set and register-level organization of the system is governed by the principles of computer architecture.

This book addresses digital systems engineering. The systems-level engineering issues in a digital system include power distribution, noise management, signaling, timing, and synchronization. Power distribution deals with how to distribute a controlled DC voltage level throughout a system that draws considerable amounts of AC current. Transmitting digital symbols over wires at maximum speed and with minimum power is the challenge of signaling. Timing deals with how computations and communications are sequenced within and between modules. The design of signaling and timing conventions is dominated by considerations of noise or uncertainty; thus, the successful digital designer must understand noise sources and formulate methods for managing noise. Synchronization is required to coordinate the operation of two systems operating from separate time bases or to sample an input that may change asynchronously.

We address these issues from systems and circuits perspectives. For each topic, we start by giving the big picture with circuit details abstracted and then return to the topic and present detailed circuits.

1.1 WHY STUDY DIGITAL SYSTEMS ENGINEERING?

As technology has advanced, the systems-level engineering problems of digital systems have become more critical. At high signaling frequencies wires can no

longer be considered equipotential regions and must be modeled as transmission lines. On-chip wires are becoming more resistive, presenting a significant signaling challenge. The number and speed of gates on a chip have increased faster than the number and speed of pins, making interchip communication a system bottleneck and placing a premium on efficient signaling and timing conventions. With reduced supply voltages, higher currents, and thinner metal layers, power distribution is an increasingly challenging engineering problem. At high frequencies timing conventions must be carefully designed to prevent skew and jitter of clock signals from degrading system performance.

Careful attention to digital systems engineering issues makes an enormous difference in the performance, reliability, and power dissipation of a system. Too often, lack of attention to these issues results in a system that is unreliable or simply does not work. Recently, a major computer manufacturer delayed the release of a new system because several custom chips failed to operate at any speed owing to excessive clock skew. Another manufacturer had to revise several chips in a new system because noise problems led to unacceptable error rates on interchip signals.

Digital systems engineering has largely been igonored by the academic community. Most curricula include courses on circuit design of logic elements, logic design, register-transfer-level design, and high-level architecture. However, the engineering problems of composing circuits into systems are only briefly touched upon. A frequent misconception at the system level is that one must abstract away the electrical properties of the system and deal with discrete symbols at the logic or architectural level. To the contrary, careful management of electrical issues at the system level is critical to the success of any modern system.

In fact, to do a credible job of computer architecture one must have an intimate understanding of system-level engineering issues. Architecture deals with organizing a system and defining interfaces (e.g., instruction sets and channel protocols) to achieve cost and performance goals. System-level engineering constrains what the architect can do and is a major determinant of the cost and performance of the resulting system. It is far too easy for one to abstract these issues and architect a system that, owing to bandwidth limitations, is unrealizable or suboptimal.

Many in industry have taken ad hoc appoaches to systems engineering problems. Clocks are distributed by open-loop trees that are tuned until the circuit works. Power is distributed on wide metal lines. Full-swing signals are used for signaling on and between chips. Typically, a company addresses a systems engineering problem using the same approach taken with its last system. This often leads to a false sense of security. As technology advances, ad hoc approaches that worked in the past become suboptimal and often fail. Many companies have been caught unawares by excessive power supply noise, unreliable signaling, and badly skewed clocks. In one classic example, manufacturers designed standard logic components using the high-inductance corner pins of their packages for power and ground. This was originally done to simplify routing of printed circuit (PC) boards without power planes. Over the years, standard logic parts increased in speed until their switching current induced so much noise across the supply

inductors that the output could not reliably be detected. Only after many parts failed in systems did the manufacturers relocate the pins to the center of the package.

In this book we will take an engineering science approach to systems engineering problems; that is, we will study the underlying physical and electrical mechanisms that relate to power distribution and signaling and timing conventions. We will develop models for these mechanisms and use these models to analyze the performance of potential solutions. We will present specific solutions to these problems that work under today's constraints, and, more importantly, we will give you the tools to understand why these solutions work and to evaluate whether they will continue to work under future constraints.

Standards, both published and de facto, have historically played a large role in digital engineering. The use of full-swing underterminated voltage-mode signaling and edge-triggered synchronous timing is almost universal even though, as we will see, there are far better signaling and timing alternatives. The popularity of the prevailing approaches derives from the many catalog parts available using these conventions and to a lesser extent to the lack of understanding about alternatives and the criteria for evaluating them.

The trend toward building digital systems from custom or semicustom very large scale integrated (VLSI) components such as application specific integrated circuits (ASICs) has lessened the importance of catalog parts, except for memories, and hence enables the system designer to control signaling and timing conventions. Because these ASICs largely communicate with one another, the system designer is free to choose any convention she wishes for communication between these components. The designer is also free to choose the conventions employed by and between the components on a single chip. Thus, at the same time that continued advances in VLSI fabrication technology drive many of the challenging systems problems, VLSI also offers an opportunity to solve these problems by adding degrees of freedom in the choice of signaling and timing conventions.

1.2 AN ENGINEERING VIEW OF A DIGITAL SYSTEM

An engineer views a digital system in terms of information flow, power flow, and timing. For example, Figure 1-1 is an engineering view of a node of a modern multicomputer. The arrows in the figure illustrate the information flow and are annotated with the signaling convention employed, and the individual components are annotated with their power and timing information. As we shall see, these are not independent variables. The choice of a signaling convention, for example, can determine both the available bandwidth from a component and the power it requires.

The heart of the system is a central processing unit (CPU) operating at 500 MHz and dissipating 80 W of power from a 2.5-V supply. The CPU is connected to a set of cache chips via data and address lines and to a controller chip via two unidirectional channels. The controller in turn connects via channels to a router,

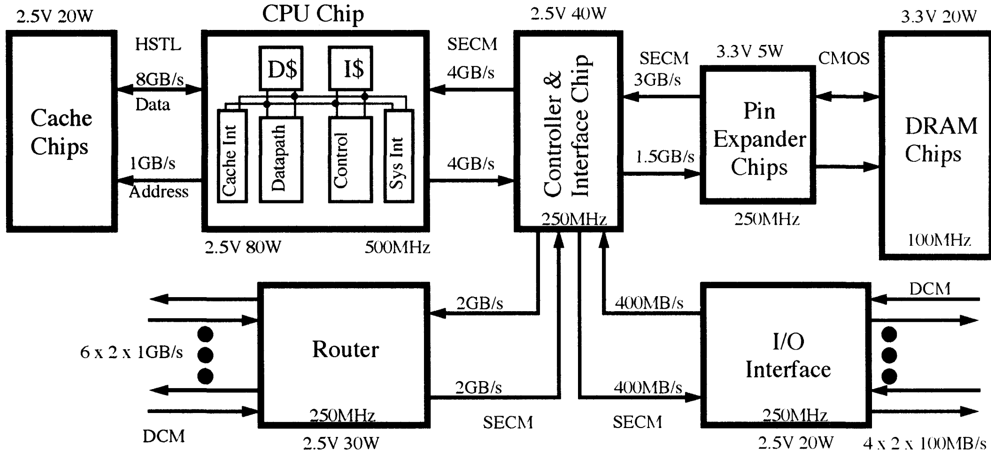


FIGURE 1-1 A Systems View of a Multicomputer Node

an I/O interface, and, via some pin expanders[1] to a set of dynamic random access memory (DRAM) chips. The router has six bidirectional channels that connect to the routers on other nodes, and the I/O interface has four channels that connect to peripherals.

1.2.1 Feeds and Speeds

Much of the art of digital systems design is managing the movement of information. A good design balances demands for bandwidth against the constraints of limited numbers of chip and module pins and limited amounts of wiring area on chips and boards. The designer controls the partitioning of the system and the topology of connections to strike the appropriate balance. This is a critical task. The efficient use of available bandwidth is probably the single most important determinant of overall system performance.

A key tool in the process of balancing a system is a diagram, like Figure 1-1, that shows the *feeds*, where information flows, and the *speeds*, how fast (in bits/s) the information flows along each path. This information is used in two ways. First, together with the signaling rate, the combined speeds into and out of each component or module indicate how many pins are required. Second, system performance can be predicted from the bandwidths at various points in the system in conjunction with an architectural model (usually a simulator).

For example, suppose our architectural model indicates that the bandwidth from the CPU to the second-level cache should equal the bandwidth from the CPU to the rest of the system. Our packaging allows at most 300 signal pins on

[1] A pin-expander chip takes a few pins that use a high-speed signaling convention, like single-ended current mode (SECM) at 500 Mb/s, and converts them to a larger number of pins in a slower signaling convention such as full-swing complementary metal-oxide semiconductor (CMOS) at 100 Mb/s.

TABLE 1-1 **Pin Counts for System of Figure 1-1**

Chip	Signal	Speed (GB/s)	Rate (Mb/s-pin)	Pins
CPU	Cache data	8	500	128
	Cache address	1	500	16
	Channels	8	500	128
	Total			272
Controller	CPU channels	8	500	128
	Router channels	4	500	64
	Memory channels	4.5	500	72
	I/O channels	0.8	500	14
	Total			278
Router	Controller channels	4	500	64
	Router channels	12	250	384
	Total			448

a component. Our signaling convention dictates that all signals must be point-to-point. Signals on the board operate single-ended at 500 Mb/s-pin, and signals that go off the board (router and I/O outputs) operate at differentially 500 Mb/s over a pair of pins for 250 Mb/s-pin. Using these numbers, Table 1-1 shows the number of pins required by the three major chips in the system.

The table shows that the CPU and controller chip are within the pin budget, and a few pins are left over. The router, however, is way over the pin budget, and thus the system as currently sketched cannot be implemented with the planned packaging. At this point the designer needs to repartition the system (see Exercise 1.1) to eliminate this bottleneck, reduce the router bandwidth, with possible performance penalties, or attempt to remove the constraint by getting a package with more pins or using a faster signaling convention.

1.2.2 Signaling Conventions

A signaling convention is the method used to encode symbols (usually 1 and 0) into physical quantities (e.g., voltage and current). An efficient signaling convention is one that maximizes the bandwidth per pin, minimizes power dissipation, and provides good noise immunity. Such a signaling convention can dramatically increase available bandwidth and hence system performance. Suppose, for example, the entire system in Figure 1-1 were implemented using a full-swing CMOS signaling convention that was limited to 100 Mb/s-pin. The CPU and controller chips would require about 1,400 signal pins! With CMOS signaling, a system of this level of performance simply could not be implemented.

Most of the channels between chips on the same PC board in Figure 1-1 use single-ended current-mode (SECM) signaling. This convention encodes the symbol 1 as a +2.5-mA current and a 0 as a −2.5-mA current. The wire over

which a signal travels is designed to be a 50-Ω transmission line and is parallel-terminated with a 50-Ω resistor on the receiving chip into a midrail (1.25-V) supply. This gives a voltage swing of 250 mV, from 1.125 to 1.375 V. The receivers detect the arriving signal by measuring the voltage across the termination resistor with a sensitive amplifier.

In contrast, most systems today use a form of full-swing signaling. For example, the CMOS signaling convention, as employed by the DRAMs in Figure 1-1, encodes 1 as the positive supply voltage (3.3 V) and 0 as the negative supply voltage (0 V). The impedance of the wires is uncontrolled, and the lines are unterminated. A CMOS inverter is used to detect the received voltage in the reference frame of the receiver power supplies.

Why is current-mode signaling preferable to CMOS signaling? Current-mode signaling offers lower power, faster operation, and better noise immunity. The only advantage of CMOS signaling is that it is used by the vast majority of standard parts. To interface to such parts (e.g., the DRAMs in Figure 1-1) one must use this signaling convention.

1.2.2.1 Signaling Speed

Current-mode signaling offers faster operation because it employs incident wave signaling for minimum delay and uses a terminated transmission line to minimize intersymbol interference. Suppose the average on-board wire is 30 cm long. The current-mode signaling system injects a 250-mV signal at one end of this line, and 2 ns later the wave carrying this signal arrives at the termination and is absorbed into the terminator. No residual energy is left on the line to corrupt subsequent bits. Thus, the line can immediately be used to send a second bit. In fact, several bits can be pipelined on a long wire. By getting the symbol on and off the line quickly, current-mode signaling enables fast signaling rates.

In contrast, the average CMOS driver, with an output impedance of 200 Ω cannot drive the 50-Ω line through 3.3 V in a single step. To do so would require a very low output impedance and enormous current (66 mA). Instead this driver injects a 660-mV (13-mA) signal into the line. After 2 ns, this wave reflects off the receiving end of the line, giving 1.3 V. Only after the wave reflects off the source and then the receiver for a second time is the line voltage sufficiently high to be detected as a logic 1. Even after 10 ns (2.5 round trips) the line is still over a half volt short of 3.3 V. This residual state leads to intersymbol interference. Because CMOS signaling takes a long time to get a symbol on the line, it is limited to signaling periods that are many times the delay of the line.

1.2.2.2 Signaling Power

By operating with a small signal swing, current-mode signaling dissipates much less power. The current-mode signaling convention draws 2.5 mA from a 2.5-V supply for a power dissipation of 6.2 mW. The actual signal power is ten times less, 620 μW (2.5 mA times the 250-mV signal swing). Operating at 500 Mb/s, the signal energy per bit is 1.25 pJ (2 ns times 620 μW), and the power supply energy per bit is 12.5 pJ.

In contrast, the CMOS system operates with a peak current of 13 mA and an average current of 9.4 mA (assuming the line toggles each cycle) from a 3.3-V supply for a power dissipation of 31 mW. Operating at 100 Mb/s, for the reasons described above, the signal energy per bit is 310 pJ or about 250 times that of the current-mode system![2] The power supply energy is the same here as the signal energy (about 25 times that of the current-mode system).

1.2.2.3 Signal Integrity

At this point you might think that with such a large signal swing the CMOS system must be much more reliable in the presence of noise. In fact the opposite is true. As we shall see in Chapter 7, the current-mode system has better noise immunity because it isolates the signal from external noise sources, such as power supply noise, and reduces internal noise sources, such as receiver sensitivity and intersymbol interference.

1.2.2.4 Other Signaling Conventions

The same arguments apply to signaling conventions on-chip. For example, using full-swing signaling to communicate between the instruction cache, I$, and control unit in Figure 1-1, would waste power. However, the constraints for on-chip signaling are different, and hence different solutions are appropriate.

Two other off-chip signaling conventions appear in the channel labels of Figure 1-1. A differential current-mode signaling convention (DCM) is used for all signals that leave the PC board to avoid the problems of signal-return cross talk that would otherwise occur owing to the large return impedance of typical off-board cables and connectors. The cache static random access memory (SRAM) chips are shown using HSTL,[3] an underterminated signaling convention (Section 7.3.4) that is just beginning to be offered in standard parts such as SRAMs.

1.2.3 Timing and Synchronization

A timing convention governs the sequencing of data through logic and across channels. The convention governs when signals can change and when they are sampled.

1.2.3.1 Synchronous Timing

Within a *clock domain*, an area where all signals have their timing referenced to a single clock signal, the convention may be built around edge-triggered flip-flops or level-sensitive latches. Some conventions, such as using multiple nonoverlapping clock phases, are very tolerant of clock skew, whereas others require very careful clock distribution to minimize skew.

[2] The fact that the CMOS system does not reach 3.3 V after 10 ns works in its favor here. The energy to charge a 30-cm, 50-pF line to 3.3 V is 545 pJ ($3.3^2 \times 50$ pF).

[3] HSTL is an acronym for high-speed transceiver logic.

For example, the system of Figure 1-1 employs a separate clock domain for the core logic of each chip. Most of the chips operate at 250 MHz, whereas the CPU operates at 500 MHz. The internal logic in each clock domain uses level-sensitive latches and a single-phase clock. Minimum delay constraints are used to relax clock skew tolerances.

1.2.3.2 Pipelined Timing

With all signals referenced to a single clock, the maximum operating frequency depends on the propagation delay through the logic or across the channel. In situations where data flow primarily in one direction and delays are predictable (e.g., most channels), substantially higher throughput can be realized by using *pipelined timing*. With pipelined timing, each pipeline stage operates in its own clock domain, and the phase of the clock is delayed to match the delay of the data. Therefore, the maximum throughput is independent of propagation delay and is limited only by the uncertainty in the delay: skew and jitter. In Figure 1-1, the channels between the controller and the other components all operate using pipelined timing, allowing operation with clock periods much shorter than the delay of the wires. Pipeline timing is also occasionally used to advantage in the design of pipelined arithmetic units. In this context, it is sometimes referred to as *wave-pipelining*.

1.2.3.3 Closed-Loop Timing

The static portion of delay uncertainty, skew, can be eliminated by using a control loop that measures the relative delay or phase of two signals and adjusts a variable delay or frequency element to match delays. This form of closed-loop timing allows operation at very high frequencies without requiring very tight tolerances or matching of wire and logic delays. The dynamic portion of uncertainty, jitter, can be reduced by using a phase-lock loop that tracks the low-frequency variations in an incoming signal while rejecting high-frequency jitter.

1.2.3.4 Clock Distribution

Distributing a clock over a large clock domain with low skew and controlled duty factor is a challenging engineering problem. Off-chip, clock trees with matched transmission lines and buffers are typically used. On-chip, the problem is more difficult, for the lossy nature of on-chip wires leads to large diffusive delays, and power supply variations modulate the delay of clock buffers. The CPU chip in Figure 1-1, for example, uses a six-level clock tree with the leaves shorted in a low-resistance, two-dimensional grid to control clock skew across the chip to within 300 ps. This skew is large compared with the typical gate delay (100 ps). However, by using a timing convention that employs a two-phase nonoverlapping clock driving transparent latches and that constrains the minimum delay between latches, the design easily tolerates the 300-ps skew without increasing the cycle time and without any danger of hold-time violations. In designing the timing of a system, an engineer chooses between methods to control skew and methods that tolerate skew. A well-designed system usually balances the two approaches.

1.2.3.5 Synchronization

Before signals from different clock domains can be combined, they must be *synchronized* to a common clock. For example, the receiving side of each channel in the controller chip of Figure 1-1 operates in its own clock domain as a result of the pipelined timing employed. Before these signals can be combined in the core of the controller, they must be synchronized to the core clock. In general, synchronizing a signal to a clock requires a delay as the synchronizer waits for metastable states to decay. However, the synchronization being performed in Figure 1-1, and in most digital systems, is a special case in that the events being synchronized are periodic. Thus, future transitions can be predicted in advance and the signals synchronized without delay.

1.2.4 Power Distribution

A digital system requires a stable DC supply voltage, to within a few hundred millivolts, to ensure proper operation of logic and communication circuits. The power distribution system must provide this steady voltage in the presence of very large AC current demands. The resistive nature of on-chip wires and the inductance inherent in most packaging elements make this a difficult problem.

A modern CMOS circuit can vary its current draw from 0 to a maximum value in a fraction of a clock cycle. In the system of Figure 1-1, for example, the CPU dissipates 80 W at 2.5 V for an average current of 32 A. The peak current is easily twice this amount, 64 A, and current demand can readily change from zero to this maximum value in half a clock cycle, 1 ns, for a peak derivative of $di/dt = 64$ GA/s. Even with hundreds of supply pins (of a few nH each), this current transient induces an unacceptable voltage transient across the parasitic inductance of the package. On-chip bypass capacitors and perhaps regulators are required to manage this transient.

A typical power distribution system is a hierarchy. Small local elements, like on-chip bypass capacitors and regulators, provide small amounts of energy to local regions and handle the high-frequency components of transients. Larger elements supply larger regions and handle lower-frequency components of the transients. Because of their physical distance from the point of use, and the inductance that implies, they are not able to manage the high-frequency transients. At higher levels of the hierarchy, the supply voltage is usually raised to allow distribution to be performed with lower currents and hence smaller and less expensive bus-bars and cables.

1.2.5 Noise

Noise is a major concern in the engineering of digital systems. It corrupts signals on channels between modules, disturbs the state of logic networks and memory cells, and adds jitter and skew to the timing of signals. Signaling and timing conventions are designed around the constraints of system noise. For example, both the amplitude of signal swing and the clock cycle of a pipelined timing system are determined primarily by noise constraints.

TABLE 1-2 **An Example Noise Budget**

Noise Source	Type	Amount (%)	Amplitude (mV)
Gross margin			125
Receiver offset	Fixed		±10
Receiver sensitivity	Fixed		±10
Unrejected power supply noise	Fixed		±5
Transmitter offset	Proportional	5	±13
Cross talk	Proportional	10	±25
Intersymbol interference	Proportional	10	±25
Total noise			±88
Net margin			37

Operating digitally, of course, we can recover the correct symbol and timing in the presence of noise as long as the noise does not exceed a fixed threshold, often called the *noise margin*. Careful design is required, however, to make the probability of exceeding this margin appropriately small.

Most noise in digital systems is created by the system itself and thus should properly be called *interference*.[4] The most significant noise sources are power supply noise, cross talk, and intersymbol interference. Power supply noise, voltage fluctuations of the power supply caused by the AC current demand, couples into signals and modulates the delay of timing elements. Cross talk occurs when a symbol on one signal line interferes with the symbol carried on another signal line and is caused by capacitive and inductive coupling between signal lines and their returns. Intersymbol interference occurs when symbols placed on a signal line interfere with later symbols on the same line because of parasitic tank circuits, slow circuit nodes, and reflections from imperfect terminations.

As engineers, we manage noise by using a *budget* that allocates our gross noise margin to expected sources of noise. For example, Table 1-2 shows the noise budget for the ±2.5-mA current-mode channels of Figure 1-1. These signals have a 250-mV swing and hence a 125-mV *gross noise margin*. This is the amount of noise that can be tolerated before a symbol will be incorrectly detected. We perform a worst-case analysis to see if this margin is adequate to account for expected noise sources. The table shows that if the six noise sources have maximum amplitude and sum in the same direction, the worst possible case, the maximum noise is 88 mV. This leaves a *net noise margin* of 37 mV to provide safety and allow for unexpected noise sources.

In designing a system to deal with noise, it is useful to separate noise sources into those that are *fixed*, independent of signal magnitude, and those that are *proportional* and scale with signal magnitude. Three of the noise sources in

[4] In this text we reserve the term *interference* to refer to interference between digital signals (e.g., crosstalk and intersymbol interference). We refer to interference from other sources simply as *noise*.

Table 1-2 are fixed, whereas the remaining three are proportional, totaling to 25% of the signal swing (which is 50% of the gross margin). The fixed sources can be dealt with by increasing signal swing. This would have no effect, however, on the proportional sources, for they would increase at the same rate. They must be reduced or rejected. They cannot be overpowered.

An efficient system design cancels noise where possible. For example, the third noise source in Table 1-2 is *unrejected* power supply noise. The system has ± 150 mV of supply noise. All but 3% of this noise is rejected as common-mode between the signal and the return. It would be far more expensive to attempt to overpower this supply noise.

1.2.6 A Systems View of Circuits

When designing signaling and timing conventions and budgeting for noise, the digital systems engineer is constrained by what can be practically built. Any real receiver will have a finite offset and sensitivity, any real flip-flop will have finite setup and hold times, and any real delay line will introduce some amount of jitter. These circuit constraints play a major role in the design of a system such as that of Figure 1-1. It is important to appreciate how these implementation constraints arise and to determine which are controllable and which are not. For this reason, we will take an in-depth look at circuits in this book from a systems perspective. Rather than study amplifiers or flip-flops in isolation, we will look at how their design affects system properties, such as required signal amplitude and cycle time.

1.3 TECHNOLOGY TRENDS AND DIGITAL SYSTEMS ENGINEERING

The phenomenal growth in the performance and range of applications of digital systems is fueled by an exponential rate of improvement in the underlying semiconductor technology. Every three years, the number of devices that can be fabricated on a chip, and more importantly the number of wiring *grids* on a chip, quadruples.

As chip density improves, however, the problems of digital systems engineering get harder. For example, smaller devices operate at lower voltages to maintain constant fields, and at the same time the increased number of devices draws larger AC currents, making the power distribution problem doubly hard. Similarly, timing and signaling are also becoming more difficult as technology advances.

1.3.1 Moore's Law

Gordon Moore [Moore79] observed that the number of devices that could be economically fabricated on a single chip was increasing exponentially at a rate of about 50% per year and hence quadrupling every 3.5 years. At the same time,

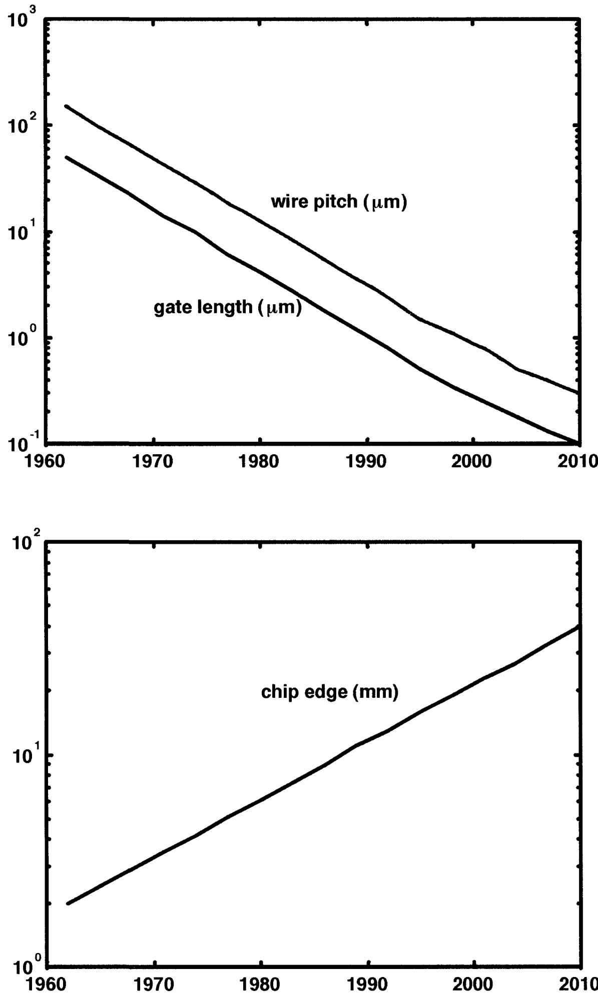


FIGURE 1-2 Scaling of Integrated Circuit Dimensions

the delay of a simple gate has been decreasing by 13% per year, halving every 5 years. The semiconductor industry has maintained this breakneck trend for the past 27 years, and there is every reason to expect that it will continue for at least the next decade.

Figure 1-2 is a plot of the exponential scaling of integrated circuit feature size and linear dimension as a function of time.[5] The upper graph shows how gate length and wire pitch decrease at a rate of 13% per year, halving every 5 years. Gate lengths have scaled from 50 μm in the 1960s to 0.35 μm today and are projected to reach 0.1 μm by 2010. At the same time that the sizes of features

[5] The scaling figures in this section make use of data from several sources. Historical data are based on surveys, including that by Hennessy and Jouppi [HennJoup91]. Future trends are based largely on projections by the Semiconductor Industry Association [SIA94], and some adjustments are based on the authors' expectations.

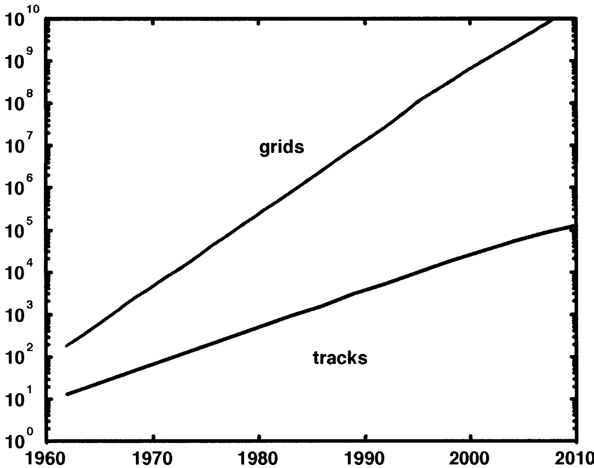


FIGURE 1-3 Scaling of Available Wiring Tracks and Grids

have been decreasing, chip size has been increasing, as illustrated by the lower panel. The length of a chip edge, the square root of chip area, is increasing by 6% per year, doubling after about a decade. Chips have scaled from 2 × 2 mm in the early 1960s to 18 × 18 mm today and are projected to reach 40 mm on a side by 2010.

The trends plotted in Figure 1-2 combine to give the increases in wiring tracks and grids shown in Figure 1-3. The combination of decreased wire pitch with increasing chip size gives a 22% annual increase in the number of wiring tracks that can be placed side-by-side across the width of a chip. Early small-scale integrated circuits (SSI) in the 1960s had only a few tens of wiring tracks. Today's chips have about 15,000 wiring tracks, and by 2010 chips will have over 100,000 wiring tracks. The number of grid squares defined by the wiring tracks in the x and y dimensions increases as the square of the number of tracks, or 50% per year, quadrupling every 3.5 years and increasing by an order of magnitude every 6 years.

The number of *grids* on the chip is a good measure of the amount of functionality that can be implemented on the chip. A bit of a given type of memory or a certain arithmetic function requires a number of grids of a chip area that remains roughly constant over time.[6] Early chips in the 1960s had only a few hundred grids. Modern chips have 10^8 grids, and by 2010 chips will have 10^{10} grids. This increase of six orders of magnitude over the last 35 years in the functionality that can be placed on a chip has led to major challenges and opportunities in deciding how to organize a system efficiently to use all this area.

The speed of on-chip functions is also increasing exponentially with time. To first order, the delay of a simple gate decreases linearly with gate length. As

[6] Of course major advances in memory or arithmetic circuits, such as the development of the one-transistor DRAM in the middle 1970s or recoded multipliers in the 1950s, can greatly reduce the number of grids required for a particular function. However, advances of this magnitude are relatively rare today.

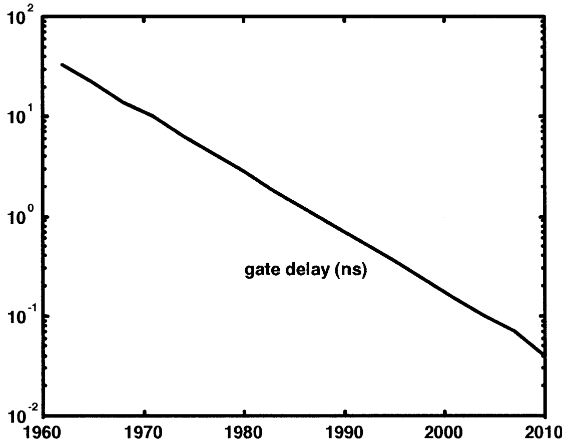


FIGURE 1-4 Scaling of Gate Delay

illustrated in Figure 1-4, gate delay decreases by 13% per year, halving every 5 years. The delay of a simple 2-input NAND gate with a fan-out of four has gone from tens of nanoseconds in the 1960s to a few tenths of a nanosecond today and is projected to be a few tens of picoseconds by 2010. Unfortunately, as described in the following paragraphs, the blinding speed of these gates is likely to be masked by wire delays that increase over time.

The overall capability of a chip depends on the number of functions it contains and the speed of these functions. Combining this 13% per year increase in speed with the 50% per year increase in grids gives an overall increase in capability of 70% per year. This measure of capability increases by an order of magnitude roughly every 4 years. Figure 1-5 plots capability, measured as grids divided by gate delay, as a function of time. This measure of capability has gone from 5 in the early 1960s to about a billion today and is projected to reach nearly a trillion by 2010.

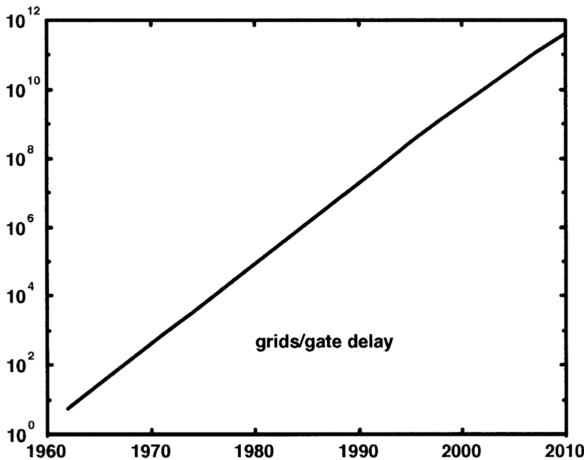


FIGURE 1-5 Scaling of Capability

TABLE 1-3 **Summary of Integrated Circuit Scaling**

Parameter	1998 Value	Sp[a]	Units
Gate length, L_d	0.35	0.87	μm
Wire pitch, χ	1.10	0.87	μm
Chip edge	19	1.06	mm
Tracks	1.8×10^4	1.22	χ
Grids	3.2×10^8	1.49	χ^2
Gate delay	0.23	0.87	ns
Capability	1.4×10^9	1.71	χ^2/ns
Pins	549	1.11	

[a]Annual scale factor.

Such a rapid increase in capability makes time a key variable in measuring performance. For example, if a new technology offers a factor of two increase in capability but is delayed for 2 years, the advantage is more than negated. A quantitative increase in capability of this magnitude also implies a qualitative difference in machine organization. The organization that worked best at one level of capability is unlikely to be optimum for a machine built from chips that are 100 times more capable.

Table 1-3 summarizes the scaling of the parameters described in this section. For each parameter, P, the table gives its 1998 value, $P(1998)$, and its annual scale factor, S_P. The value of the parameter for a given year, x, can be estimated by

$$P(x) = P(1998)S_P^{(x-1998)} \tag{1-1}$$

1.3.2 Scaling of Chip Parameters

As linear dimensions scale by a factor, x, chip parameters scale, as illustrated in Table 1-4, so that switching energy decreases as the cube of x while current draw increases at 30% per year. The first five rows show the scaling described in Section 1.3.1. To maintain constant field strength, as linear dimensions scale by x, voltage must also be scaled by x. The area of a minimum-sized device scales quadratically with x, whereas the dielectric thickness scales linearly, giving a linear decrease in device capacitance. Similarly, the resistance of a minimum-sized wire scales as $1/x$ because the cross section scales as x^2 and the length scales as x. Hence, resistance of a device-sized wire increases by 15% per year. As to first approximation, charge carriers travel at constant speed under a constant field, device speed is proportional to gate length and thus scales linearly with x.

From these basic parameters, we can derive how the energetics of the technology scale. The switching energy, $E_{sw} = C_{dev}V^2$ scales with the cube of x, decreasing 34% per year! However, because the frequency of switching events has increased and the voltage has decreased, the current scales only linearly with x. More importantly, the current per unit area scales as $1/x$, increasing by

TABLE 1-4 Scaling of Chip Parameters

Parameter	Symbol	Scaling		Per Year
Wire pitch	x			0.87
Chip edge	y			1.06
Area	A		y^2	1.13
Tracks/chip	T		y/x	1.22
Grids/chip	G	T^2	$(y/x)^2$	1.49
Voltage	V		x	0.87
Wire capacitance	C_{dev}		x	0.87
Wire resistance	R_{dev}		$1/x$	1.15
Switching time	τ		x	0.87
Switching energy	E_{sw}	$C_{dev}V^2$	x^3	0.66
Current	I_{dev}	$C_{dev}V/\tau$	x	0.87
	I_{area}	I_{dev}/x^2	$1/x$	1.15
	I_{chip}	AI_{area}	y^2/x	1.30

15% per year and, because area increases as well, the current per chip is increasing even faster at 30% per year. Although the energy of each switching event is decreasing at a phenomenal rate, the current draw per chip is increasing nearly as fast.

1.3.3 Scaling of Wires

As technology scales, wires become increasingly important compared with devices in terms of power, delay, and density. The reasons for this are illustrated in Table 1-5, which shows how wire properties scale as linear dimensions decrease and chip area increases. The table contains three pairs of columns. They show the scaling formula and the change per year for a device-sized wire, a 1-μm wire, and a wire that crosses the entire chip. The first two rows show how the capacitance and resistance of these wires scale over time. Although the capacitance of a device-length wire decreases over time, the capacitance for a fixed-length wire

TABLE 1-5 Scaling of Wire Properties

Parameter	Device		1-μm		Chip	
C	x	0.87	1	1.00	y	1.06
R	$1/x$	1.15	$1/x^2$	1.32	y/x^2	1.40
I	x	0.87	1	1.00	y	1.06
IR	1	1.00	$1/x^2$	1.32	$(y/x)^2$	1.49
IR/V	$1/x$	1.15	$1/x^3$	1.51	y^2/x^3	1.71
RC	1	1.00	$1/x^2$	1.32	$(y/x)^2$	1.49
RC/τ	$1/x$	1.15	$1/x^3$	1.51	y^2/x^3	1.71

remains constant, and the capacitance for a *global* wire that crosses the width of the chip increases. The resistance of all three wires increases but at a significantly more rapid rate for the longer wires.

1.3.3.1 Scaling of Power Distribution

Rows three through five of Table 1-5 show how distributing power becomes increasingly difficult as technology scales. These three rows consider the current supplied by a single supply wire. The current draw of a device-wide region is a constant times the length of the region, decreasing for a device-long region and increasing with chip length for a chip-long region. This increasing current must be carried over wires with resistance that is increasing rapidly over time. Row four shows that the *IR* drop of a constant-length wire is increasing at 32% per year, and the drop across a chip-long wire is increasing at 49% per year. Worse yet, as voltages decrease, the tolerable *IR* drop decreases proportionally. Thus, the relevant parameter is the *IR* drop as a fraction of voltage, IR/V. This scales as $1/x$, even for a device-sized wire, and as y^2/x^3 for a chip-sized wire, increasing at a whopping 71% per year! In short, if a power distribution strategy is left unchanged while all the linear dimensions and voltages decrease by 13%, the *IR* power supply noise, as a fraction of voltage, nearly doubles.

Careful engineering is required to combat this increase in supply noise. Brute-force solutions like more and thicker metal layers for power distribution or area bonding play an important role. These methods are costly, however, and economical chips complement them with better power distribution techniques involving on-chip regulation and bypass capacitors as well as low-power design methods. These issues are discussed in depth in Chapter 5.

1.3.3.2 Scaling of On-Chip Communication

The increasing difficulty of on-chip communication is shown in rows six and seven of Table 1-5. The RC delay of a constant-length wire increases by 32% per year, and the delay of a cross-chip wire increases by 49% per year. The relevant quantity, however, is how the delay of the wire compares with the decreasing device delay. This parameter, RC/τ is increasing by 51% and 71% for the constant length and cross-chip wire, respectively. To complicate matters, this delay is diffusive, and thus the rise-time, and hence the uncertainty of the delay, increase at the same rate.

In the near future this scaling will cause wire delay to dominate gate delay entirely in most systems. In 1998, the delay of a loaded gate, 200 ps, is about the same as the delay of a 5-mm on-chip wire. In 2008, the gate delay decreases to 50 ps, whereas the wire delay increases to 12 ns. The 1:1 ratio becomes 250:1. In the future, systems must be organized to keep wires short, and signaling methods that make the most of lossy RC wires must be employed to reap the advantages of faster gates.

This dramatic increase in the delay of a global wire relative to gate delay, almost doubling every year, affects the signaling, timing, and architecture of digital systems. Signaling conventions tailored to RC lines involving repeaters and pulsed low-voltage swing drivers are needed to get the maximum performance

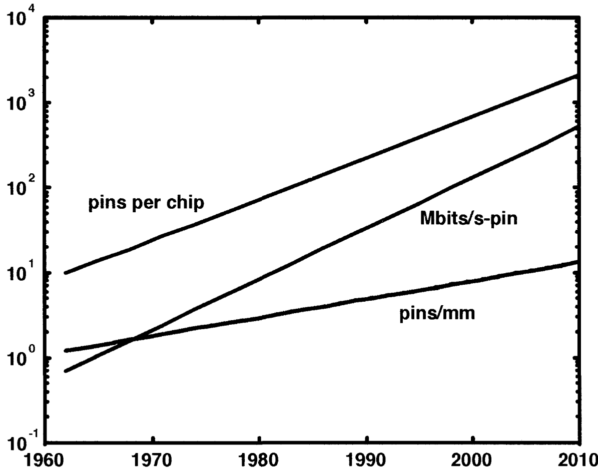


FIGURE 1-6 Scaling of Pins per Chip

from these lossy wires. The lossy nature of the wires makes it extremely difficult to distribute a global clock with low skew. A brute-force solution to this problem involves devoting a large quantity of metal to building a low-impedance clock grid. More clever solutions relax the need for global synchrony by dividing the chip into separate clock domains or employing asynchronous design techniques. Finally, architectures that exploit locality, thus reducing the need for communication over global wires, will perform much better than those that incur the long latencies inherent in global communication over lossy wires.

1.3.3.3 Scaling of Off-Chip Communications

The number of pins that can be economically fabricated on a chip increases as the perimeter of the chip times the linear density of pins. This is true even for area bonding because the pins must still be routed out from under the perimeter of the chip in the next-level package. As illustrated in Figure 1-6, pin count increases by 12% per year. Early chips had only about ten pins, contemporary chips have hundreds of pins, and chips with several thousand pins should be economical by 2010. The growth in pin count is due to the 6% annual increase in chip perimeter combined with a 6% annual increase in pin density. Pin density depends on the track spacing of the second-level packaging technology, which typically scales at about half the rate as on-chip wire spacing.

To complicate matters, off-chip signaling rates have historically trailed on-chip clock rates. The bandwidth per pin for an average chip is also shown in Figure 1-6. Signaling rates of about 1 Mb/s-pin were typical in the 1960s, whereas modern chips approaching 100 Mb/s-pin and rates of 500 Mb/s-pin should be widespread by 2010. A chip using aggressive signaling technology can do about an order of magnitude better than this curve.

The number of available pins is being rapidly outpaced by the amount of on-chip functionality. The bandwidth demand of a partition of an architecture is often captured in a rule-of-thumb known as *Rent's rule*. This rule states that the

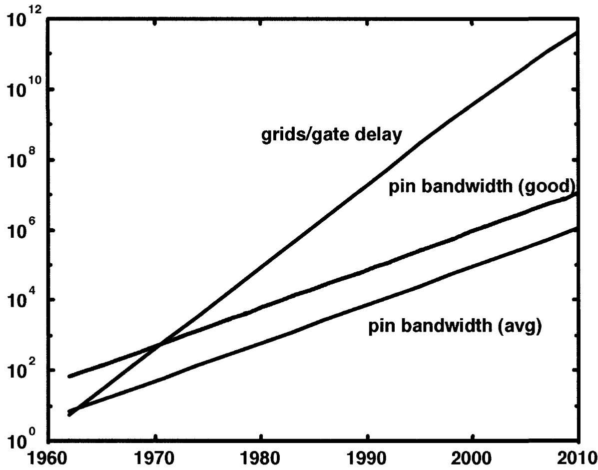


FIGURE 1-7 Scaling of the Gap Between Pin Bandwidth and Capability

bandwidth,[7] $B(b/s)$, from a module with capability C (gates $\times$ Hz) grows as C^{α}

$$B = K_R C^{\alpha} \tag{1-2}$$

For machines designed without much concern for locality, the exponent α is typically in the range 0.5 to 0.75, with 0.7 being typical.

It was possible to build machines with Rent exponents of 0.7 in the 1960s when this rule was coined; however, the relative scaling of capability to off-chip bandwidth will not sustain such a large exponent. As illustrated in Figure 1-7, capability increases annually by 71%, whereas the available pin bandwidth increases annually by only 28%. The largest exponent that can be sustained under this scaling is 0.46 (log 1.28/ log 1.71). The gap between the required bandwidth (with $\alpha = 0.7$) and the available bandwidth (with $\alpha = 0.46$) is large and growing.

The widening gap between available bandwidth and demand poses many challenges for the digital systems designer. For many systems, off-chip bandwidth is the single largest performance bottleneck. As with on-chip communication, this problem must be addressed through better signaling and better architecture. With good signaling techniques (Chapter 7), off-chip signaling can operate at speeds higher, rather than lower, than the on-chip clock and increase at the same rate.

A switch to faster signaling gives a one-time increase in off-chip bandwidth that, however large, is rapidly canceled by the 33% annual growth in the capability-to-bandwidth ratio. For example, the order of magnitude increase between the average and good bandwidth curves in Figure 1-7 only postpones a bandwidth problem for 8 years. In the long run, better architecture is needed to keep most communication on-chip, thus reducing demand.

[7] The original Rent's Rule [LandRuss71] relates the number of pins on a module to the number of gates. When the clock cycle and the signaling rate are identical, this rule is the same as ours. We prefer to relate bandwidth to capacity, however, because it covers the more general (and quite common) case in which the signaling rate and clock cycle differ.

1.3.4 High Levels of Integration Permit New Approaches

The rapid improvement of VLSI technology gives us the tools to tackle many of the technical challenges that the evolving technology has created. Modern VLSI chips have several hundred million grids of chip area containing millions of transistors. Using a small fraction of this area, we can build circuits that solve many of the system-level electrical problems. By applying logic, we can solve these problems without the need for brute-force application of power, large pin-counts, or expensive package design. Some examples include the following:

1. Using custom VLSI chips at both ends of a signal allows us to tailor a signaling convention that isolates noise sources to enable signaling at rates significantly higher than in systems employing conventional signaling.
2. By encoding a stream of symbols before transmission, we can solve a multitude of problems. Band-limited coding reduces the problems of frequency-dependent attenuation and aids clock recovery. DC-balanced coding greatly reduces signal-return cross talk (sometimes called ground bounce).
3. Using pipelined timing conventions can increase both throughput and reliablity over that achieved with conventional synchronous timing.
4. Rather than carefully matching the delay of signal paths and still having a considerable amount of residual skew, using closed-loop per-line clock recovery can completely eliminate skew between a clock and a data signal.
5. Regulating the power at the point-of-use to address the problems of on-chip power distribution locally.

This book explores these examples, and others, in detail.

1.3.5 Digital Systems Problems and Solutions Continue to Change

The rapid advance of VLSI technology and the pace of change in applications make digital systems engineering a very dynamic field. Solutions that work today may neither be appropriate nor adequate in the future. Each generation of technology and each new application area demand new solutions.

Often, compatibility with existing parts, processes, and infrastructure along with a reluctance to try new techniques acts as a retardant, slowing the rate of change. Many engineers prefer tuning to innovation and operate by making a few small changes to an existing solution. Time is usually a factor as well. With today's short development cycles there is little time for experimentation or risk taking. In the end, the marketplace sorts out the prudent from the reactionary and generally rewards those who take calculated risks.

Decoupling technology development from product development reduces the risks associated with new techniques. Developing a new signaling, timing, or power distribution technology independent of, and in advance of, a product development limits the impact of technology failures on product development and creates an atmosphere that is more conducive to innovation. Unfortunately, few companies today have the foresight to invest in such advanced development.

1.4 ORGANIZATION OF THIS BOOK

This book is organized into three main sections. In Chapters 2 to 4, we discuss the elements of digital systems design: wires, transistors, and simple circuits. These chapters lay the groundwork for the remainder of the book by developing engineering models of the important digital systems building blocks. Chapter 2 begins by discussing how digital systems are packaged into chips, modules, circuit boards, and chassis and how these modules are interconnected with various forms of wires. In Chapter 3, we look at the electrical properties of these packaging technologies and develop a set of engineering models for wires that are used throughout the rest of the book. Finally, in Chapter 4 we investigate the properties of the metal-oxide semiconductor (MOS) transistor and the composition of transistors into simple digital circuits.

The next part of the book, Chapters 5 to 10, investigates the key issues of digital systems engineering at the system level. We begin with the problem of power distribution in Chapter 5. Noise is discussed in Chapter 6. Signaling methods are covered in Chapters 7 and 8. Timing conventions are discussed in Chapter 9, and synchronization issues are explored in Chapter 10.

The final part of the book, Chapters 11 and 12 describe circuits for signaling and timing, respectively. The discussion of signaling circuits in Chapter 11 complements the system-level treatment of signaling in Chapters 7 and 8. Similarly, the treatment of timing circuits in Chapter 12 complements the system-level discussion of timing in Chapters 9 and 10.

1.5 BIBLIOGRAPHIC NOTES

Two excellent texts on topics relating to Digital Systems Engineering are Bakoglu [Bakoglu90] and Johnson and Graham [JohnGrah93]. Bakoglu gives a strong theoretical treatment of VLSI circuits and interconnects. Johnson and Graham give a more practical treatment of topics relating to system-level interconnects and timing. The example discussed in Section 1.2 is based loosely on modern multicomputers such as the Cray T3E [Scott96] and the SGI Origin 2000 [LaudLeno97]. The scaling of integrated circuit technology is described in [Moore79], [HennJoup91], and [SIA94]. Rent's rule, relating module pinout-to-gate complexity is discussed in [LandRuss71].

1.6 EXERCISES

1-1 Pin Budgets: The router chip in Figure 1-1 is over the pin budget, as indicated in Table 1-1. The following suggestions have been proposed to repartition the system to bring the individual chips within pin budget:

1. Make the router channel pins single-ended on the router chip and use single-ended-to-differential converter chips to generate differential versions of these signals.

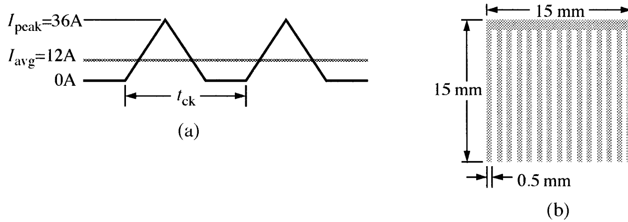


FIGURE 1-8 Current Waveform and On-Chip Power Distribution

2. Partition the router across two chips with each handling three of the incoming channels. The controller bandwidth is split evenly across these two subrouters, and a 2-GB/s channel is used for communication between the two half routers.

Evaluate these suggestions in terms of total numbers of chips and pins required. Which would you pick?

1-2 **Signaling:** The system shown in Figure 1-1 uses four types of signaling: SECM (single-ended current mode), DCM (differential current mode), CMOS (full-swing), and HSTL. Compare the first three of these in terms of power per bit. Assume that all signal lines are 50-Ω and parallel-terminated.

1-3 **Off-Chip Power Distribution:** Suppose the router chip from Figure 1-1 draws an average current of 12 A from the 2.5-V supply over an inductance of 20 nH. The supply network must be designed so that when the router idles and draws zero current, the supply voltage of the router must remain below 2.75 V. Draw a simple circuit that illustrates this situation. How large a capacitor must be connected across the router to meet this requirement? How low must the lead inductance of this capacitor be?

1-4 **Noise Budgets:** If the signal swing of the SECM channels of Figure 1-1 is doubled from 250 to 500 mV, what happens to the noise budget of Table 1-2? What is the new net noise margin? What is the ratio of gross noise margin to noise before and after this change?

1-5 **Arithmetic Scaling:** A 32-bit integer multiplier uses a 32 × 16 array of full adders.[8] Each full adder, along with its associated multiplexer, is 16χ on a side. How many chips, or what fraction of a chip, does it take to make a 32 × 32 integer multiplier in 1965, 1985, and 2005? Each full adder has a delay of one *gate delay* (as shown in Figure 1-4), and the worst-case path through the multiplier goes through nine full adders. How many multiplications per second can be realized per chip in 1965, 1985, and 2005? At what rate does this quantity scale?

1-6 **Scaling of On-Chip Power Distribution:** The router chip of Figure 1-1 draws 12 A *average* current with the waveform shown in Figure 1-8(a). The peak instantaneous current here is 36 A. This power is distributed over a comb of thin-film aluminum conductors, as shown in Figure 1-8(b). If these conductors have a resistivity of 0.04 Ω/square, and the current draw is distributed uniformly across the chip,

[8] Booth recoding allows the multiplication to be performed with a 32 × 16 array rather than a 32 × 32 array.

what is the worst-case drop across one *tine* of the comb? Express this quantity as an absolute voltage and as a fraction of the power supply. Assume the number you just calculated applies to a chip in 1995. Applying the scaling relationships for I, R, and V from Table 1-5, what do you expect the IR drop to be (in volts and % supply) in the years 2000, 2005, and 2010?

1-7 **Scaling of Wire Delay:** Suppose a system has a clock cycle of ten gate delays. What clock frequency does this imply in the years 1990, 2000, and 2010? In these same years, how many clocks, or what fraction of a clock, does it take to drive a wire from one corner of the chip to the opposite corner? Assume this wire runs only horizontally and vertically so that its length is twice the linear dimension of the chip. For now, assume that the delay of a wire is its RC time constant. We will see how to do better in Chapter 8.

1-8 **Chip Architecture:** As described above in Exercise 1-5, a full-adder takes an area of $256\chi^2$ (16χ on a side). Thus, a 32-bit adder is $8\text{K}\chi^2$, and a 32-bit recoded multiplier is $128\text{K}\chi^2$. How many multiplications and additions per second can be performed on a chip in 2005 and 2010? Discuss some possible uses for chips with this level of performance.

1-9 **Component Specification:** One task of a digital systems engineer is keeping up-to-date on the specifications of available *standard* components – memory components in particular. Using a Web browser (the new way) or a databook library (the old way), look up the specifications of two competing contemporary memory chips. For example you could investigate high-bandwidth DRAM components (RAMBUS versus synchronous DRAM) or static memory components (HSTL versus TTL SRAMs). Describe the two components you have chosen and evaluate their advantages and disadvantages. Which would you choose? For which application?

1-10 **Literature Search:** It is also important for a digital systems engineer to keep up-to-date with the latest custom chips and the techniques employed in their design. These chips are reported in conferences such as the International Solid-State Circuits Conference (ISSCC), the Microprocessor Forum, and Hot Chips. Industry newsletters such as *Microprocessor Report* are also a good source of information on this topic. Look up a contemporary microprocessor chip in a recent copy (within the last year) of one of these conference proceedings or newsletters. Discuss the digital-systems techniques employed. How is clock distribution performed? How are on-chip power supplies handled? What types of signaling conventions are used?

2 PACKAGING OF DIGITAL SYSTEMS

2.1 A Typical Digital System 27
2.2 Digital Integrated Circuits – On-Chip Wiring 29
2.3 Integrated Circuit Packages 30
- **2.3.1** Wire Bonds and Solder Balls 30
- **2.3.2** Package Types 32
- **2.3.3** Package Manufacturing Processes 35
- **2.3.4** Multichip Modules 36
- **2.3.5** A Typical Package Model 36

2.4 Printed Circuit Boards 39
- **2.4.1** PC Board Construction 40
- **2.4.2** Electrical Properties 41
- **2.4.3** Manufacturing Process 42
- **2.4.4** Vias 42
- **2.4.5** Dimensional Constraints 44
- **2.4.6** Mounting Components: Surface-Mount and Through-Hole 44
- **2.4.7** Sockets 46

2.5 Chassis and Cabinets 47
2.6 Backplanes and Mother Boards 48
- **2.6.1** Daughter Cards 48
- **2.6.2** Backplanes 48

2.7 Wire and Cable 49
- **2.7.1** Wires 49
- **2.7.2** Signaling Cables 51
- **2.7.3** Bus Bars 53

2.8 Connectors 54
- **2.8.1** PC Board Connectors 55
- **2.8.2** Interposers 56
- **2.8.3** Elastomeric Connectors 57
- **2.8.4** Power Connectors 58
- **2.8.5** Wire and Cable Connectors 58

2.9 Optical Communication 62
- **2.9.1** Optical Transmitters 63

2.9.2 Optical Fiber 64
2.9.3 Optical Receivers 67
2.9.4 Multiplexing 68
2.9.5 Optical Amplifiers 70
2.9.6 Free-Space Optical Interconnect 70
2.10 Radio Communication 71
2.10.1 A Typical Digital Radio 71
2.10.2 The Power Equation 72
2.10.3 Modulation 74
2.10.4 Multipath 76
2.11 Bibliographic Notes 77
2.12 Exercises 77

Digital systems are packaged in a hierarchy of chips, carriers, circuit boards, chassis, and cabinets. Signals are transported on batch-fabricated thin-film and thick-film wires on chips, carriers, and boards, and over cables between boards and chassis. The physical characteristics of the wires at each level determine their electrical properties, their cost, and the maximum signal density that can be supported.

At the lowest level, electronic components (mostly field-effect transistors) are fabricated on the surface of silicon integrated circuit *chips* and interconnected by layers of thin-film metal wiring. One or more of these chips are mounted in a *package* that connects their pads to package pins. Packages are mounted on a circuit board that provides low-impedance power and signal return planes and interconnects the package pins with thick-film metal layers. Circuit boards in turn may be connected to one another by other circuit boards (backplanes or mother boards) or by cables. To share mechanical support, several circuit boards are usually packaged together in a chassis, and several chassis may be housed in a cabinet or rack.

At each level of the hierarchy, from chips to cables, a set of design rules governs the physical size and spacing of wires and connections. The geometry that arises from these design rules constrains available wire density and determines the electrical properties of the resulting wires. In this chapter we discuss available packaging technology at each level of the hierarchy and survey current design rules. In Chapter 3, we study the electrical properties and cost models of the wires at each level of the hierarchy.

Wires are not the only means available for moving digital symbols from one place to another in a system. For completeness, we will briefly review the two principal *wireless* signaling technologies: optical and radio. Although a complete discussion of these technologies is beyond the scope of this book, we present sufficient information for the digital systems engineer to understand the capabilities and limitations of these techniques and where they can be best applied.

The remainder of this section discusses in more detail each level of the packaging hierarchy. We begin in Section 2.1 by examining a complete digital system to see how the different packaging levels fit together. We then work from the bottom up, examining each packaging level, from chips to cables, in detail in Sections 2.2

through 2.8. Finally, we give a brief overview of optical signaling in Section 2.9 and radio communications in Section 2.10.

2.1 A TYPICAL DIGITAL SYSTEM

Figure 2-1 is a photograph of a typical digital system, a high-end personal computer. The system's main components, including its processor and memory interfaces, are mounted on a larger mother board. Specialized I/O hardware plugs into an I/O bus card, which itself plugs into the mother board. Memory is mounted on SIMs (single in-line modules). Power is distributed from a central power supply in the lower part of the cabinet to the boards, discs, and other components over wiring harnesses made up of individual conductors. Data are distributed from boards to discs (e.g., over ribbon cables). Like all but the highest performance computers, this machine is air-cooled; a single fan pulls air in through the bottom of the enclosure past the power supply, discs, and circuit cards. The entire system is housed in a metal enclosure welded from individual pieces of stamped steel and covered with plastic panels that provide noise reduction and improved appearance.

Table 2-1 summarizes the properties of the packaging levels in a digital system built using 1997 technology. For each level the table shows the size of the level, the number of wiring layers typically used (for both signals and power), various wire properties, and the number of pins to the next level. The wire properties include pitch (the spacing between two adjacent wires on the same layer), wire cross section, resistance per unit length, and the type of transmission line used to

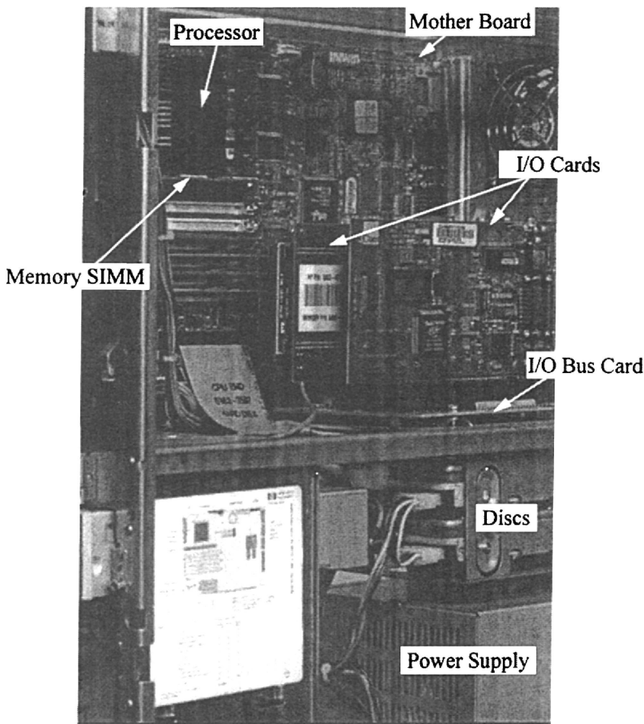


FIGURE 2-1 Packaging of a Typical Digital System

TABLE 2-1 **Properties of Packaging Levels**

Level	Size	Layers	Pitch (μm)	Tracks/ Layer	Wire Cross Section (μm)	Resistance (Ω/mm)	Wire Model	C (pF/mm)	Pins
Chip	15×15 mm	4	1.2	12,500	0.6×0.6	150	*C/RC*	0.2	500
Package	5×5 cm	4	150	333	75×20	1 Ω (total)	*LC*	1 pF (total)	500
Circuit board	40×30 cm	8	625	2000	125×20	0.05	*LC/LRC*	0.1	2,000
Backplane	40×30 cm	8	625	2000	125×20	0.05	*LC/LRC*	0.1	1,000
Chassis	$40 \times 40 \times 40$ cm	N/A	N/A	N/A	N/A		*LRC*		1,000
Cabinet	$1.5 \times .6 \times .6$ m	N/A	N/A	N/A	N/A		*LRC*		3,000

model the wire. Perhaps the single most relevant number is the number of tracks per layer (the linear dimension divided by the wire pitch) because this is a good measure of the wiring capacity of a given level.

The numbers in this table are for a typical system. Examples exist of systems with larger chips and boards, finer wiring pitches, more layers, and more pins between levels. However, exceeding these typical numbers usually carries a heavy price premium.

As technology advances, the numbers in this table will improve. However, they do not all improve at the same rate. Integrated circuit wire density is advancing at a rate of 22% per year, doubling about every 3 years. The number of tracks per layer on a circuit board, on the other hand, increases by only about 7% per year, doubling only after 10 years. This nonuniform scaling changes the nature of the digital system design problem over time, making solutions that were optimal 10 years ago inappropriate today.

2.2 DIGITAL INTEGRATED CIRCUITS – ON-CHIP WIRING

Integrated circuits are at the core of a modern digital system. In fact, today there are many systems that fit entirely on a single integrated circuit. A single 15-mm square chip today (1997) can hold up to 64 Mbits of memory, several million gates, or some combination of the two. A simple 32-bit CPU (without memory) can be realized in an area just 1 mm on a side. Over 200 of these simple CPUs would fit on a 15-mm square chip!

A typical 0.35-μm CMOS chip has four wiring layers on which 0.6-μm square aluminum wires are routed.[1] Aluminum wires (resistivity, $\rho = 2.65 \times 10^{-8}$ Ω-m) of this size have a resistance of 0.04 Ω/square, which implies 74 Ω/mm for a minimum-width wire at 25°C. The top layer may be made thicker, with coarser design rules, to facilitate power distribution at the expense of signal density.

The signal layers on such a chip are spaced by 0.6-μm dielectric layers of SiO_2 (dielectric constant, $\varepsilon_r = 3.9$) that are sometimes doped with other materials. Silicon nitride (Si_3N_4, $\varepsilon_r = 7.5$) is used for the top-most dielectric layer. A wire in a dense region of a chip is surrounded by minimum-pitch parallel wires on its layer and minimum-pitch perpendicular wires on adjacent layers. Such a wire has a significant capacitance, not to ground or the substrate, but to these other wires. Only the bottom wiring layer has a significant capacitance to the substrate. The total capacitance for a wire in a dense region is 160 fF/mm. As described in Section 6.3.1, the presence of this capacitance between signal wires leads to capacitive crosstalk between signals.

Because of their high resistivity, short length, and tight pitch, on-chip wires almost always have an inductance low enough to be safely ignored. Thus, in our analysis we will model short on-chip wires as lumped capacitive loads and longer on-chip wires as lossy RC transmission lines. Any wire whose resistance is small compared with the circuit driving it will be considered short. Typically, wires

[1] Some chips use refractory metals such as tungsten (W) for on-chip wiring to allow high-temperature processing after the wiring layer is deposited.

under 1 mm are short, but resistance must be considered for all longer wires. Because of their resistivity, driving long on-chip wires at high speed is among the most difficult signaling challenges we will encounter.

Wire density is a limiting factor in most integrated circuits. Except for very regular arrays (such as memories), on-chip function units tend to be *wire-limited*. That is, their area is determined not by the number of gates or transistors but rather by the wiring required to connect them.

Perhaps the biggest limitation of a modern digital integrated circuit is the large reduction in signal count between on-chip wires and package pins. With over 10^4 wiring tracks in each dimension on each of four layers, it is not unusual to have several million on-chip signals. However, packaging constraints require that only 500 or fewer signals can leave the chip. For inexpensive plastic packages, fewer than 200 signals can leave the chip. Because of this limitation, chips are often *pad-limited*, and the area of the chip is determined by the number of pads required to bring signals off-chip and not by the density of on-chip circuits. This is particularly common with peripheral-bonded chips for which the required chip area increases as the square of the number of pads.

2.3 INTEGRATED CIRCUIT PACKAGES

Most integrated circuits are bonded to small ceramic or plastic packages. Although it is possible to attach chips directly to boards, a method used extensively in low-cost consumer electronics, placing the chips in packages enables independent testing of packaged parts, simplifies reworking of boards, and eases the requirements on board line pitch by spreading out the pins. In some cases, several chips are packaged together on a ceramic or laminated module. Such *multichip modules* allow high-density interconnection between the chips packaged together.

Electrically, the signal leads of first-level packages have significant mutual and self-inductance. If the leads are fabricated in pairs or at uniform height over a return plane, the lead is best modeled as a transmission line. With proper control of conductor size and spacing, the impedance of this line can be matched to other transmission media (e.g., a 50-Ω stripguide on a PC board). If spacing to a return is large or irregular, the lead is modeled as a lumped inductor. Depending on how the chip is bonded and packaged, it is not unusual for the conductor from the chip pad to the PC board to have as much as 10 nH of self-inductance and 5 nH of mutual inductance to adjacent conductors.

2.3.1 Wire Bonds and Solder Balls

Figure 2-2 illustrates the most common technique for electrically attaching chips to packages. Wire bonds are formed by compression welding one end of a thin (1-mil-diameter) gold wire to a bond pad (typically 100 μm on a side on 125 μm centers) on the chip and the other end to a pad on the package. Wire bonds are inexpensive and compliant, allowing unequal thermal expansion of the chip and package. However, they have significant inductance, about 1 nH/mm, and only allow one or two rows of pads around the periphery of the chip.

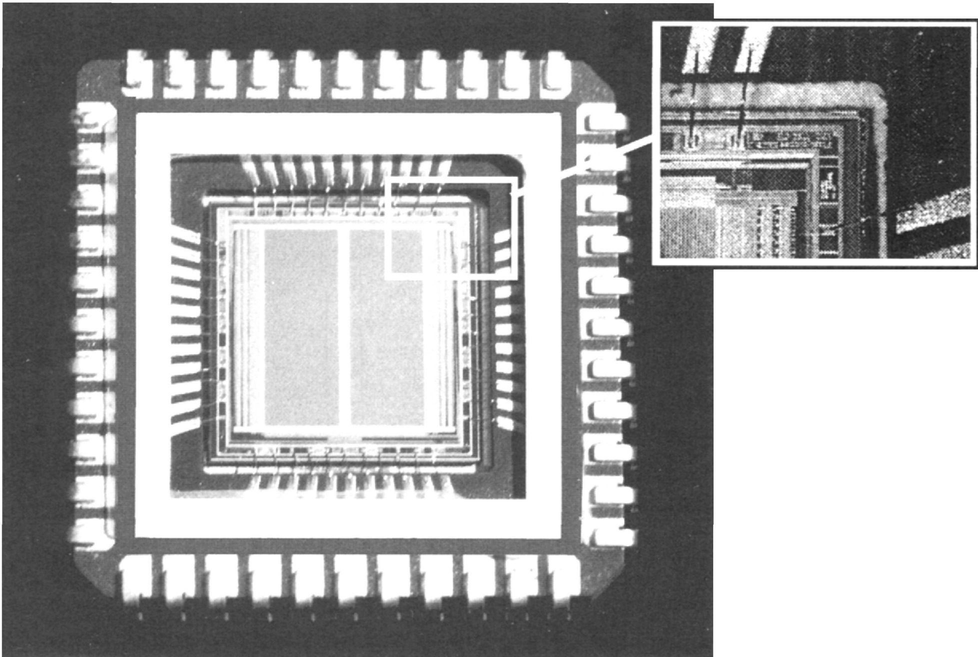


FIGURE 2-2 Wire-Bonding a Chip to a Package

An area array of solder balls can also be used to attach a chip to a package, as in Figure 2-3. In this case, a chip's aluminum bond pads are processed with additional metal layers to allow adhesion of solder balls to pads. This style of die-to-package connection is often called "C4" or "flip-chip" after pioneering packaging processes developed at IBM. Connections to a PC board or package are made by depositing a solder paste on an array of contacts (on either side of the board), placing the chip in rough alignment, and then heating the assembly to reflow the solder. The surface tension of the solder pulls the chip into precise alignment. Solder ball contacts are typically placed on 250-μm centers, and the balls themselves are 100 μm or less in diameter.

Solder balls have the advantages of very low inductance (a few tenths of a nanohenry per contact) and the ability to form contacts with pads located over the entire area of the chip, not just the periphery. Area bonding is particularly advantageous for power distribution because it allows the global power distribution to be done in thick-film (20 μm typical) power planes in the package rather than in thin-film (1 μm typical) wires on the chip. Area bonding also significantly

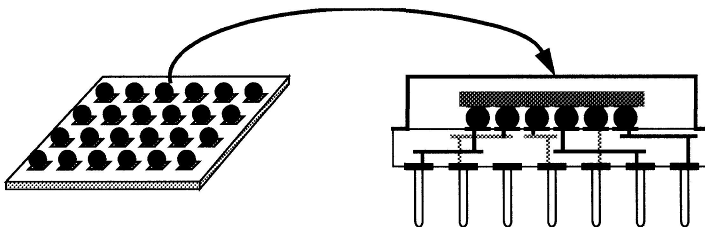


FIGURE 2-3 Flip-Chip Bonding

reduces the number of chips that are pad-limited, since a very large number of pads can be placed in a small area. However, because area bonding requires expensive multilayer packages and special processing to apply the solder, its use tends to be reserved for expensive, high-performance chips.

2.3.2 Package Types

Integrated circuit packages come in two primary varieties: *through-hole* and *surface-mount*. Through-hole packages have pins that are inserted into through-holes in the circuit board and soldered in place from the opposite side of the board. Surface-mount packages, on the other hand, have leads or lands that are soldered directly to corresponding exposed metal lands on the top surface of the circuit board. Surface-mount packages can be attached to both sides of a board. Because they eliminate through-holes, which are costly to drill, and reduce circuit-board wiring density, surface-mount packages are generally preferred.

Integrated circuit packages intended for through-hole mounting (see Section 2.4.6) are shown in Figure 2-4 (dual-in line packages or DIPs) and Figure 2-5 (pin-grid arrays or PGAs). The PGAs are shown from both the top and bottom for three different pin counts.

DIPs have lead counts between 8 and 64, with leads on 100-mil centers (a *mil* is 0.001 inch or 25.4 μm). DIP packages are made from either plastic or ceramic materials, as discussed in the next section. They have the unpleasant characteristic that their corner leads are much longer than their center leads, and thus there is a wide range of parasitic lead inductance and capacitance. DIPs are large by today's standards, generally have poor high-frequency performance, and are nearly obsolete.

Pin-grid arrays are usually made of ceramic materials, but relatively inexpensive plastic versions are also available. They have lead counts that range from 64 up to 500 or so. The smaller pin-count packages have pins on 100-mil centers in several rows around the outside of the package; larger PGAs, such as the one shown on the right of Figure 2-5, have pins on 50-mil centers in staggered rows. This example package also has provisions for mounting discrete power supply bypass capacitors.

Most modern IC packages are intended for surface attachment to a PC board. Examples of the most common types are shown in Figure 2-6. Leaded chip carriers

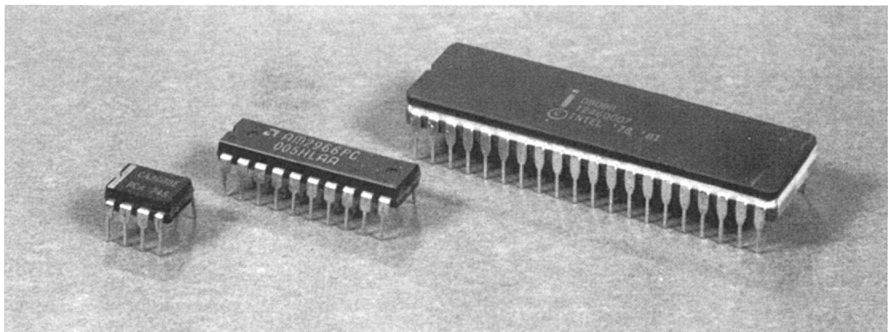


FIGURE 2-4 Examples of Dual-In Line (DIP) Packages

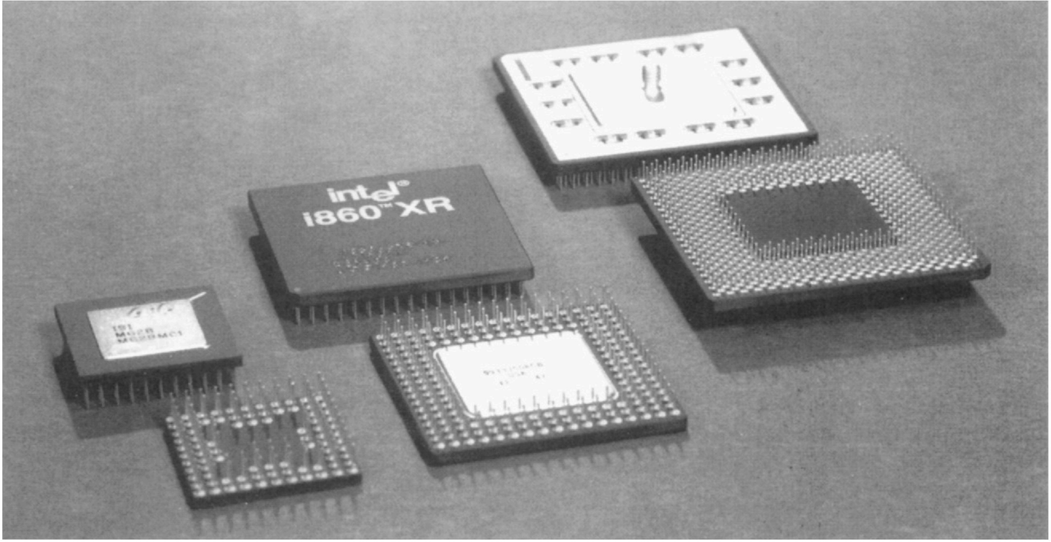


FIGURE 2-5 Examples of Pin-Grid Array (PGA) Packages

(LCCs) are shown at the center of the figure. They have leads emerging from all four sides of the package, and the external part of each lead is bent into a J form that tucks up underneath the package. This arrangement allows some flexibility in the lead so that, as the board and package heat up at different rates, differential expansion between the two can be accommodated. LCCs are made using plastic, ceramic, or metal manufacturing processes. Their leads are on 50-mil (1.27-mm) centers, and the packages are available in lead-counts from 20 to 84.

Small-outline (SO) packages, shown at the right of Figure 2-6, were intended to replace DIPs. They are found with gull-wing leads, like those shown in the figure, and less often with J leads; both lead types provide thermal expansion compliance. These packages are generally made of plastic. Lead pitch is 50 mils for all but the TSOP (tiny small-outline package) style, which has smaller, typically 25-mil, pitch.

At the left side of the figure are quad flat pack (QFP) packages. Available in plastic, metal, and ceramic variations, these packages have gull-wing style leads and pin counts ranging from 44 to 352. QFPs are available in "families," which share a fixed body size but have varying lead counts. Lead pitch varies from 1.00 mm down to 0.40 mm.

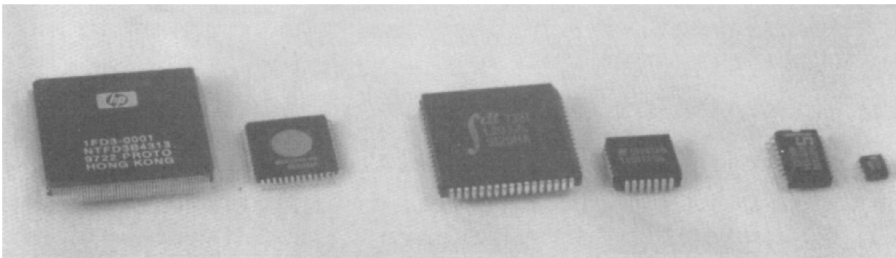


FIGURE 2-6 Examples of Surface-Mount Packages

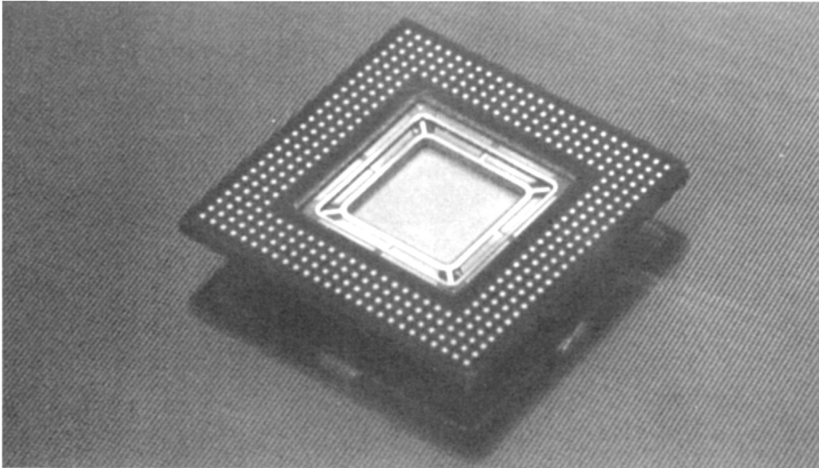


FIGURE 2-7 Ball-Grid Array (BGA) Packages

Figure 2-7 shows a typical ball-grid array (BGA) package. These packages are made using either ceramic or PC-board-like laminates. Package "leads" consist of a solder ball attached to a package pad, and these balls are arranged in a grid on the bottom of the package.

There are two types of BGAs. Type-I or *cavity-up* BGAs have a die-attach cavity and wire bond pad area on the top of the package. These are wired through internal wiring layers in the package to the bottom pads and balls, and the array of ball connections generally covers the entire bottom of the package. Type-II or *cavity-down* BGAs have their die-attach area on the bottom of the package, with several rows of balls surrounding the die cavity, like the package shown in Figure 2-7. The advantage of this arrangement is that the top of the package can be made into a thick metal heat slug to which the die is attached on the bottom side. This heat slug can easily be augmented with an external heat sink. Thus, where Type-I's are usually limited to dissipation less than a watt or so, Type-II BGAs can handle chips that have power dissipation up to tens of watts. BGAs are still fairly new (1997), and standards for size, shape, and lead count have not been fully settled. Currently, off-the-shelf packages are available with lead counts up to 400 or so and with balls on a 50-mil (or greater) grid. Some custom BGA packages have been built with lead counts up to 1,000.

Most of the package types we have discussed are available in two different forms, cavity-up and cavity-down, a designation that describes which side of the package has the die-attach cavity. As with Type-II BGAs, cavity-down packages mount the IC die into the bottom of the package and allow heat to be drawn out of the top of the package, often with an attached heat sink. Cavity-down packages can therefore handle much higher power dissipation than cavity-up packages, which shed heat mainly through their leads into the PC board.

Lastly, each of the package types we have discussed is generally available in a range of die cavity sizes. This avoids the necessity of placing a small die in a large cavity, a situation which would result in very long bond wires. Designers usually

want to keep bond wires as short as possible to reduce their electrical parasitics and improve their mechanical robustness.

2.3.3 Package Manufacturing Processes

Integrated circuit packages are made from plastic, ceramic, PC-board-like laminates, and metal. Often, IC manufacturers will encode the material type in the package abbreviation, and thus a "PQFP" is a plastic quad flat pack, a "CERDIP" is a ceramic DIP, and so forth.

Plastic packages are formed by die-bonding and wire-bonding the chip to a metal (usually copper) lead frame and then encapsulating the assembly in injection-molded plastic. Plastic packages are inexpensive, but, because they lack ground planes, leads do not have controlled impedances and generally have high inductance. Plastic packages usually have high thermal resistance, making it difficult to cool a chip that dissipates a great deal of power. An array of thermal vias or a metal *slug* is often used to improve the heat conduction of a plastic package.

Metal package analogs exist for the plastic LCC and QFP packages. In these packages, the integrated circuit die is mounted and wire-bonded to a metal lead frame that is then captured between two thin metal shells. An epoxy resin is injected between the shells to seal the assembly, to insulate the lead frame from the case, and to cement the assembly together. The lead frame is in thermal contact with the metal shell, thereby greatly improving the thermal performance of the package relative to the plastic version. The leads of the package emerge through thin slits in the metal case, and the proximity of the (electrically floating) metal case reduces the lead mutual and self-inductances. Metal QFPs generally have exactly the same outside dimensions as their plastic cousins and so can be used interchangeably in a given PC-board design.

A ceramic package consists of several layers of conductors, usually about 100 μm wide on 200-μm centers separated by layers of insulating ceramic, usually alumina (Al_2O_3). The chip is placed in a cavity in the package and bonded (wire or solder ball) to the conductors. A metal lid is then soldered on the package to seal the cavity against the environment. Because multiple layers are used, power supply and signal return planes can be included in the package, reducing supply inductance and providing controlled-impedance signal paths. One disadvantage is that the high permittivity of alumina ($\varepsilon_r = 10$) results in a low propagation velocity in the package but is advantageous for power–ground connection layers because additional capacitance is a significant benefit for supply bypassing. It is also possible to include discrete bypass capacitors within a ceramic package.

Laminated packages are similar to ceramic ones but are built using the same materials and processes used for PC boards. Like the ceramic package, a chip is placed in a cavity in the laminated package and bonded to the conductors. The multiple layers of conductors permit the use of supply and return planes and controlled impedance lines. Compared with ceramic packages, laminated packages enjoy the advantage of a relatively low permittivity ($\varepsilon_r = 4$). They have the disadvantage that they are not as well sealed, particularly against water vapor, as a ceramic package, making them unsuitable for use in a hostile environment.

2.3.4 Multichip Modules

With multiple wiring layers and supply planes, ceramic and laminated packages can hold and interconnect several integrated circuits. By making interchip connections locally, these *multichip* modules or MCMs reduce the pin count of the package and the capacitance of the interchip wires. However, MCMs are usually more expensive than single-chip packages and increase the size (and cost) of the smallest unit that can be independently tested and replaced.

2.3.5 A Typical Package Model

In this section we will develop a simple electrical model for a typical package, a type-II BGA. We will assume that the 30-mm square package houses a 15-mm square chip and provides about 300 signal pins between chip and board.

2.3.5.1 Physical Construction

The construction of a typical type-II BGA is shown in Figure 2-8. It is built using PC board fabrication techniques. The interconnect for the package is of multilayer laminated construction and is bonded to a metal substrate that serves as the heat sink for the package. The interconnect portion of the package has a large central hole that allows a die to be bonded onto the metal heat slug. Solder balls (shown at the top of the figure) make connections to a PC board, and thus the package is referred to as a cavity-down design because the die is toward the bottom of the package.

Copper conductors in the laminate are 20 μm thick, and signal leads are drawn about 75 μm wide. Planes are provided for power supply and ground connections. A dozen or so solder balls are used to connect each of these planes to the printed circuit on which the package is mounted. Plane conductors are wrapped around the edges of their dielectrics to form continuous annular metal lands around the die area. Power and ground bond wires can be connected anywhere along these lands. Signal and plane layers are separated from neighboring planes by 150 μm. Some of the details of BGA construction can be seen in Figure 2-8.

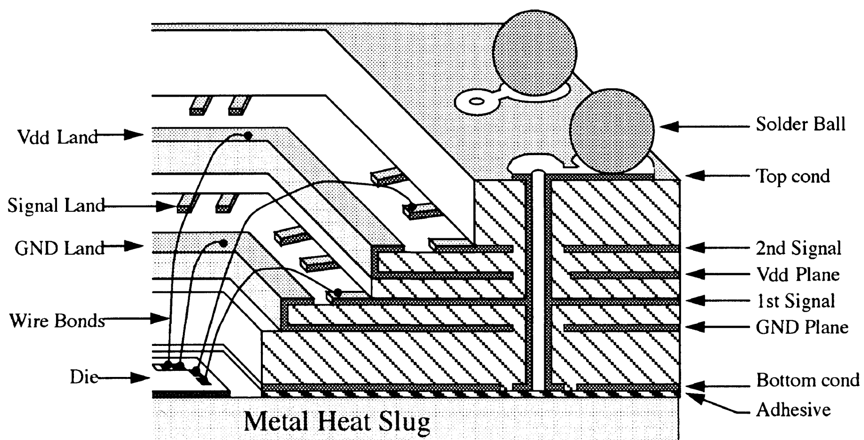


FIGURE 2-8 Typical Type-II BGA Construction

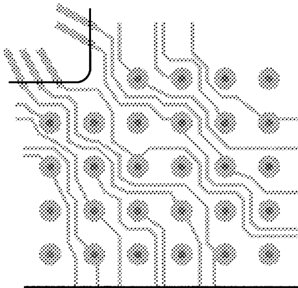


FIGURE 2-9 Typical Signal Wiring Pattern

The chip's I/O pads are connected to package lands with 50-μm-diameter bond wires. There are two tiers of signal lands on the package to which bond wires are attached. The longest bond wires, connected to the upper tier at the package corners, are about 6 mm long; the shortest, connected to the lower tier at the center of each side, are about 3 mm long.

A typical wiring pattern for a signal layer in a lamintated BGA is shown in Figure 2-9. The following features are worth noting:

1. A typical wiring trace may run at minimum distance from other traces for a substantial fraction of its total length, and thus neighboring traces are strongly coupled.
2. Wiring traces are often run from the wire-bond land to a via (that connects to a solder-ball land on the surface) and then onward to the edge of the package. Package laminates are made up on large panels that hold many packages, and, before these are cut into individual parts, wiring traces are connected to a common terminal that allows gold to be electroplated onto the wire-bond lands. Unfortunately, the vestigial traces out to the package edge form unterminated stubs that tend to degrade signal quality. Not all package manufacturing processes require these extra traces, and they should be avoided if possible.
3. Signal wiring traces are 10–15 mm in length. Signals propagate along them at about half the free-space speed of light, about 150 mm/ns, and thus package traces represent a delay of about 100 ps. This time is short compared with most signal rise and fall times; therefore, we can treat package leads as lumped circuit elements with reasonable accuracy.

2.3.5.2 Package Electrical Model

The package electrical model has four parts, as shown in Figure 2-10: the bond wire between chip and package, the package land to which the bond wire is attached, the signal trace between land and via, and the via and solder ball (perhaps including the PC board land to which the ball is attached). In principle, an electrical model of this package would have to take into account all of the currents and charges on a complex three-dimensional arrangement of conductors and dielectrics. In practice, we can use fairly simple tools to build a lumped-circuit-element equivalent model that will serve well for most high-performance signaling designs.

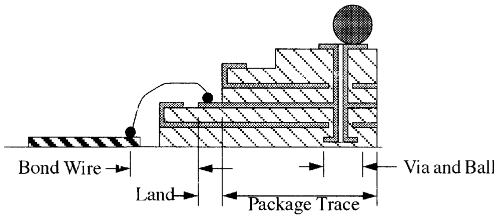


FIGURE 2-10 Package Electrical Model

For the bond wire, we can develop a crude model by assuming that the wire is a uniform round conductor over a conducting plane. In Chapter 3 we will develop some simple formulas for capacitance and inductance per unit length for this situation (Eqs. 3-6 and 3-9, respectively); applying these formulas and assuming a 50-μm-diameter ($2r$) wire 0.5 mm away from a plane

$$C = \frac{2\pi \epsilon_0}{\ln\left(2\frac{s}{r}\right)} = 15 \text{ fF/mm} \tag{2-1}$$

$$L = \frac{\mu_0 \ln\left(2\frac{s}{r}\right)}{2\pi} = 0.75 \text{ nH/mm}$$

The capacitance of a bond wire is almost always negligible compared with the capacitance at its two ends (the chip I/O pad and the wire-bond land), and thus a single lumped inductor is a reasonably accurate model. A value of 1 nH per millimeter of length is a good rule of thumb for the inductance of bond wires.

The signals on neighboring bond wires interact via coupling capacitance and mutual inductance. We can estimate these effects either by using exact or empirical formulas (outlined in Chapter 3 and in several of the references) or by using a two-dimensional field solver such as LINPAR [DjorHarr89]. Bond wires typically run parallel for their entire length, spaced apart by about 200 μm or so. Mutual capacitance is about 20 fF/mm between neighboring bond wires. Next nearest neighbors are reasonably well shielded for mutual capacitance; therefore, this coupling can usually be ignored. However, mutual inductances may need to be considered for wires that are some distance apart. Mutual inductance coupling coefficients are about 0.4, 0.3, and 0.2 for the geometry we have assumed for wires that are 1, 2, and 3 spacings apart, respectively.

A 2D solver is also useful for estimating the effects of the land and package trace in our model. The actual impedances of traces and coupling between neighboring traces are quite complex, but a useful model can be constructed assuming uniform traces spaced an "average" distance apart. If we assume an average 100-μm spacing for lands and traces and a dielectric constant of 4.7 (for the glass-epoxy material typical of laminated packages), then the lands can be modeled as small lumped capacitors of about 0.15 pF and coupling capacitors between lands of about 0.08 pF. The package traces have capacitance to the neighboring planes of about 90 fF/mm; coupling capacitance to the neighboring lines of about 40 fF/mm; series inductance of about 0.7 nH/mm; and first, second, and third mutual inductive coupling coefficients of about 0.5, 0.3, and 0.2, respectively.

The via, solder ball, and PC board trace have very low inductance compared with the signal traces on the package and board; a reasonably accurate model is