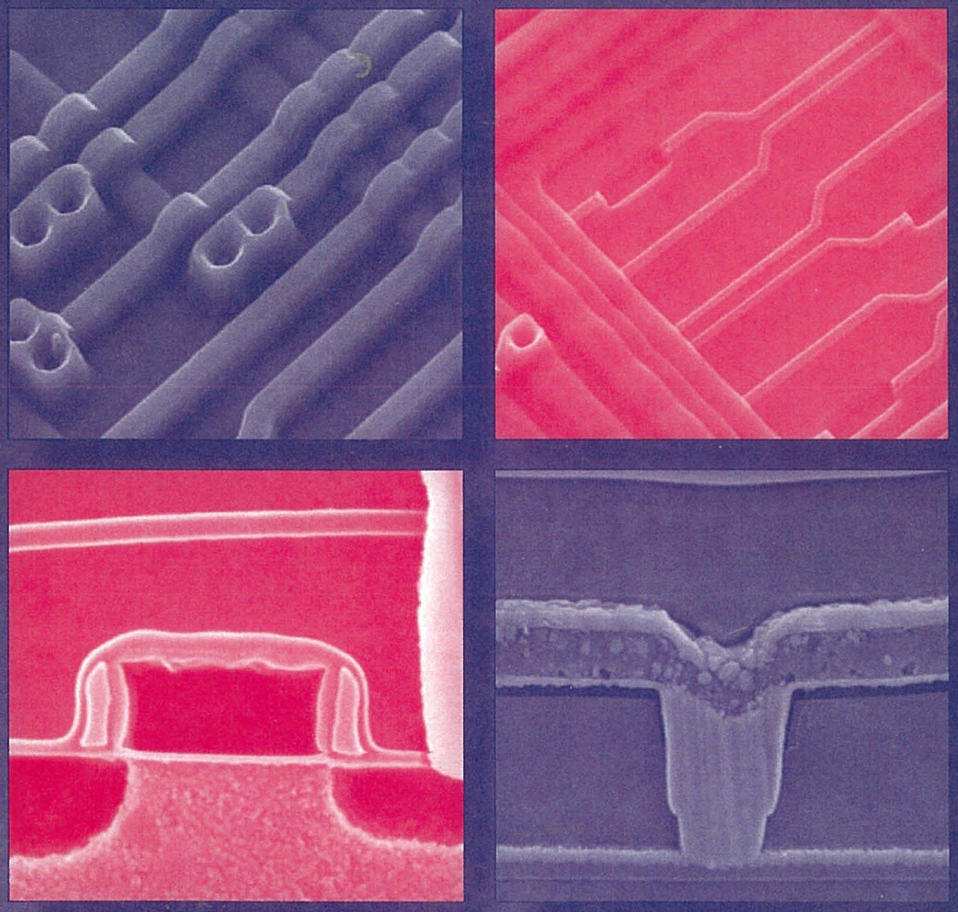
# CMOS

## CIRCUIT DESIGN, LAYOUT, AND SIMULATION



R. Jacob Baker ■ Harry W. Li ■ David E. Boyce

IEEE Press Series on Microelectronic Systems
Stuart K. Tewksbury, *Series Editor*

Covers downloading and using the LASI Layout Software

Microchip Ex. 1055, p. 1
Microchip v. Aptiv
IPR2024-00494

# Process Information for Hand-Calculations

| Process | *CN*20 NMOS | *CN*20 PMOS | *CMOS*14 NMOS | *CMOS*14 PMOS |
|---|---|---|---|---|
| | | | | |
| Information | App. A | App. A | App. C | App. C |
| $VDD$ ($VSS = 0$) | 5 V | 5 V | 3.3 V | 3.3 V |
| minimum $L_{drawn}$ | 2.0 μm | 2.0 μm | 0.6 μm | 0.6 μm |
| $DL = L_{drawn} - L_{eff}$ | 0.6 μm | 0.8 μm | 0.06 μm | 0.09 μm |
| minimum $W_{drawn}$ | 3.0 μm | 3.0 μm | 0.9 μm | 0.9 μm |
| $DW = W_{drawn} - W_{eff}$ | 0.14 μm | 0.16 μm | 0.35 μm | 0.35 μm |
| $V_{THN}$ or $V_{THP}$ | 0.83 V | 0.91 V | 0.7 V | 0.9 V |
| KP, μA/V$^2$ | 50 | 17 | 170 | 50 |
| λ, V$^{-1}$ | 0.06 for $L \geq 5$ μm | 0.06 for $L \geq 5$ μm | | |
| $I_{drive}$ (μA/μm) | | | 380 (typ.) | 190 (typ.) |
| $C'_{ox}$ | 800 aF/μm$^2$ | 800 aF/μm$^2$ | 3.7 fF/μm$^2$ | 3.7 fF/μm$^2$ |
| $R_n$ or $R_p$ | 12 kΩ · ($L$/$W$) | 36 kΩ · ($L$/$W$) | (9 kΩ · μm)/$W$ | (18 kΩ · μm)/$W$ |
| $\tau_n$ or $\tau_p$ | 38 ps | 114 ps | 20 ps | 40ps |

# Multipliers

| Name | Symbol | Value |
|---|---|---|
| | | |
| terra | T | $10^{12}$ |
| giga | G | $10^{9}$ |
| mega | M or (MEG) | $10^{6}$ |
| kilo | k | $10^{3}$ |
| milli | m | $10^{-3}$ |
| micro | μ (or u) | $10^{-6}$ |
| nano | n | $10^{-9}$ |
| pico | p | $10^{-12}$ |
| femto | f | $10^{-15}$ |
| atto | a | $10^{-18}$ |

# Physical Constants

| Name | Symbol | Value/Units |
|---|---|---|
| | | |
| Vacuum dielectric constant | $\varepsilon_0$ | 8.85 aF/μm |
| Silicon dielectric constant | $\varepsilon_{si}$ | $11.7\varepsilon_0$ |
| $SiO_2$ dielectric constant | $\varepsilon_{ox}$ | $3.97\varepsilon_0$ |
| $SiN_3$ dielectric constant | $\varepsilon_{Ni}$ | $16\varepsilon_0$ |
| Boltzmann's constant | k | $1.38 \times 10^{-23}$ J/K |
| Electronic charge | q | $1.6 \times 10^{-19}$ C |
| Temperature | T | K |
| Thermal voltage | $V_T$ | kT/q<br>= 26 mV @ 300 K |

# Equations

| Parameter | NMOS equations in terms of BSIM1 parameters |
|---|---|
| | for PMOS use $V_{SD}$, $V_{SG}$, $V_{BS}$, and $V_{THP}$ |
| | |
| $V_{THN}$ | $VFB + PHI + K1 \cdot \sqrt{PHI + V_{SB}} - K2 \cdot (PHI + V_{SB})$ |
| $C'_{ox}$ | $\varepsilon_{ox}/TOX$ |
| $KP$ | $MUZ \cdot C'_{ox}$ |
| β | $KP \cdot (W/L)$ |
| $I_D$ (triode)<br>$V_{DS} \le V_{GS} - V_{THN}$ | $\beta\left((V_{GS} - V_{THN})V_{DS} - \frac{V_{DS}^2}{2}\right)$ (for Long L) |
| $I_D$ (saturation)<br>$V_{DS} \ge V_{GS} - V_{THN}$ | $\frac{\beta}{2}(V_{GS} - V_{THN})^2[1 + \lambda(V_{DS} - V_{DS,sat})]$ (for Long L) |
| $g_m$ | $\beta(V_{GS} - V_{THN}) = \sqrt{2\beta I_D}$ or $V_T/I_D$ for subthreshold |
| η | $K1 \cdot \left(2\sqrt{PHI + V_{SB}}\right)^{-1} - K2$ and $g_{mb} = g_m \cdot \eta$ |
| $r_o$ | $1/(\lambda I_D)$ |

# CMOS

Microchip Ex. 1055, p. 4
Microchip v. Aptiv
IPR2024-00494

IEEE Press
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331

**Editorial Board**
John B. Anderson, *Editor in Chief*

| | | |
|---|---|---|
| P. M. Anderson | R. Herrick | R. S. Muller |
| M. Eden | G. F. Hoffnagle | W. D. Reeve |
| M. E. El-Hawary | R. F. Hoyt | D. J. Wells |
| S. Furui | S. Kartalopoulos | |
| A. H. Haddad | P. Laplante | |

Kenneth Moore, *Director of IEEE Press*
John Griffin, *Senior Editor*
Linda Matarazzo, *Assistant Editor*

IEEE Circuits & Systems Society, *Sponsor*
CAS-S Liaison to IEEE Press, Phillip Allen
IEEE Solid-State Circuits Society, *Sponsor*
SSC-S Liaison to IEEE Press, Stuart K. Tewksbury

**Technical Reviewers**

Jeff Bruce, *Micron Technology, Inc.*
Alan Buchholz, *Comlinear Corporation*
Ian Galton, *University of California, Irvine*
Bruce Johnson, *University of Nevada, Reno*
Joseph Karniewicz, *Micron Technology, Inc.*
Brent Keeth, *Micron Technology, Inc.*
William Kuhn, *Kansas State University*
Wen Li, *Micron Technology, Inc.*
Brian P. Lum Shue Chan, *Crystal Semiconductor Corporation*
Alan Mantooth, *Analogy, Inc.*
Adrian Ong, *Stanford University*
Terry Sculley, *Crystal Semiconductor Corporation*
Don Thelen, *Analog Interfaces*
Axel Thomsen, *Crystal Semiconductor Corporation*
Kwang Yoon, *Inha University, Korea*

**Microchip Ex. 1055, p. 5**
**Microchip v. Aptiv**
**IPR2024-00494**

# CMOS

## Circuit Design, Layout, and Simulation

R. Jacob Baker, Harry W. Li and David E. Boyce
*Department of Electrical Engineering*
*Microelectronics Research Center*
*The University of Idaho*

IEEE Press Series on Microelectronic Systems
Stuart K. Tewksbury, *Series Editor*

IEEE Circuits & Systems Society, *Sponsor*
IEEE Solid-State Circuits Society, *Sponsor*



**IEEE**
**PRESS**

The Institute of Electrical and Electronics Engineers, Inc., New York

**Microchip Ex. 1055, p. 6**
**Microchip v. Aptiv**
**IPR2024-00494**

This book and other books may be purchased at a discount
from the publisher when ordered in bulk quantities. Contact:

IEEE Press Marketing
Attn: Special Sales
Piscataway, NJ 08855-1331
Fax: (908) 981-9334

For more information about IEEE PRESS products,
visit the IEEE Home Page: http://www.ieee.org/

© 1998 by the Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394

*All rights reserved. No part of this book may be reproduced in any form, nor may it be stored in a retrieval system or transmitted in any form, without written permission from the publisher.*

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

**ISBN 0-7803-3416-7**
**IEEE Order Number: PC5689**

**Library of Congress Cataloging-in-Publication Data**

Baker, R. Jacob (date)
CMOS circuit design, layout, and simulation / R. Jacob Baker, Harry W. Li, and David E. Boyce.
p. cm. -- (IEEE Press series on microelectronic systems)
Includes bibliographical references and index.
ISBN 0-7803-3416-7
1. Metal oxide semiconductors, Complementary--Design and construction. 2. Integrated circuits--Design and construction. 3. Metal oxide semiconductor field-effect transistors. I. Li, Harry W. II. Boyce, David E. III. Title. IV. Series
TK7871.99.M44B35 1997
621.3815--DC21 97-21906
CIP

**Microchip Ex. 1055, p. 7**
**Microchip v. Aptiv**
**IPR2024-00494**

*To*

*Julie and Melanie*

**Microchip Ex. 1055, p. 8**
**Microchip v. Aptiv**
**IPR2024-00494**

# Contents


**Preface** xix

## Part I CMOS Fundamentals 1

### Chapter 1 Introduction 3

1.1 The CMOS IC Design Process ..... 3

1.1.1 Fabrication 5

1.2 Using the LASI Program ..... 5

1.2.1 Cells in LASI 7

1.2.2 Navigating LASI 10

1.2.3 Adding Objects 10

1.2.4 Editing Objects 11

1.2.5 Placing Cells 12

1.2.6 Common Problems 14

1.3 MOSIS ..... 15

### Chapter 2 The Well 23

2.1 The Substrate ..... 23

2.1.1 Patterning 24

2.1.2 Patterning the N-well 27

2.2 Laying out the N-well ..... 27



**Microchip Ex. 1055, p. 9**
**Microchip v. Aptiv**
**IPR2024-00494**

2.2.1 Design Rules for the N-well 28
2.2.2 Using the LASIDRC program 30
2.3 Resistance Calculation ..... 30
2.3.1 The N-well Resistor 32
2.4 The N-well/Substrate Diode ..... 32
2.4.1 Depletion Layer Capacitance 33
2.4.2 Storage Capacitance 35
2.4.3 SPICE Modeling 37
2.5 The RC Delay Through an N-well ..... 38
**Chapter 3 The Metal Layers 45**
3.1 The Bonding Pad ..... 45
3.1.1 Laying out the Pad 45
3.1.2 Design Rules for Pads 48
3.2 Design and Layout Using the Metal Layers ..... 50
3.2.1 Design Rules for the Metal Layers 50
3.2.2 Parasitics Associated with the Metal Layers 50
3.2.3 Current Carrying Limitations 54
3.2.4 Parasitics Associated with the Via 55
3.3 Crosstalk and Ground Bounce ..... 56
3.3.1 Ground Bounce 57
3.4 Layout Using Cell Hierarchy ..... 59
**Chapter 4 The Active and Poly Layers 65**
4.1 Design Rules ..... 66
4.1.1 Design Rules for the n+/p+ Active Layers 66
4.1.2 Design Rules for Poly1 69
4.2 Layout of a Standard Cell Frame ..... 71
4.3 Patterning the Active Layers ..... 71
4.4 Layout of the MOSFET ..... 77
4.4.1 Parasitics Associated with the Active Layers 77
**Chapter 5 The MOSFET 83**
5.1 The MOSFET Capacitances ..... 84
5.1.1 Case I: Accumulation 84
5.1.2 Case II: Depletion 85

**Microchip Ex. 1055, p. 10**
**Microchip v. Aptiv**
**IPR2024-00494**

5.1.3 Case III: Strong Inversion 86
5.1.4 Summary 87
5.2 The Threshold Voltage ..... 88
5.3 IV Characteristics of MOSFETs ..... 93
5.3.1 MOSFET Operation in the Triode Region 94
5.3.2 The Saturation Region 96
5.4 SPICE Modeling of the MOSFET ..... 98
5.4.1 Level 1 Model Parameters Related to $V_{THN}$ 98
5.4.2 Level 1 Model Parameters Related to Transconductance 99
5.4.3 SPICE Modeling of the Source and Drain Implants 100
5.4.4 Layout of the MOSFET 102
**Chapter 6 The BSIM SPICE Model 107**
6.1 BSIM1 Model Parameters ..... 109
6.2 BSIM1 DC Equations ..... 111
6.2.1 The Threshold Voltage 111
6.2.2 The Drain Current 113
6.2.3 The Subthreshold Current 116
6.3 Short Channel MOSFETs ..... 119
6.3.1 MOSFET Scaling 120
6.3.2 Short Channel Effects 122
6.4 The BSIM3 SPICE Model ..... 124
6.5 Convergence ..... 127
**Chapter 7 CMOS Passive Elements 131**
7.1 The Second Poly Layer (poly2) ..... 131
7.1.1 Design Rules for Capacitor Formation 131
7.1.2 Parasitics of the Poly Cap 132
7.1.3 Other Types of Capacitors 134
7.2 Temperature and Voltage Dependence of Capacitors and Resistors .... 134
7.2.1 Resistors 134
7.2.2 Capacitors 140
7.3 Noise in Resistors ..... 141
**Chapter 8 Design Verification with LASICKT 149**
8.1 Fundamentals of LASICKT ..... 150

**Microchip Ex. 1055, p. 11**
**Microchip v. Aptiv**
**IPR2024-00494**

8.1.1 The Inverter 151
8.1.2 Running LASICKT 157
8.1.3 Higher-ranking Cells; The OR Gate 160

**Chapter 9 Analog MOSFET Models 165**

9.1 Low-Frequency MOSFET Model ..... 165
9.1.1 Small-Signal Model of the MOSFET in Saturation 166
9.2 High-Frequency MOSFET Model ..... 171
9.2.1 Variation of Transconductance with Frequency 173
9.3 Temperature Effects in MOSFETs ..... 175
9.4 Noise in MOSFETs ..... 178

**Chapter 10 The Digital Model 185**

10.1 The Digital MOSFET Model ..... 185
10.1.1 Capacitive Effects 187
10.1.2 Process Characteristic Time Constant 188
10.1.3 Delay- and Transition-Times 189
10.2 Series Connection of MOSFETs ..... 192
10.2.1 DC Behavior of Series-Connected MOSFETs 192
10.2.2 Delay Through Series-Connected MOSFETs 194
Possible Student Projects ..... 196

**Part II CMOS Digital Circuits 199**

**Chapter 11 The Inverter 201**

11.1 DC Characteristics ..... 202
11.1.1 Noise Margins 203
11.1.2 Inverter Switching Point 204
11.2 Switching Characteristics ..... 205
11.2.1 The Ring Oscillator 209
11.2.2 Dynamic Power Dissipation 210
11.3 Layout of the Inverter ..... 212
11.3.1 Latch-up 212
11.4 Sizing for Large Capacitive Loads ..... 216
11.4.1 Distributed Drivers 221

Microchip Ex. 1055, p. 12
Microchip v. Aptiv
IPR2024-00494

11.4.2 Driving Long Lines 223
11.5 Other Inverter Configurations ..... 224
11.5.1 N-Channel Only Output Drivers 224
11.5.2 Inverters with Tri-State Outputs 226
11.5.3 The Bootstrapped NMOS inverter 226
**Chapter 12 Static Logic Gates 231**
12.1 DC Characteristics of the NAND and NOR Gates ..... 231
12.1.1 DC Characteristics of the NAND Gate 231
12.1.2 DC Characteristics of the NOR Gate 234
12.2 Layout of the NOR and NAND Gates ..... 235
12.3 Switching Characteristics ..... 236
12.3.1 NAND Gate 238
12.3.2 Number of Inputs 242
12.4 Complex CMOS Logic Gates ..... 243
12.4.1 Cascode Voltage Switch Logic 247
12.4.2 Differential Split-Level Logic 247
12.4.3 Tri-State Outputs 252
**Chapter 13 The TG and Flip-Flops 255**
13.1 The Pass Transistor ..... 255
13.2 The CMOS TG ..... 257
13.2.1 Layout of the CMOS TG 258
13.2.2 Series Connection of Transmission Gates 260
13.3 Applications of the Transmission Gate ..... 260
13.4 The Flip-Flop ..... 265
13.4.1 Clocked Flip-Flops 266
**Chapter 14 Dynamic Logic Gates 275**
14.1 Fundamentals of Dynamic Logic ..... 275
14.1.1 Charge Leakage 276
14.1.2 Simulating Dynamic Circuits 277
14.1.3 Nonoverlapping Clock Generation 279
14.1.4 CMOS TG in Dynamic Circuits 281
14.2 Clocked CMOS Logic ..... 281

**Microchip Ex. 1055, p. 13**
**Microchip v. Aptiv**
**IPR2024-00494**

**Chapter 15 VLSI Layout** 289
15.1 Chip Layout ..... 290
15.2 Layout Steps ..... 301
**Chapter 16 BiCMOS Logic Gates** 313
16.1 Layout of the Lateral BJT ..... 314
16.2 Modeling the lateral NPN ..... 316
16.3 The BiCMOS Inverter ..... 320
16.4 Other BiCMOS Logic Gates ..... 323
16.5 CMOS and ECL Conversions Using BiCMOS ..... 326
**Chapter 17 Memory Circuits** 331
17.1 RAM Memory Cells ..... 332
17.1.1 The DRAM Cell 333
17.1.2 The SRAM cell 337
17.2 The Sense Amplifier ..... 338
17.3 Row/Column Decoders ..... 341
17.4 Timing Requirements for DRAMs ..... 343
17.5 Modern DRAM Circuits ..... 345
17.5.1 DRAM Memory Cell Layout 345
17.5.2 Folded/Open Architectures 347
17.6 Other Memory Cells ..... 350
**Chapter 18 Special-Purpose Digital Circuits** 355
18.1 The Schmitt Trigger ..... 355
18.1.1 Design of the Schmitt Trigger 356
18.1.2 Switching Characteristics 359
18.1.3 Applications of the Schmitt Trigger 359
18.1.4 High-Speed Schmitt Trigger 361
18.2 Multivibrator Circuits ..... 362
18.2.1 The Monostable Multivibrator 362
18.2.2 The Astable Multivibrator 363
18.3 Voltage Generators ..... 364
18.3.1 Improving the Efficiency 366
18.3.2 Generating Higher Voltages 367
18.3.3 Example 368

**Microchip Ex. 1055, p. 14**
**Microchip v. Aptiv**
**IPR2024-00494**

**Chapter 19 Digital Phase-Locked Loops** 373
19.1 The Phase Detector ..... 375
19.1.1 The XOR Phase Detector 375
19.1.2 Phase Frequency Detector 379
19.2 The Voltage-Controlled Oscillator ..... 383
19.2.1 The Current-Starved VCO 384
19.2.2 Source Coupled VCOs 387
19.3 The Loop Filter ..... 391
19.3.1 XOR DPLL 391
19.3.2 PFD DPLL 398
19.4 System Considerations ..... 403
19.4.1 Clock Recovery from NRZ Data 407
19.5 Delay-Locked Loops ..... 417

**Part III CMOS Analog Circuits** 425

**Chapter 20 Current Sources and Sinks** 427
20.1 The Current Mirror ..... 427
20.1.1 The Cascode Connection 433
20.1.2 Sensitivity Analysis 438
20.1.3 Temperature Analysis 439
20.1.4 Transient Response 442
20.1.5 Layout of the Simple Current Mirror 444
20.1.6 Matching in MOSFET Mirrors 446
20.2 Other Current Sources/Sinks ..... 450
**Chapter 21 References** 463
21.1 Voltage Dividers ..... 463
21.1.1 The Resistor-MOSFET Divider 464
21.1.2 The MOSFET-Only Voltage Divider 465
21.2 Current Source Self-Biasing ..... 469
21.2.1 Threshold Voltage Referenced Self-Biasing 469
21.2.2 Diode Referenced Self-Biasing 470
21.2.3 Thermal Voltage Referenced Self-Biasing 474

**Microchip Ex. 1055, p. 15**
**Microchip v. Aptiv**
**IPR2024-00494**

21.3 Bandgap Voltage References ..... 477
21.3.1 Bandgap Referenced Biasing 478
21.4 Beta Multiplier Referenced Self-Biasing ..... 480
21.4.1 A Voltage Reference 481
21.4.2 Operation in the Subthreshold Region 483
**Chapter 22 Amplifiers 489**
22.1 Gate-Drain Connected Loads ..... 489
22.1.1 Common Source Amplifiers 489
22.1.2 The Source Follower 498
22.1.3 Common Gate Amplifiers 500
22.2 Current Source Loads ..... 500
22.2.1 The Cascode Connection 506
22.2.2 The Push-Pull Amplifier 510
22.3 Noise and Distortion in Amplifiers ..... 512
22.3.1 Modeling Amplifier Noise 515
22.4 A Class AB Amplifier ..... 517
**Chapter 23 Feedback Amplifiers 525**
23.1 The Feedback Equation ..... 526
23.2 Properties of Negative Feedback on Amplifier Design ..... 527
23.2.1 Gain Desensitivity 527
23.2.2 Bandwidth Extension 528
23.2.3 Reduction in Nonlinear Distortion 529
23.2.4 Input and Output Impedance Control 530
23.3 Recognizing Feedback Topologies ..... 532
23.3.1 Input Mixing 533
23.3.2 Output Sampling 533
23.3.3 The Feedback Network 534
23.3.4 Calculating Open-Loop Parameters 537
23.3.5 Calculating Closed-Loop Parameters 539
23.4 The Voltage Amp (Series-Shunt Feedback) ..... 540
23.5 The Transimpedance Amp (Shunt-Shunt Feedback) ..... 547
23.5.1 Simple Feedback Using a Gate-Drain Resistor 555
23.6 The Transconductance Amp (Series-Series Feedback) ..... 558
23.7 The Current Amplifier (Shunt-Series Feedback) ..... 562

**Microchip Ex. 1055, p. 16**
**Microchip v. Aptiv**
**IPR2024-00494**

23.8 Stability ..... 564
23.8.1 The Return Ratio 568
**Chapter 24 Differential Amplifiers 579**
24.1 The Source Coupled Pair ..... 579
24.1.1 Current Source Load 582
24.1.2 Common-Mode Rejection Ratio 591
24.1.3 Noise 592
24.1.4 Matching Considerations 594
24.2 The Source Cross-Coupled Pair ..... 596
24.2.1 Current Source Load 599
24.3 Cascode Loads ..... 602
24.4 Wide-Swing Differential Amplifiers ..... 605
24.4.1 Current Differential Amplifier 607
24.4.2 Constant Transconductance Diff-Amp 608
**Chapter 25 Operational Amplifiers 617**
25.1 Basic CMOS Op-Amp Design ..... 618
25.1.1 Characterizing the Op-Amp 630
25.1.2 Compensating the Op-Amp Without Buffer 634
25.1.3 The Cascode Input Op-Amp 636
25.2 Operational Transconductance Amplifiers ..... 637
25.2.1 Wide-Swing OTA 642
25.2.2 The Folded-Cascode OTA 654
25.3 The Differential Output Op-Amp ..... 664
25.3.1 Fully Differential Folded-Cascode OTA 666
25.3.2 Gain Enhancement 674

**Part IV Mixed-Signal Circuits 683**

**Chapter 26 Nonlinear Analog Circuits 685**
26.1 Basic CMOS Comparator Design ..... 685
26.1.1 Characterizing the Comparator 691
26.2 Adaptive Biasing ..... 699
26.3 Analog Multipliers ..... 704

Microchip Ex. 1055, p. 17
Microchip v. Aptiv
IPR2024-00494

26.3.1 The Multiplying Quad 705
26.3.2 Level Shifting 710
26.3.3 Multiplier Design Using Squaring Circuits 715
**Chapter 27 Dynamic Analog Circuits 719**
27.1 The MOSFET Switch ..... 719
27.2 Switched-Capacitor Circuits ..... 728
27.2.1 Switched-Capacitor Integrator 731
27.3 Circuits ..... 745
**Chapter 28 Data Converter Fundamentals 755**
28.1 Analog Versus Discrete Time Signals ..... 755
28.2 Converting Analog Signals to Digital Signals ..... 756
28.3 Sample-and-Hold (S/H) Characteristics ..... 759
28.4 Digital-to-Analog Converter (DAC) Specifications ..... 762
28.5 Analog-to-Digital Converter (ADC) Specifications ..... 772
28.6 Mixed-Signal Layout Issues ..... 783
**Chapter 29 Data Converter Architectures 791**
29.1 DAC Architectures ..... 791
29.1.1 Digital Input Code 791
29.1.2 Resistor String 792
29.1.3 R-2R Ladder Networks 797
29.1.4 Current Steering 800
29.1.5 Charge Scaling DACs 805
29.1.6 Cyclic DAC 810
29.1.7 Pipeline DAC 811
29.2 ADC Architectures ..... 813
29.2.1 Flash 813
29.2.2 The Two-Step Flash ADC 818
29.2.3 The Pipeline ADC 823
29.2.4 Integrating ADCs 827
29.2.5 The Successive Approximation ADC 832
29.2.6 The Oversampling ADC 837
**Appendix A Orbit's CN20 Process 855**
A.1 Process Specifications ..... 857

**Microchip Ex. 1055, p. 18**
**Microchip v. Aptiv**
**IPR2024-00494**

A.1.1 Electrical Specifications 857
A.1.2 N-Channel SPICE Models 859
A.1.3 P-Channel SPICE Models 861
A.2 Hand Calculations ..... 862
A.2.1 The N-channel MOSFET Equations 862
A.2.2 The P-channel MOSFET Equations 865
A.3 Design Rules ..... 867
**Appendix B MOSIS Scalable Design Rules 873**
**Appendix C HP's CMOS14TB 885**
**Index 893**
**About the Authors 903**
**LASI Software 904**

**Microchip Ex. 1055, p. 19**
**Microchip v. Aptiv**
**IPR2024-00494**

# Preface

Over the last ten years the electronics industry has exploded. A recent report by the Semiconductor Industry Association (SIA) [1] proclaimed that in 1995 alone, world chip revenues increased by 41.7 percent and for the past five years the growth had been exponential. By the year 1999, the report estimates that world chip sales will surpass $234.5 billion, up from $154 billion in 1996. The largest portion of total worldwide sales is dominated by the MOS market. Composed primarily of memory, micro and logic sales, the total combined MOS revenue contributed approximately 75 percent of total world-wide sales ($114.2 billion), illustrating the strength of CMOS technology. The percentage of MOS sales relative to all chip revenues is expected to remain constant through 1999, when MOS sales will total $178 billion.

CMOS technology continues to mature, with minimum feature sizes now approaching 0.1 μm. Texas Instruments recently announced a 0.18 μm process [2] in which the equivalent of 20 high-performance microprocessors could exist on the same substrate, with a transistor density of 125 million transistors. This high density allows for true system-level integration on a chip, with digital signal processors, microprocessors or microcontroller cores, memory, analog or mixed-signal functions all residing on the same die.

As educators we are often asked by our students, "Isn't analog dead? I thought everything was going digital!" How untrue! The prediction of the future demise of analog electronics has been around since the mid-1970s. According to the SIA report [1], the revenues generated by analog products closely parallel the MOS logic market and achieved a 22.5 percent increase in 1995. The analog market expects to reach $18.2 billion in 1996 (a 9.5 percent increase) with double-digit growth projected for the next three years. In 1999, the total revenues generated by analog sales is forecasted to peak at $26.6 billion (11.3 percent of total chip sales!). However, while there is still

Microchip Ex. 1055, p. 20
Microchip v. Aptiv
IPR2024-00494

demand for analog designers, their role is definitely changing. As was communicated by Paul Gray in [3], the days of pure analog design are over, meaning that very few systems remain purely analog. More and more systems are integrated, with increased functionality being performed in the digital domain. He goes on to state that the analog designer should become broad-based, with analog transistor-level design as the core skill. This means that the analog designer should also

- Have a good understanding of digital very large scale integration (VLSI) and be competent at using the latest computer-aided design (CAD) tools.
- know how to apply digital signal processing (DSP), analog signal processing (ASP), and filtering concepts to system-level design.
- possess insight into system implications of component-level performance.

For example, DSP and transistor-level analog design skills are needed for oversampling applications such as data converters, filtering, and a host of relatively new circuit topologies based on sigma-delta modulation. Being able to design both analog and digital circuits, as well as understand the interactions between the two domains, will provide an added dimension to a designer's portfolio that is difficult to match. Analog designers are in demand more than ever, simply because the end limitations of digital electronics need to be examined under the "analog" microscope to fully understand the mechanisms that are occurring. Therefore, this text attempts to combine digital and analog IC design in one complete reference.

Layout is the process of physically defining the layers that compose an integrated circuit. Typically, layouts are constructed using a computer-aided design program. CAD companies such as Mentor Graphics, Synopsis, and Cadence specialize in providing extremely powerful CAD software for the entire integrated circuit (IC) design process, including design, synthesis, simulation, and layout tools within an integrated framework. These workstation-based software tools can literally cost millions of dollars, but provide convenient and powerful features found nowhere else. CAD tools also exist for the PC. Tanner Tool's L-Edit provides a complete IC design CAD program for the personal computer. The program discussed in this book, LAyout System for Individuals (LASI) (pronounced "LAZY"), also provides the student with the ability to lay out ICs on a PC and includes design rule checking and design verification capability. It is distributed as shareware, free for educational purposes.

With decreases in feature size come added complexities in the design. Layouts must now be considered heavily in the design process as matching and parasitic effects become the limiting factors in many precision and high-speed applications. The more the designer knows about the process with respect to layout and modeling, the more performance the engineer can "squeeze" out the design. However, performance is not the only reason to consider the layout. The economic impact of IC layouts can be detrimental to the circuit's marketing potential. In some cases a 20 percent increase in chip area can reduce the profits of a chip by several hundreds of thousands of dollars. Chip area should be considered as premium real estate. Therefore, much of the first ten

Microchip Ex. 1055, p. 21
Microchip v. Aptiv
IPR2024-00494

chapters of this book is devoted to fundamental layout issues, with other issues presented as the need arises.

Modeling is also a key issue. A simulation is only as accurate as its model. Although the Berkeley Short-channel IGFET Model (BSIM) model has become the industry standard its relatively nonintuitive structure makes hand analysis using BSIM model parameters an intimidating process. To many students (and engineers), the BSIM parameters are nothing more than sets of numbers at the end of their SPICE decks. However, some very useful information can be gleaned from the BSIM model which helps make the hand analysis more closely resemble the simulated result. Chapter 6 provides a great deal of information that relates the BSIM model to first-order hand-analysis equations.

A successful CMOS integrated circuit design engineer has knowledge in the areas of device operation, circuit design, layout, and simulation. Students learning CMOS IC design should be trained at a fundamental level in these areas. In the past, courses on CMOS integrated circuits dealt mainly with circuit design or analysis. Little to no time was spent on layout of the integrated circuits. This may have been justified. It is difficult to find a reason to lay out an entire chip and then not have the chip fabricated. However, through the use of the MOSIS[1] program, students can submit their chip designs for fabrication through one of the MOSIS contracted vendors. In approximately ten weeks the chips are returned to the university for evaluation. The MOSIS program is an outstanding way of introducing students to the design of ICs.

Although many texts [4-32] are available covering some aspects of CMOS analog or digital circuit design, none integrates the coverage of both topics with layout and includes layout software as is done in this text. Our focus, when writing this text, was on the fundamentals of custom CMOS integrated circuit design. It was our goal that a student who studies and masters the material in this text will possess the fundamental skills needed to design high-performance analog and digital CMOS circuits and have the basic understanding and problem-solving skills needed to enhance the performance of an IC or to determine why an IC doesn't function as simulated.

*Use of This Text*

This text can be used for two courses. Both courses can be offered at the senior/first-year graduate level. The first course concentrates on the physical design of CMOS digital integrated circuits with prerequisites of junior level Electronics I and a course on digital logic design. A possible semester course outline is as follows.

| | |
|---|---|
| Week 1 | Chs. 1 & 2, introduction, course requirements, layout and SPICE demonstrations, the n-well, sheet resistance. |
| Week 2 | Chs. 2 & 3, the n-well, pn junction, capacitance, resistance, delay through the well, introduction to the metal layers. |

[1] MOSIS - MOS Implementation System through the Information Sciences Institute at the University of Southern California, (310) 822-1511 or http://www.mosis.org

Microchip Ex. 1055, p. 22
Microchip v. Aptiv
IPR2024-00494

| | |
|---|---|
| Week 3 | Chs. 3 &4, the metal layers, parasitics, electomigration, layout of the padframe, active/poly layers, layout of the MOSFET and standard frame. |
| Week 4 | Ch. 5, MOSFET operation |
| Week 5 | Chs. 5 & 6, completion of MOSFET operation, discussion of modeling using the BSIM model. |
| Week 6 | Chs. 6 & 7, completion of BSIM model, layout of a capacitor, MOS temp dependence. |
| Week 7 | Chs. 10 & 11, digital models and the inverter. |
| Week 8 | Ch. 11, the inverter, switching point voltage and switching times, layout, latch-up, and design. |
| Week 9 | Ch. 12, static logic gates, switching point voltages, speed, and layout. |
| Week 10 | Chs. 13 & 14, the transmission gate, flip-flops, and dynamic logic gates. |
| Week 11 | Chs. 15 & 16, VLSI layout and BiCMOS logic. |
| Week 12 | Ch. 17, memory circuits, basic memory cells, and organization. |
| Week 13 | Ch. 18, special-purpose digital circuits. |
| Week 14 | Ch. 19, introduction to digital phase locked loops, phase detectors, VCOs. |
| Week 15 | Ch. 19, digital PLL design. |

The second course concentrates on CMOS analog circuit design. A possible semester course outline is as follows.

| | |
|---|---|
| Weeks 1 & 2 | Review of Chs. 1-6. |
| Week 3 | Chs. 7, CMOS passive elements, noise characteristics. |
| Week 4 | Ch. 9, analog MOSFET models. |
| Week 5 | Ch. 20, current sources and sinks. |
| Week 6 | Ch. 21, references. |
| Week 7 | Ch. 22, amplifiers. |
| Week 8 | Ch. 23, selected topics in feedback amplifier design. |
| Week 9 | Ch. 24, differential amplifiers. |
| Weeks 10-12 | Ch. 25, operational amplifiers. |
| Week 13 | Ch. 26, nonlinear analog circuits. |
| Week 14 | Ch. 27, dynamic analog circuits. |

**Microchip Ex. 1055, p. 23**
**Microchip v. Aptiv**
**IPR2024-00494**

Week 15 Chs. 28 &29, selected topics in data converter design.

This text can also be used as an accompanying text in a VLSI systems course that focuses on the implementation of systems rather than circuits. Use of the text in this manner is benefited by inclusion of the LASI layout software.

## REFERENCES

[1] *Revised Forecast for World Chip Market Shows Growth of 6.7% in 1996, 19% by 1999*, Semiconductor Forcast Summary 1995-1998, Semiconductor Industry Association.

[2] "New TI Technology Doubles Transistor Density," T*exas Instruments Integration Newsletter*, Vol. 13, No. 5, July 1995.

[3] P. Gray, "Possible Analog IC Scenarios for the 90's," http://kabuki. eecs.berkeley.edu/slides.html

*Digital Circuits and VLSI System Design*

[4] C. Mead and L. Conway, *Introduction to VLSI Systems*, Addison-Wesley, 1980.

[5] Glasser and Dopperpuhl, *The Design and Analysis of VLSI Circuits*, Addison Wesley, 1985.

[6] M. Annaratone, *Digital CMOS Circuit Design*, Kluwer, 1986.

[7] A. Mukherjee, *Introduction to NMOS and CMOS VLSI Systems Design*, Prentice-Hall Publishers, 1986. ISBN 0-13-490947-X

[8] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits*, McGraw-Hill , 2nd ed., 1988. ISBN 0 - 07 - 029158 - 6.

[9] M. Shoji, *CMOS Digital Circuit Technology*, Prentice-Hall, 1988. ISBN 0-13-138850-9.

[10] J. P. Uyemura, *Fundamentals of MOS Digital Integrated Circuits*, Addison-Wesley, 1988. ISBN 0-201-13318-0.

[11] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI - Design Techniques for Analog and Digital Circuits*, McGraw-Hill, 1990. ISBN 0-07-023253-9.

[12] J. Y. Chen, *CMOS Devices and Technology for VLSI*, Prentice-Hall, 1990. ISBN 0-13-138082-6.

[13] Fabricius, *Introduction to VLSI Design*, McGraw-Hill, 1990.

[14] M. I. Elmasry, *Digital MOS Integrated Circuits II*, IEEE Press, 1992. ISBN 0-87942-275-0.

[15] N.H.E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Addison Wesley, 2nd ed., 1993. ISBN 0-201-53376-6.

**Microchip Ex. 1055, p. 24**
**Microchip v. Aptiv**
**IPR2024-00494**

[16] J. P. Uyemura, *Circuit Design for Digital CMOS VLSI*, Kluwer, 1992.

[17] D. A. Pucknell and K. Eshraghian, *Basic VLSI Design*, 3rd ed., Prentice Hall Publishers, 1994. ISBN 0-13-079153-9

[18] D. Pucknell, *Basic VLSI Design: Systems and Circuits*, 3rd ed., Prentice Hall, 1994. ISBN: 0-13-079153-9

[19] W. Wolf, *Modern VLSI Design: A Systems Approach*, Prentice Hall, 1994. ISBN: 0-13-588377-6

[20] S. Kang and Y. Leblebici, *CMOS Digital Integrated Circuits - Analysis and Design*, McGraw-Hill, 1996. ISBN 0-07-038046-5.

[21] K. Gopalan, *Introduction to Digital Microelectronic Circuits*, Irwin, 1996. ISBN 0-256-12089-7.

[22] J. M. Rabaey, *Digital Integrated Circuits - A Design Perspective*, Prentice Hall, 1996, ISBN 0-13-178609-1.

*Analog Circuits*

[23] A. B. Grebene, *Bipolar and MOS Analog Integrated Circuit Design*, John-Wiley, 1984. ISBN 0-471-08529-4

[24] R. Gregorian and G. C. Temes, *Analog MOS Integrated Circuits for Signal Processing*, John Wiley, 1986. ISBN 0-471-09797-7.

[25] P. E. Allen and D. R. Holberg, *CMOS Analog Circuit Design*, Holt, Rinehart and Winston, 1987. ISBN 0-03-006587-9.

[26] P. R. Gray, B. A. Wooley and R. W. Broderson, *Analog MOS Integrated Circuits II*, IEEE Press. ISBN 0-87942-246-7.

[27] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI - Design Techniques for Analog and Digital Circuits*, McGraw-Hill, 1990. ISBN 0-07-023253-9.

[28] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 3rd ed., John Wiley, Inc., 1993. ISBN 0-471-57495-3.

[29] M. Ismail and T. Fiez, *Analog VLSI - Signal and Information Processing*, McGraw-Hill, Inc. 1994. ISBN 0-07-032386-0.

[30] K. R. Laker and W. Sansen, *Design of Analog Integrated Circuits and Systems*, McGraw-Hill, 1994. ISBN 0-07-036060-X.

[31] G.A.S. Machado, *Low-Power HF Microelectronics a unified approach*, IEE, 1996. ISBN 0-85296-874-4.

[32] D. Johns and K. Martin, *Analog Integrated Circuit Design*, John Wiley, 1997. ISBN 0-471-14448-7.

**Microchip Ex. 1055, p. 25**
**Microchip v. Aptiv**
**IPR2024-00494**

# Acknowledgments

We would like to thank the reviewers, contributors, and colleagues who helped make this book possible; Dr. Phil Allen, Savoula Amanatidis, Ben Ba, Jan Bissey, Dr. William Black, Jeff Bruce, Alan Buchholz, Dr. Joseph Cavallaro, Brian P. Lum Shue Chan, Lisa Dayne, Dr. Ian Galton, Dr. Randall Geiger, John Griffin, Wes Hansford, Aaron Huntsinger, Dr. Bruce Johnson, David Kao, Dr. Joe Karniewicz, Brent Keeth, Dr. William Kuhn, Wen Li, Dr. H. Alan Mantooth, Dr. Richard Marks, Dean Moriarty, Dr. Ken Noren, Dr. Adrian Ong, Dr. James Rochelle, Dr. Terry Sculley, Dr. Stuart Tewksbury, Dr. Don Thelen, Dr. Vance Tyree, Scott Ward, Veronica Wilson, Dr. Jeff Wu, and Dr. Kwang S. Yoon.

The authors would also like to thank Orbit Semiconductor, MOSIS and HP for allowing us to include their process information and design rules with the text.

*R. Jacob Baker*

*Harry W. Li*

*David E. Boyce*

**Microchip Ex. 1055, p. 26**
**Microchip v. Aptiv**
**IPR2024-00494**

Part

I

# CMOS Fundamentals

Microchip Ex. 1055, p. 27
Microchip v. Aptiv
IPR2024-00494

Chapter

1

# Introduction

This chapter discusses the CMOS (complementary metal oxide semiconductor) integrated circuit (IC) design process, how to set up the LASI (LAyout System for Individuals) layout software, and fabrication of CMOS integrated circuits through MOSIS (MOS Implementation Service).

## 1.1 The CMOS IC Design Process

The CMOS circuit design process consists of defining circuit inputs and outputs, hand calculations, circuit simulations, layout of the circuit, simulations including parasitics, reevaluation of the circuit inputs and outputs, fabrication, and testing. A flowchart of this process is shown in Fig. 1.1. The circuit specifications are rarely set in concrete; that is, they can change as the project matures. This can be the result of tradeoffs made between cost and performance, changes in the marketability of the chip, or simply changes in the customer's needs. In almost all cases, major changes after the chip has gone into production are not possible.

This text concentrates on custom IC design. A custom-designed chip is often called an ASIC (application-specific integrated circuit). Other (noncustom) methods of designing chips, including field-programmable-gate-arrays (FPGAs) and standard cell libraries, are used when low volume and quick-design turnaround are important. Most chips that are mass produced, including microprocessors and memory, are examples of chips that are custom designed.

The task of laying out the IC is often given to a draftsman. However, it is extremely important that the engineer be able to lay out a chip (and direct the draftsman on how to lay the chip out) and understand the parasitics involved in the layout. Parasitics are the stray capacitances, inductances, pn junctions, and bipolar transistors, with the associated problems (breakdown, stored charge, latch-up, etc.). A fundamental understanding of these problems is important in precision/high-speed design.

Microchip Ex. 1055, p. 28
Microchip v. Aptiv
IPR2024-00494

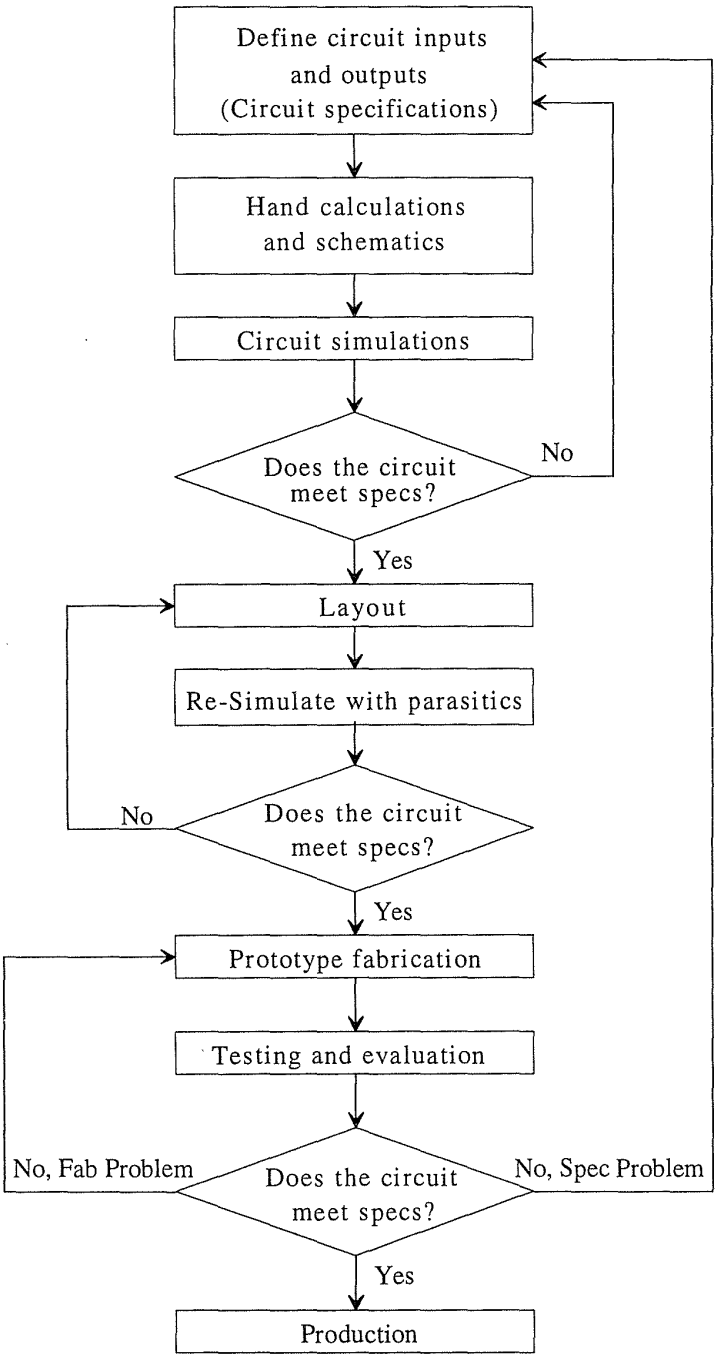



**Figure 1.1** Flowchart for the CMOS IC design process.

Microchip Ex. 1055, p. 29
Microchip v. Aptiv
IPR2024-00494

### 1.1.1 Fabrication

CMOS integrated circuits are fabricated on thin circular slices of silicon called wafers. Each wafer contains several individual chips or "die" (Fig. 1.2). For production purposes each die on a wafer is usually identical. Added to the wafer are test structures and process monitor plugs (sections of the wafer used to monitor process parameters).

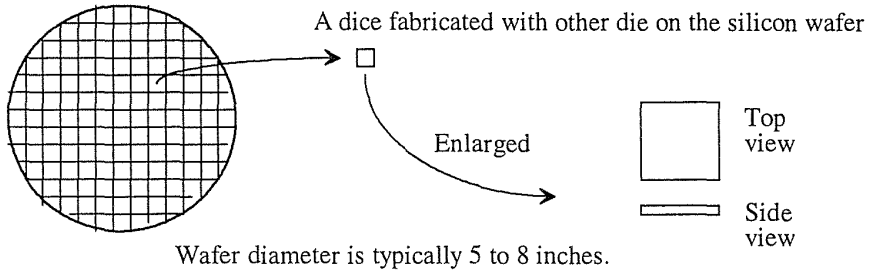



**Figure 1.2** CMOS integrated circuits are fabricated on and in a silicon wafer.

The ICs we design and lay out using LASI can be fabricated through MOSIS [1] on what is called a multiproject wafer; that is, the wafer consists of chip designs of varying sizes from different sources (educational, private, government, etc.). MOSIS combines multiple chips on a wafer to split the fab cost among several designs to keep the cost low. MOSIS subcontracts the fabrication of the chip designs out to one of many commercial manufacturers, including Orbit and HP.

The view we see when laying out an IC is always a top view of the die. When laying out the chip, we draw boxes or polygons on differing layers indicating how to assemble the circuit. We may specify a box on layer 1 (n-well) from the coordinates (0,0) to the coordinates (10,10). The coordinates of this box and other shapes defining the circuitry are specified in a binary file using Calma Stream Format (abbreviated CSF, GDSII, or simply GDS). This file describes the completed chip design. When using the LASI program, TransportabLe LASI Cell files (TLC files) are used to store the design information. When the design is finished, the TLC files can be converted to a GDS file and sent via the Internet to MOSIS for fabrication.

## 1.2 Using the LASI Program

The LASI program discussed in this text is a powerful CAD (Computer-aided design) package used in the design of integrated circuits. This section discusses the installation and operation of LASI in the DOS environment.

*Installing LASI*

To install LASI, place the LASI disk accompanying this text or downloaded from the Internet into the a: floppy. From the DOS prompt type:

Microchip Ex. 1055, p. 30
Microchip v. Aptiv
IPR2024-00494

C:\>a:2install

Answer the installation instructions, selecting "yes" for installing the setups and modifying the autoexec.bat file, and run the tutorial supplied with the program. When finished, reboot the computer. Note that a DOS mouse driver must be installed if not running under Windows.

After the installation is complete, your hard disk should have a directory C:\LASI5 containing all executable files and directories C:\LASI5\CN20, C:\LASI5\MOSIS, and C:\LASI5\2UCHIP containing the setups for the CN20 process, MOSIS and MOSIS 2 μm processes, respectively. The autoexec.bat file should have had C:\LASI5 added to the beginning of the PATH statement. To verify that the path is set correctly, type "set" at the DOS prompt (after rebooting the computer). If the path is not set correctly, use the **edit autoexec.bat** command to manually add C:\LASI5 to the PATH statement. If your autoexec.bat file does not have a PATH statement, simply add PATH C:\LASI5; at the beginning of the file. The PATH statement permits LASI or any LASI program to run from any directory on the hard disk.

*Drawing Directories*

Under the C:\LASI5 directory should be a directory \TUTOR containing the example of a bipolar op-amp supplied with LASI. *All chip designs should reside in a directory other than the* LASI5 *directory.* The \LASI5 directory is used only for the executable programs in the layout system. The directory of a chip design can be a subdirectory of \LASI5, similar to \TUTOR above, or a directory somewhere else on the hard disk.

Throughout the book we will be using Orbit's CN20 CMOS process. Change directories into the \LASI5\CN20 directory and verify that the setups were successfully copied into the directory.

*Starting LASI*

We are now ready to run the LASI program. At the DOS prompt, from within the drawing directory, type "lasi" or

C:\LASI5\CN20>lasi

The screen shown in Fig. 1.3 should appear (in color).

*Help*

The LASI layout system comes with a complete on-line manual. To use the manual at any time, while LASI is running, press F1. Pressing F1 while the mouse cursor is over a command button will give information related to the command button. The manual can also be accessed without LASI running by typing, within the directory shown, the following:

C:\LASI5\>lhi

**Microchip Ex. 1055, p. 31**
**Microchip v. Aptiv**
**IPR2024-00494**

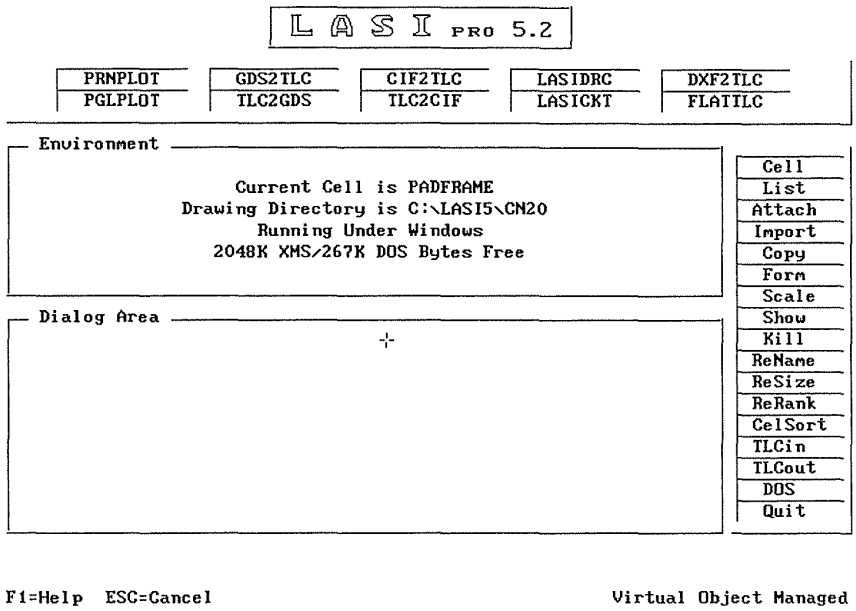



**Figure 1.3** LASI startup screen.

### 1.2.1 Cells in LASI

Complex IC designs are made from simpler objects called cells. A cell might be a logic gate or an op-amp. To show a listing of the cells in this drawing directory, select **List** from the commands. LASI will show that the drawing directory is empty. Click, with the mouse, anywhere on the screen to return to the previous screen (the system screen).

*TLC Files*

Cells in LASI are backed up using transportable LASI cell files (or TLC files for short). The setup files that were copied into the CN20 directory during installation contain several TLC files. By selecting the **TLCin** command (running LASI from the C:\LASI5\CN20 directory), we can convert these TLC files into a binary format (files with the *.BP5 and *.CL5 extensions) for use with the LASI program.

Pressing the **TLCin** command button will begin this process (do this now). Specifying "." for the source path indicates the current drawing directory. Specifying "*" for the cell names will convert all TLC files in the directory into *.BP5 and *.CL5 format. The cells in the LASI drawing directory are backed up, normally on a floppy, using the **TLCout** command button. In Fig. 1.4, pressing "y" to "All Correct & Ready," and then "y" to overwrite any existing cells in the drawing directory (at this point, none should exist) will cause several cells to be changed into LASI binary format. Using the **List** command button and clicking on the cell "rule1" puts us into the drawing display (cell mode). Using the **Fit** command button will center the contents of the cell in the window. The result is seen in Fig. 1.5.

Microchip Ex. 1055, p. 32
Microchip v. Aptiv
IPR2024-00494

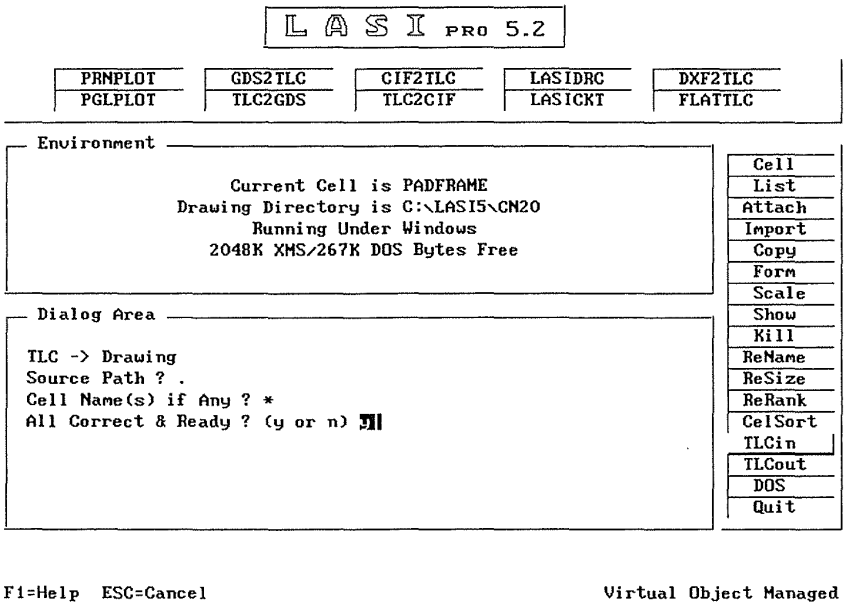



**Figure 1.4** Using the **TLCin** command.

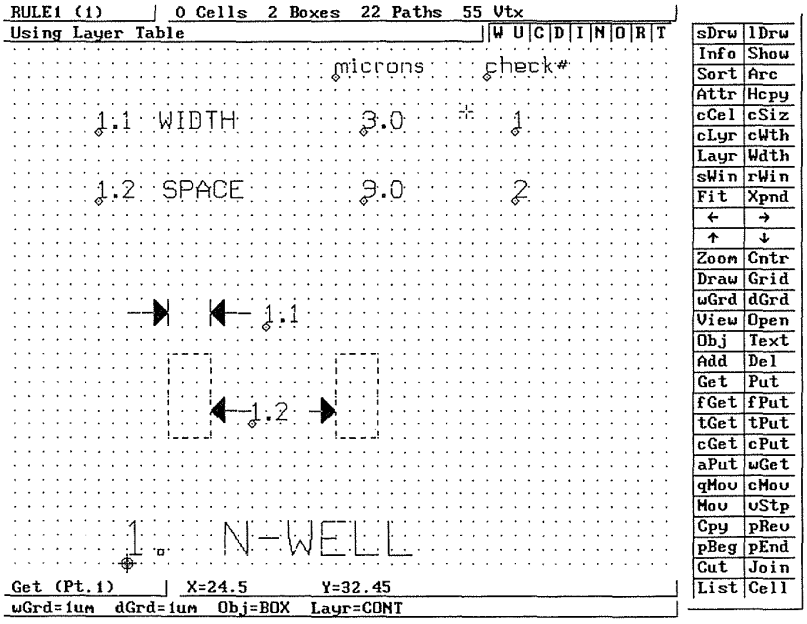



**Figure 1.5** LASI display after opening the file containing rule 1.

Microchip Ex. 1055, p. 33
Microchip v. Aptiv
IPR2024-00494

*Toggling Menus*

Notice, in Fig. 1.5, that clicking the RIGHT mouse button causes some of the menu items to change or toggle between different commands. Again, a menu item can be selected using the LEFT mouse button. Return to the System menu by selecting the **Sys** command in the bottom left corner of the command buttons. If the **Sys** command is not shown, click the RIGHT mouse button once.

*Creating a Cell*

To begin the layout of a new cell, the **Cell** command is used. The cell is assigned a name, for example, AND, and a rank. Since the AND gate is a basic building block, we will assign a rank of 1 (the lowest rank). If a cell is created that uses the AND gate cell, then the new cell will be assigned a rank of 2 or higher. That is, a cell with a rank of 2 can contain cells with a rank of 1. If the cell has a rank of 5, then it can contain any cell with a rank of 4 or lower. Another analogy is to consider a chip, a printed circuit board, and a computer. A chip can be put on a circuit board, and the circuit board can be put into a computer. Using the ranking analogy, the chip has a rank of 1, the circuit board has a rank of 2, and the computer has a rank of 3. Therefore, the computer with rank 3 cannot be put into the chip with rank 1.

Create a cell called "test" for experimentation by selecting the command **Cell** and entering the name "test" with a rank 1. The bare test cell is shown in Fig. 1.6. Notice in the top left corner that the cell name and rank are displayed.

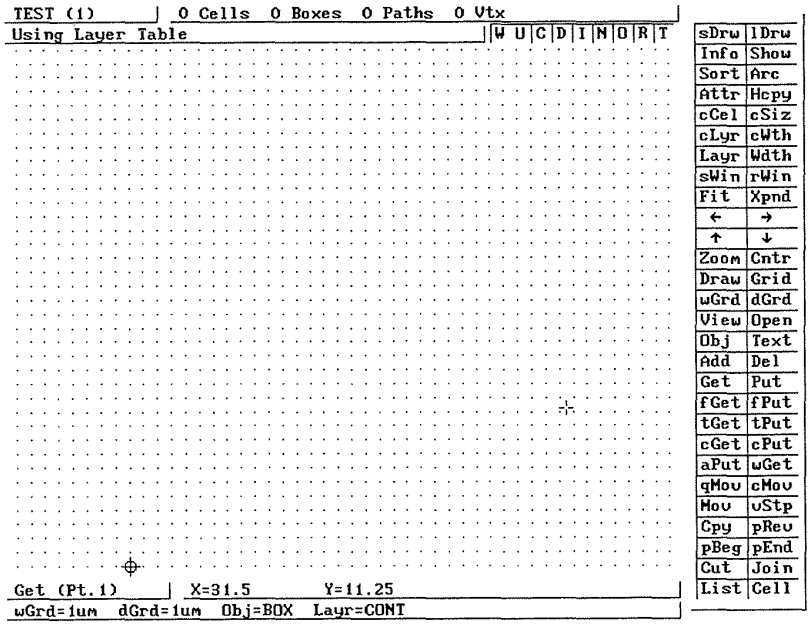



**Figure 1.6** LASI display using the layer table, CONT selected as the current layer, Box selected as the current object, and grids are set at 1 μm.

Microchip Ex. 1055, p. 34
Microchip v. Aptiv
IPR2024-00494

### 1.2.2 Navigating LASI

Notice in Fig. 1.6 that the cross-hairs show the location of the origin. Pressing **R** in the top right corner (or typing **r** on the keyboard) of the display turns this reference designator on or off. Now hit the **Draw** command. This command redraws the screen with (or without) the reference cross-hairs showing. Press **r** so that the reference designator is showing. Placing the mouse cursor over the origin causes the distance indicator at the bottom of the display to read (0,0). The distance indicator shows the distance away from the origin. If the indicator is showing distances of $10^6$ or larger ($10^6$ μm is 1 meter), you are viewing the entire drawing universe. Selecting the **Fit** command with nothing drawn could easily result in this situation. To get back to a display that shows tens of microns, simply select the **Zoom** command and then double click, with the LEFT mouse button, on the reference designator. If you cannot find the reference designator, select **Fit,** keeping in mind that **R** in the top right of the display should be highlighted. One other useful command is the **Orig** command. This command allows the user to set the origin at any location in the drawing display.

*Grid*

The grid can be turned on and off by clicking on the **Grid** command button. The grid will not display when "zoomed out."

*Cursor*

The cursor can be changed between a small-cross and full-screen cross-hairs by pressing the **Tab** button on the keyboard.

*Measurements*

By pressing **z** on the keyboard, a zero reference point is established independent of the origin. When the **spacebar** is pressed, the distance between the mouse cursor and the zero reference point is displayed. Pressing **w** causes the mouse to snap to the working grid.

### 1.2.3 Adding Objects

Select the **Layr**[1] command to display the layer table used in this process. From the layer table with the LEFT mouse button, select layer 1, the n-well. Notice that the selected layer is displayed at the bottom of the screen. Now select **Obj** or the object command. The object command will accept three arguments, "b" for box, "p" for path (or polygon), or the name of a cell if we want to add a lower ranking cell to the current cell. Type "b" for box and hit Enter. Now the bottom of the drawing display should show that the working and dot grids are 1 μm, the object is a Box, and the layer "nwel." We are now ready to begin drawing the layout.

---

[1] Again, remember that if the command is not showing, press the RIGHT mouse button to toggle between menus.

Microchip Ex. 1055, p. 35
Microchip v. Aptiv
IPR2024-00494

Click on the **Add** command with Obj = BOX and Layr = NWEL. Click the LEFT mouse button in the drawing area over the origin once. Move the mouse toward the top right corner of the display until you get a box similar to the one shown in Fig. 1.7 and click the LEFT mouse button again. Using the **arrow keys** on the keyboard or the command buttons, change the display view. Now select **Fit** and notice that the n-well is centered in the drawing display. The **Xpnd** command will expand the view. Notice how the viewing area increases. Use the **Zoom** command to zoom in on the corner of the n-well box by drawing an imaginary box around one of the corners of the n-well box. When drawing, it is often useful to show a grid in the drawing area. The commands **wGrd** and **dGrd** are used to change between the working (the grid the cursor snaps to) and dot (the one you see) grids. For the setups included with this book, both grids are set to 1 μm. Using the **Set** command, the user can set the number of working and dot grids as well as the grid spacing. It is not recommended that any of the settings be changed until the user becomes very familiar with LASI.

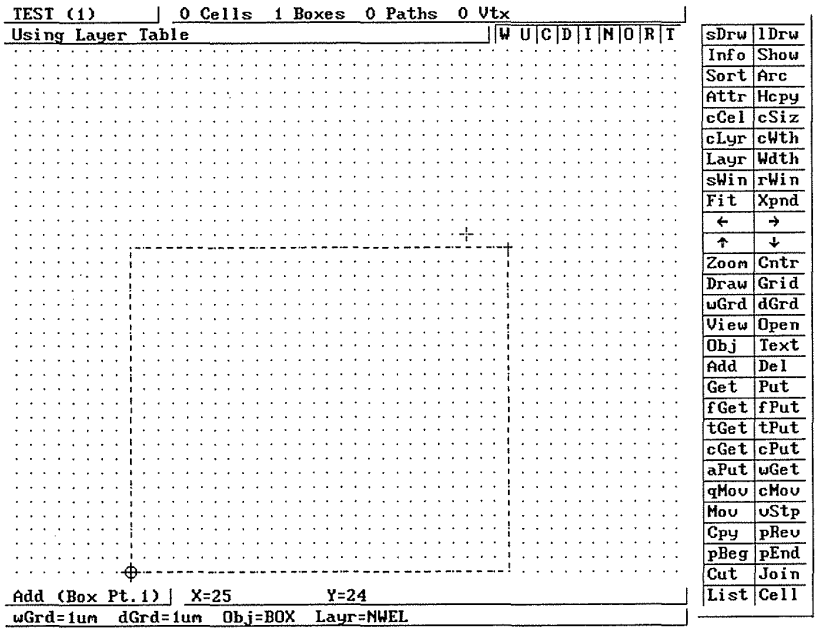



**Figure 1.7** Drawing a box in LASI.

### 1.2.4 Editing Objects

Only one basic operation, moving, can be performed on a Box after it has been added to a cell. The entire box or any of the four sides can be moved. This operation consists of *getting* the object or side to be moved and performing the *move*, and then *putting* the object back (de-selecting the object).

Microchip Ex. 1055, p. 36
Microchip v. Aptiv
IPR2024-00494

Using the previously generated n-well layout as an example, use the **Get** command and select the right side of the n-well box using the mouse. This is done by clicking the LEFT mouse button on the left of the side and then again on the right, ensuring the box drawn by the mouse intersects the side. Objects that are selected using **Get** should be highlighted. Next, select the **Mov** command. Using the LEFT mouse button, click once somewhere in the display. Move the mouse a small distance to the left and click the LEFT mouse button again. Notice that the highlighted side moved an equal distance to the left of its original position. Use the **Put** command to unselect the highlighted side. A simpler method of unselecting all of the active elements is to use the **aPut** (all put) command.

Sometimes we may want to get the entire box. The **fGet** (full get) command allows dragging the mouse through a portion of the object to select the entire object. Try selecting the entire box using **fGet** and then unselecting it using **aPut**.

LASI also has the feature that allows the user to execute a set of commands by pressing a single key. This is easily accomplished by adding a line to the form.dbd file. For more information, read the on-line help (by pressing F1).

*Viewing and Editing Specific Layers*

Suppose that it is desired to view only a few layers of a complicated layout. This can be accomplished by selecting the desired layers using the **View** command. The user needs to be warned, however, that if the invisible layers are not made visible again, frustration could abound.

The **Open** command determines which layers can be selected using the **Get** command. This feature allows the user to make certain layers uneditable. In the current drawing, select the **Open** command and unselect the n-well layer. Return to the drawing area by clicking the RIGHT mouse. Attempt to get any part of the n-well box and notice that none of the sides can be highlighted or edited in any fashion. Highlighting the n-well layer under the **Open** command will then allow any editing operations on that layer. This command could also be frustrating if the user forgets that certain layers are unselectable.

### 1.2.5 Placing Cells

Select **Cell** from the command button and make another cell. Call the new cell test2 and assign a rank of 2. Next, using the **Obj** command, use the previously laid out cell (test) as the new object. Using the **Add** command, add the cell, test, to the cell test2 by simply clicking the LEFT mouse button in the drawing area. Repeat this several times until the layout is similar to Fig. 1.8. Remember that if a cell has a rank equal to or greater than the cell you are currently editing, nothing will happen if the cell is **Add**ed to the current one.

Microchip Ex. 1055, p. 37
Microchip v. Aptiv
IPR2024-00494

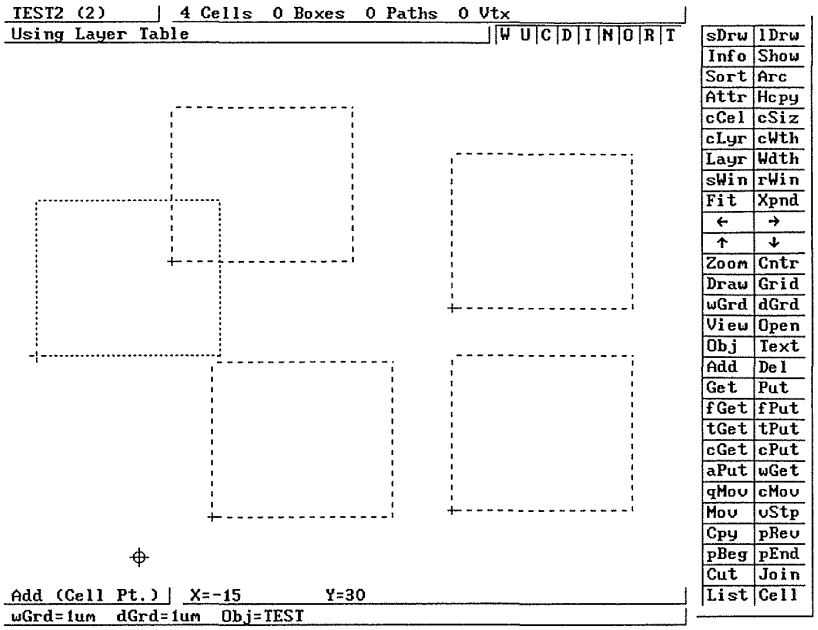



**Figure 1.8** Inserting the cell, test, using the **Add** command.

*Viewing Complex Layouts*

Cells can be drawn as outlines using **Outl** (Fig. 1.9). The mouse is used to enclose the cell you want drawn as an outline. Using the **Full** command draws the entire cell. This becomes helpful when a large number of cells are present and the redraw time is long. Also, cells will be drawn as outlines if the ESC key is pressed when LASI is redrawing the screen or the **Dpth** command is used. The **Dpth** command is used to draw cells as outlines if they are nested deeper than the cells depth setting.

*Moving Cells*

Moving a cell consists of getting the cell using the **cGet** (cell get) command, moving the cell using the **Mov** command, and putting the cell using the **aPut** (all put) or **cPut** (cell put) commands.

*Displaying Cells*

When cells are drawn as outlines, using the **Outl** command, the names, on the outline, can be displayed or hidden by pressing **n** on the keyboard (indicated by **N** in the top right corner of the drawing display). If the cells are not drawn as outlines, pressing **i** on the keyboard draws a dotted line around the cell. This is useful as a reminder of which parts of the layout are cells and which parts are boxes or polygons.

*A Note on Editing*

Only objects that are drawn in the current cell can be edited. The objects resulting from adding a cell to the current (open) cell cannot be edited. Trying to change the size of

Microchip Ex. 1055, p. 38
Microchip v. Aptiv
IPR2024-00494

any of the n-well boxes of Fig. 1.8 from the *test2* cell (the open cell) would result in frustration. To change the size of these boxes, we must first open the *test* cell.

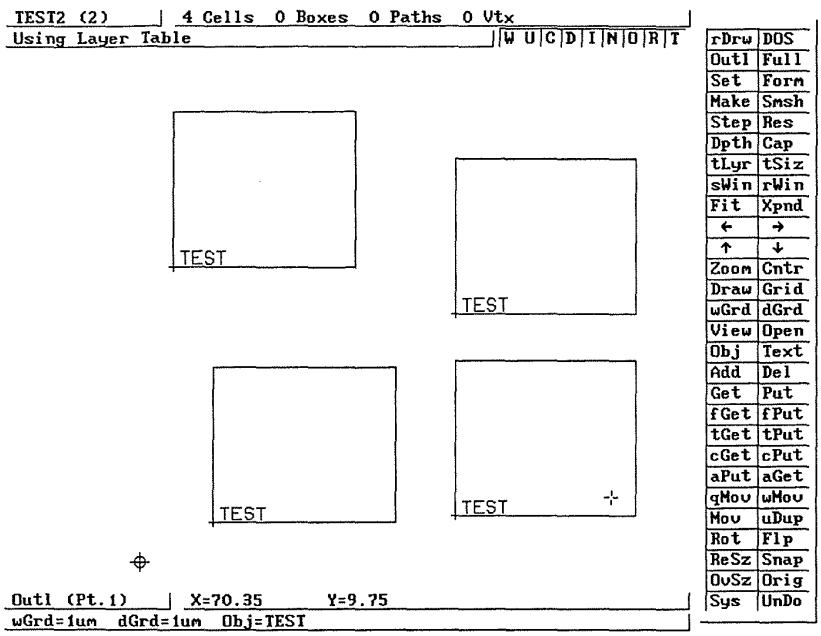



**Figure 1.9** Using the **Outl** command.

*Saving Your Work*

LASI can automatically generate a TLC file in the drawing directory each time the cell mode is entered or exited. This is specified by including the line "misc = autoTLCout" in the form.dbd file in the drawing directory. A backup to floppy should also be made periodically using the **TLCout** command button on the system mode.

*Understanding the UnDo Command*

Using the **UnDo** command with argument "c" restores the cell to the state it was in when it entered the cell mode. It *does not* simply undo the last command executed. This command can cause major loss of information if the cells are not backed up periodically. More information regarding this command can be found in the on-line manual.

### 1.2.6 Common Problems

*After adding an object, the object cannot be seen.*

Check the **View** layers in the drawing display to ensure that the layer is not in hidden mode. The **Draw** command must be used after using **View**.

*Cannot* **Get** *an object.*

Microchip Ex. 1055, p. 39
Microchip v. Aptiv
IPR2024-00494

(1) Check, using the **Open** command, that the layer can be opened (or moved). (2) Verify that the object is not part of another cell. (3) When trying to get an object made using the path or polygon object, make sure the cursor encompasses a vertex.

*The menu doesn't redraw when the* **Layr** *button is pressed.*

The layer table is not being used. Press CNTRL-ENTER so that the top left corner of the display shows "Using Layer Table."

*Cells are drawn as outlines, or the perimeter of a cell has a dashed line.*

(1) Use the **Full** command to show the contents of a cell. (2) Use the **Dpth** command to limit the depth of the cells shown. Increasing the depth to the rank of the cell shows all cells. (3) Press **i** on the keyboard to force an outline to be drawn around a cell. This is indicated by the **I** in the top right corner of the drawing display.

*The* **Fit** *command causes the drawing window to expand much larger than the current cell.*

There is an unknown object someplace in the cell. Use the **fGet** command to get any objects outside the main cell area. Use the **Del** command to delete the unknown object.

*LASI or any other LASI program that uses XMS memory when run under DOS does not use XMS memory when run under Windows 3.1.*

Use Windows PIF editor to allocate at least 2048 KB of XMS memory in the PIF file for that program.

*The cursor movement is not smooth.*

The cursor may be in the octagonal mode. Press "o" on the keyboard or the button "O" in the top right portion of the display to toggle this mode on and off.

## 1.3 MOSIS

The MOSIS[2] IC fabrication service is available to universities provided they have access to the Internet, software for layout and simulation, and the capability to test the completed designs. The instructor must contact MOSIS and submit a proposal for funding. If the proposal is funded, the university will have an account set up with the dollar amount determined by the number of students in the CMOS course. At the present time, funding from DARPA/NSF[3] for introductory classes is $510 per two students. This is the cost of one "tiny chip" fabricated in a 2.0 µm process measuring 2.2 mm by 2.2 mm edge to edge. A minimum quantity of four chips is supplied with each order. Commercial companies may also fabricate ICs through MOSIS but receive no government funding.

[2] MOSIS - MOS Implementation Service through the Information Sciences Institute at the University of Southern California, (310) 822-1511 or http://www.mosis.org

[3] Defense Advanced Research Projects Agency and the National Science Foundation.

Microchip Ex. 1055, p. 40
Microchip v. Aptiv
IPR2024-00494

After an account has been established by MOSIS and the university has an account number, completed chip designs can be submitted in UUEncoded GDS (UUGDS[4]) format or CIF (CalTech Intermediate Form). This text will only describe the fabrication process using GDS format. For information regarding CIF, contact MOSIS. To translate a TLC file into GDS format, simply select the TLC2GDS command button on the LASI system menu. The screen shown in Fig. 1.10 will appear. Here, the name of the highest ranking cell, that is, the cell to be fabricated, is QCELL.TLC. All cells used in QCELL.TLC will be converted into GDS format and placed in the binary file QCELL.GDS (this is transparent to the user). Of course, the **TLCout** command must be executed prior to calling the TLC2GDS utility. The conversion process is started by selecting "Go" on the screen in Fig. 1.10. Note that text is made using polygons so selecting "Y" to "Make Only Proper Poly" can cause problems. For example, text on the metal1 layer will be changed into polygons with the result of changing the metal1 mask.

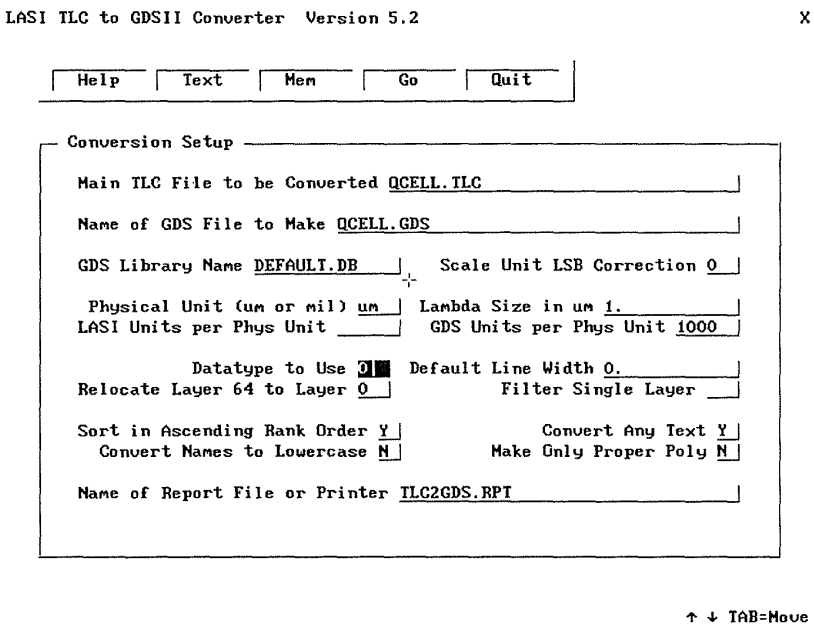



**Figure 1.10** Converting TLC files to GDS format.

To translate the binary GDS file (QCELL.GDS) into an ASCII file (QCELL.UUE) suitable for transmitting to MOSIS, the following command is used:

C:\LASI5\IDAHO> uuen -j qcell.gds qcell.uue

The executable file UUEN.EXE is located in the C:\LASI5 directory. The switch "j" creates a Unix-compatible ASCII file.

[4] UUEncoding is used to change a binary file such as the GDS file generated with the TLC2GDS into an ASCII file for transmission over the Internet. Here UUGDS is a UUEncoded GDS file.

Microchip Ex. 1055, p. 41
Microchip v. Aptiv
IPR2024-00494

The final step, before submission of the file to MOSIS, is to run the checksum program (CKSUM.EXE) located in C:\LASI5 on QCELL.UUE, or

C:\LASI5\IDAHO> cksum qcell.uue

The result is two numbers; the layout-checksum and the byte count. These numbers are used in the submission process discussed below.

*Submission of Chips to MOSIS*

The basic submission of chips to MOSIS consists of requesting an ID for a new project, submitting the project for check, and requesting the project be fabricated. Checking the status of the chips and changing or canceling project parameters can also be performed before the chips are sent out, from MOSIS, for fabrication.

A project ID can be requested from MOSIS by sending an appropriate e-mail address to mosis@mosis.org for each of the chip designs that will be fabricated. The MOSIS command language syntax is used when communicating with MOSIS. When making this request, the user must specify several items, as illustrated in the following example. The project check is used to ensure that the ASCII file containing the UUGDS specifications of your chip does not get corrupted when transmitted over the Internet. If the GDS file is accepted, the users will be notified. The user can then instruct MOSIS, via the Internet, to fabricate the chip.

---

**Example 1.1**

Submit a chip to MOSIS for fabrication using Orbit's 2.0 μm n-well process (CN20). Information on this process is given in Appendix A and is used throughout the text.

*Step 1: Request New Project ID*

Assuming that MOSIS specified an account number of 123-ABC, the password is WINFECT, and the instructor's name is SMITH, note that the first step in submitting a chip for fabrication is to send MOSIS an e-mail requesting a new project ID (the e-mail message is shown below). Again the e-mail address is mosis@mosis.org.

| | |
|---|---|
| REQUEST: | NEW-PROJECT |
| ACCOUNT: | 123-ABC |
| D-NAME: | SMITH |
| D-PASSWORD: | WINFECT |
| P-NAME: | CHIP1 |
| P-PASSWORD: | UNIVER |
| PHONE: | (123) 456-7890 |
| TECHNOLOGY: | FORESIGHT-CN20 |

**Microchip Ex. 1055, p. 42**
**Microchip v. Aptiv**
**IPR2024-00494**

| | |
|---|---|
| SIZE: | 2160 X 2160 |
| PADS: | 40 |
| PACKAGE: | DIP40 |
| QUANTITY: | 4 |
| DESCRIPTION: | HIGH SPEED OP-AMP |
| REQUEST: | END |

The part name and password are defined by the user. It was assumed that the pads were designed by the students and measured 180 μm square; that is, the outline of the pads is 180 μm, while the actual pad is 100 μm on a side (see Ch. 3). The resulting padframe measures 2160 μm by 2160 μm. The "PADS:" line specifies the number of pads on the chip. The description line of the request file is also defined by the user. MOSIS replies to the above request with an ID number such as 876543. The MOSIS system is automated so that precise syntax is required.

*Step 2: GDS File Submission*

To submit the chip for project check, send the message shown below. The final cell that references all the other cells is the "TOP-STRUCTURE" cell. This cell is the chip that will be fabricated.

| | |
|---|---|
| REQUEST: | SUBMIT |
| ID: | 876543 |
| P-PASSWORD: | UNIVER |
| LAYOUT-CHECKSUM: | 123456 1234 |
| LAYOUT-FORMAT: | UUGDS |
| TOP-STRUCTURE: | QCELL |
| LAYOUT: | |
| Insert UUGDS file here (qcell.uue). Do not add characters. | |
| REQUEST: | END |

If the file is received uncorrupted, MOSIS responds by sending an e-mail message to the user stating that the project is ready to be queued for fabrication.

For some chip designs, the UUGDS ASCII file will be very large. MOSIS has the capability to FTP large design files from the user. The LAYOUT statement can be changed, in the submission above, so that this is possible. The general form of the LAYOUT statement is

LAYOUT-FTP-PATH: !hostname!username!password!filename

Microchip Ex. 1055, p. 43
Microchip v. Aptiv
IPR2024-00494

where "!" is used as a delimiter and hostname is the name of the computer connected to the Internet where the design file resides. An example is

LAYOUT-FTP-PATH: !mycom.univ.edu!anonymous!guest!pub/chips/qcell.uue

where the entire statement must fit on one line.

*Step 3: Queuing the Project for Fabrication*

Queuing the project for fabrication occurs by sending

| | |
|---|---|
| REQUEST: | FABRICATE |
| ID: | 876543 |
| P-PASSWORD: | UNIVER |
| REQUEST: | END |

MOSIS will respond a short time after receiving the message, showing that the chip is in the queue for fabrication. When the chip is actually sent to the foundry, the user will be notified. Information about the status of the chip while being fabricated is available via anonymous FTP or the World Wide Web (http://www.mosis.org). Consult the MOSIS user's guide and the on-line information for additional information on submitting a chip for fabrication and the fabrication schedule. ■

To summarize the procedure for submitting a chip to MOSIS for fabrication, begin with a LASI-generated TLC file. Then

1. Using the command button TLC2GDS on the LASI system menu, generate a Calma Stream Format (GDS) binary file.
2. UUENCODE the GDS file. The result is a UUGDS (ASCII) file.
3. Next, the checksum program, cksum.exe, is run on the UUGDS file. This results in the generation of two numbers: checksum and byte count.
4. Send MOSIS a request for a new project ID (assuming you already have an account).
5. Submit the UUGDS file to MOSIS for syntax check (not a design rule check).
6. After MOSIS replies that the file has no syntax errors, submit the fabrication request.

*Other Information*

The MOSIS user's manual with additional information and extensive examples is located at http://www.mosis.org/manual.html. A postscript file containing the MOSIS scalable design rules is located at ftp://ftp.mosis.org/pub/mosis/designrules/scmos-rules-rev7.ps

Microchip Ex. 1055, p. 44
Microchip v. Aptiv
IPR2024-00494

## REFERENCES

[1] W. Tanner, *MOSIS User Manual,* Release 4.0, August 1994. Also located at http://www.mosis.org/manual.html.

[2] D. E. Boyce, *LASI Users Manual,* available by pressing F1 while running LASI or may be printed by running the MANUAL.EXE program included with the LASI system files.

## PROBLEMS

For the following problems, use the LASI setups given in Appendix A and in the C:\LASI5\CN20 directory for the CN20 process.

**1.1** Create a cell called test3 with a rank of 1 using LASI. In this cell, draw a 10 μm by 10 μm box using the poll layer. Place the lower left corner of the box at the origin. Use the "z" (used to set the zero point) and the spacebar to measure the distance between opposite corners.

**1.2** Explain how the **qMov** command can be used to edit the box in Problem 1 so that it measures 5 x 8 $\mu m^2$. How would this be accomplished using **Get**, **Mov**, and **Put**?

**1.3** What functions do the **sWin** and **rWin** commands perform?

**1.4** What functions do the **cGet** and **cPut** commands perform?

**1.5** The **Text** command allows text to be used for labeling in LASI. The **tLayr** selects which layer the text will be written in, while the **tSiz** sets the size of the text in 1.5 μm increments. Write the word "test" on the met1 layer with sizes of 3, 9, and 24 μm in the test3 cell of Problem 1. Show the result without using the reference mark. (The reference mark is removed by pressing **t** on the keyboard or by selecting the **T** in the top right corner of the drawing display, remembering to execute a **Draw** command afterward.) Labeling is extremely important in layout.

**1.6** Create a cell named test4 with a rank of 1. Generate the layout shown in Fig. P1.6a in this cell. The text and boxes are written using the met1 layer. Next create a cell named test5 with a rank of 2. Add the test4 cell into the test5 cell five times as shown in Fig. P1.6b. The **cGet** and **Mov** commands may come in handy at this point. Next draw the cells as outlines, shown in Fig. P1.6c, using the **Outl** command. Note that we could have used the **Cpy** command to copy the layout in test4 five times and avoid adding the test5 cell. The problem with this is that as the layout becomes complicated the memory required in a "flat" cell increases dramatically. The hierarchical layout using the nested cells keeps memory usage to a minimum. The **Cpy** command should be used as little as possible.

Microchip Ex. 1055, p. 45
Microchip v. Aptiv
IPR2024-00494

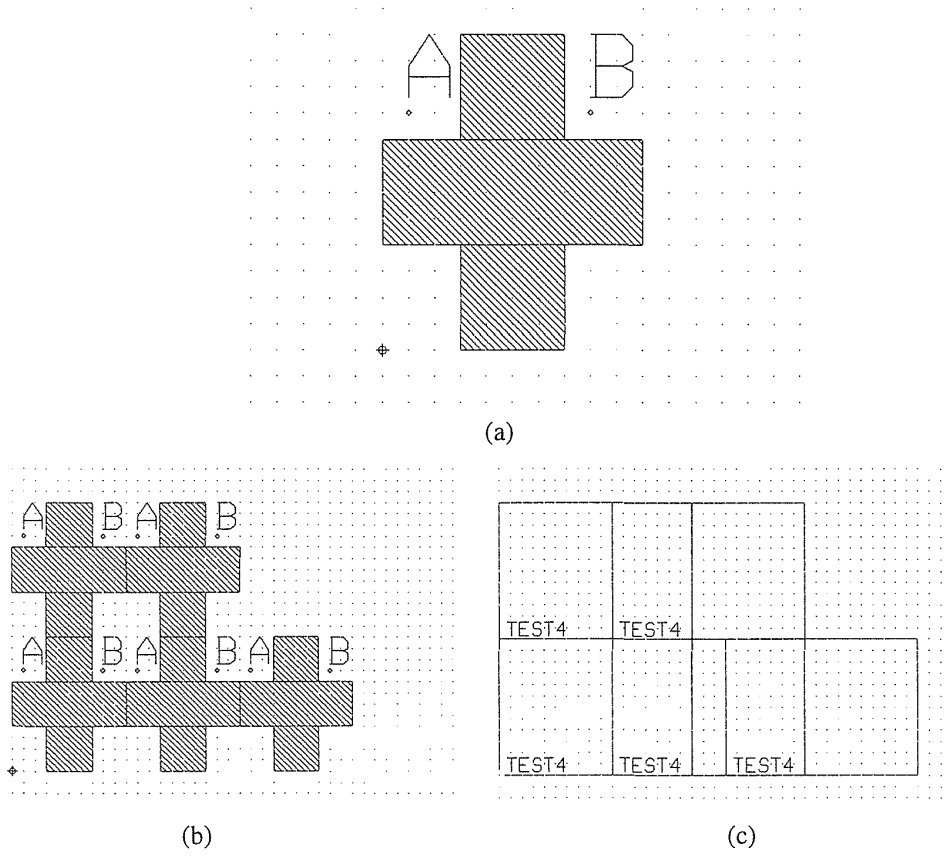



**Figure P1.6**

**1.7** Polygons or paths can be drawn using LASI by setting the **Obj** to "p" (instead of "b" for box). Setting the **Wdth** size to 0 causes LASI to draw polygons, while setting the width greater than 0 causes LASI to draw paths. When finished drawing a path or polygon, use the **aPut** command. Using LASI, copy the layout shown in Fig. P1.6a using the polygon object.

**1.8** What part of an object made using the polygon or path must be encompassed to **Get** the object?

**1.9** Using the poll layer, draw a triangle that measures nominally 10 μm on each side. How many vertices does the object have?

**1.10** Circles can be drawn using a polygon (path with zero width) and the **Arc** command. Consider the layout shown in Fig. P1.10. Copy this layout in LASI. Begin by adding a vertex at point A using the **Add** command. This is followed by selecting the **Arc** command, moving and clicking the mouse in the desired center of the circle, and returning and clicking the mouse at point A. The

Microchip Ex. 1055, p. 46
Microchip v. Aptiv
IPR2024-00494

bottom of the display will then inquire how many segments should be used and which direction to draw, that is, counterclockwise or clockwise. Hit the Enter key to both of these questions, and LASI will draw the following circle.

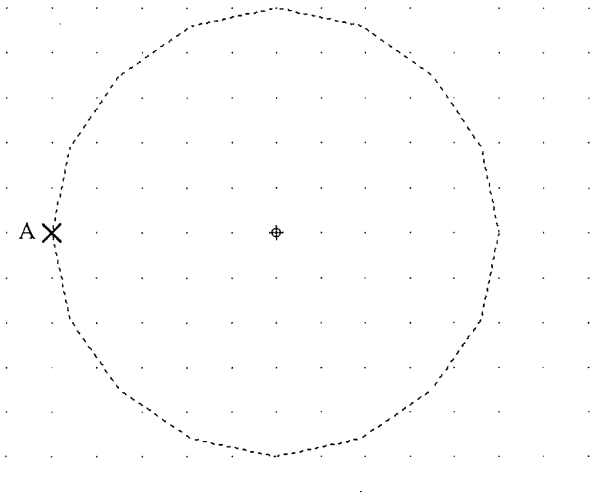



**Figure P1.10**

**1.11** Pressing F1 on the keyboard within the LASI program calls the LASI help file. What would be added to the end of the form.dbd file in a drawing directory so that F2 performs the LASI command **Fit** and F3 performs an **aPut**? Frequently used commands can be executed using function keys to help speed up layout time.

**1.12** Using the **Dpth** command, show how the cells in Fig. P1.6b can be drawn as outlines. What does the depth level mean? Show that, by pressing **i** on the keyboard (or top right corner of the drawing screen), the cells are also drawn as outlines.

**1.13** What do the keyboard buttons **w**, **u**, **a**, **z**, and **space** do in LASI?

**1.14** Describe how to add text in LASI and how to set the text size and layer.

**1.15** Using LASI, show example layouts that show the difference between path objects and the poly objects. Use poly1 in your examples. How do you **Get** a poly or path object?

Microchip Ex. 1055, p. 47
Microchip v. Aptiv
IPR2024-00494

Chapter

2

# The Well

In order to develop a fundamental understanding of CMOS integrated circuit layout and design, we begin with a study of the n-well. This approach will build a solid foundation for understanding the performance limitations and parasitics (the pn junctions, capacitances, and resistances inherent in a CMOS circuit) of the CMOS process.

## 2.1 The Substrate

CMOS circuits are fabricated on and in a silicon wafer as was discussed in Ch. 1. This wafer is doped with donor atoms, such as phosphorus for an n-type wafer, or acceptor atoms, such as boron for a p-type wafer. Our discussion will center around a p-type wafer. When designing CMOS integrated circuits with a p-type wafer, n-channel MOSFETs (NMOS for short) are fabricated directly in the p-type wafer, while the p-channel transistors, PMOS, are fabricated in an "n-well." The substrate or well are sometimes referred to as the bulk or body of a MOSFET. CMOS processes that fabricate MOSFETs in the bulk are known as "bulk CMOS processes." The well and the substrate are illustrated in Fig. 2.1, though not to scale.

Often an epitaxial layer is grown on the wafer. We will not make a distinction between this layer and the substrate. Some processes use a p-well or both n- and p-wells (sometimes called twin tub processes). A process that uses a p-type substrate with an n-well is called an "n-well process." The process described in Appendix A, CN20, is an n-well process. The n-well acts as the substrate or body of the p-channel transistors.

Another important consideration is that the n-well and the p-substrate form a diode (Fig. 2.2). In CMOS circuits, the substrate is usually tied to the lowest voltage in the circuit to keep this diode from forward biasing. Ideally, zero current flows through the substrate connection.

Microchip Ex. 1055, p. 48
Microchip v. Aptiv
IPR2024-00494

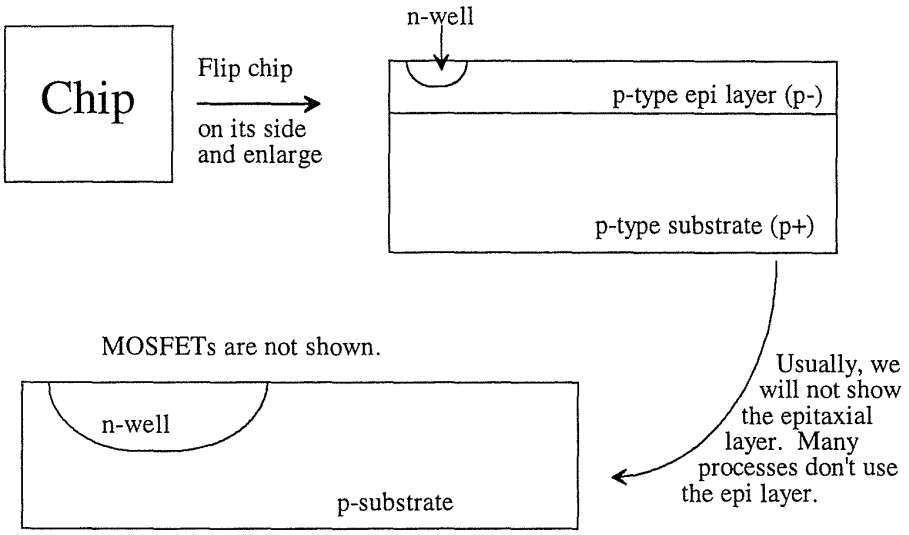



**Figure 2.1** Illustration of the top and side view of a die.

Besides being used for fabricating p-channel transistors, the n-well can be used as a resistor. The voltage on either side of the resistor must be large enough to keep the substrate/well diode from forward biasing.

### 2.1.1 Patterning

CMOS integrated circuits are formed by patterning different layers on and in the CMOS wafer. Consider the following sequence of events that apply, in a fundamental way, to any layer we need to pattern. We start out with a clean, bare wafer as shown in Fig. 2.3a. The distance given by the line A to B will be used as a reference in Figs. 2.3b-j. Figures 2.3b-j are cross-sectional views of the dashed line shown in (a).

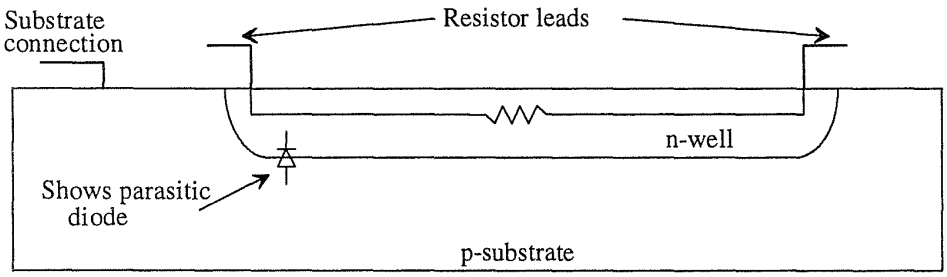



**Figure 2.2** The n-well can be used as a resistor.

Microchip Ex. 1055, p. 49
Microchip v. Aptiv
IPR2024-00494

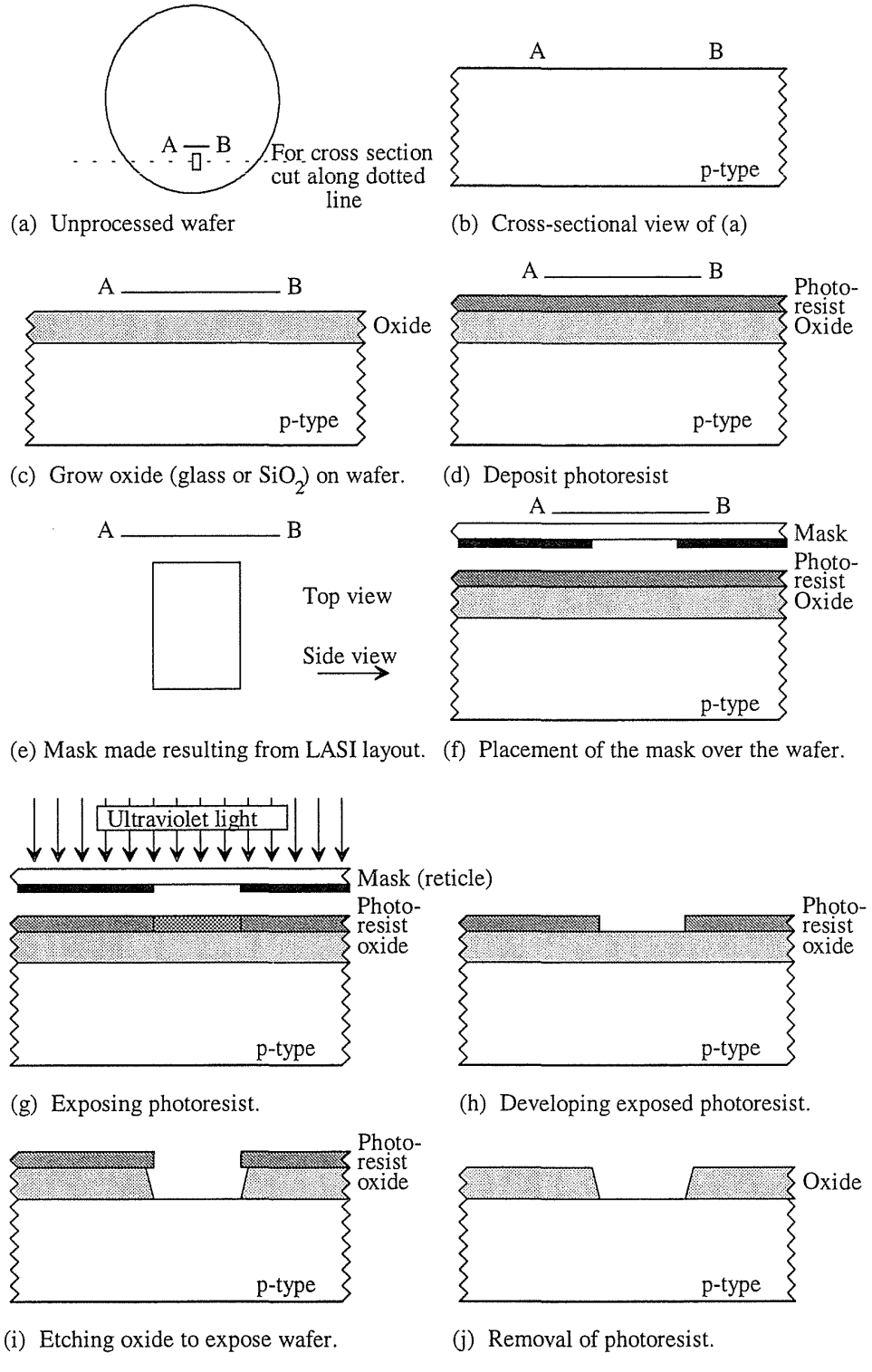



**Figure 2.3** Sequence of events used in patterning.

Microchip Ex. 1055, p. 50
Microchip v. Aptiv
IPR2024-00494

The first step in our generic patterning discussion is to grow an oxide, $SiO_2$ or glass, a very good insulator, on the wafer. Simply exposing the wafer to air yields the reaction $Si + O_2 \rightarrow SiO_2$. However, semiconductor processes must have tightly controlled conditions to precisely set the thickness and purity of the oxide. Therefore, we can grow the oxide using a reaction with steam, $H_2O$, or with $O_2$ alone. The oxide resulting from the reaction with steam is called a wet oxide, while the reaction with $O_2$ is a dry oxide. Both oxides are called thermal oxides due to the increased temperature used during oxide growth. The growth rate increases with temperature. The main benefit of the wet oxide is fast growing time. The main drawback of the wet oxide is the hydrogen byproduct. In general terms, the oxide grown using the wet techniques is not as pure as the dry oxide. The dry oxide, as we can conclude, generally takes a considerably longer time to grow. Both methods of growing oxide are found in CMOS processes.

An important observation we should make when looking at Fig. 2.3c is that the oxide growth actually consumes silicon. This is illustrated in Fig. 2.4. The overall thickness of the oxide is related to thickness of the consumed silicon by

$$t_{Si} = 0.45 \cdot t_{ox} \tag{2.1}$$

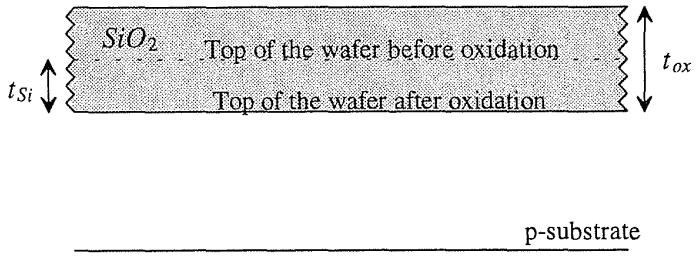



**Figure 2.4** How growing oxide consumes silicon.

The next step of the CMOS patterning process is to deposit a photosensitive resist layer across the wafer (see Fig. 2.3d). Keep in mind that the dimensions of the layers, that is, oxide, resist, and the wafer, are not drawn to scale. The thickness of a wafer is typically 500 μm, while the thickness of a grown oxide or a deposited resist may be only a few μm or even less. After the resist is baked, the mask derived from the layout program, Figs. 2.3e and f, is used to selectively illuminate areas of the wafer, Fig. 2.3g. In practice, a single mask called a reticle, with openings several times larger than the final illuminated area on the wafer, is used to project the pattern and is stepped across the wafer with a machine called a stepper to generate the patterns needed to create multiple copies of a single chip. The light passing through the opening in the reticle is photographically reduced to illuminate the correct size area on the wafer.

Microchip Ex. 1055, p. 51
Microchip v. Aptiv
IPR2024-00494

The photoresist is developed (Fig. 2.3h), removing the areas that were illuminated. This process is called a positive resist process because the area that was illuminated was removed. A negative resist process removes the areas of resist that were not exposed to the light. Using both types of resist allows the process designer to cut down on the number of masks needed to define a CMOS process. Since creating the masks is expensive, lowering the number of masks is equated with lowering the cost of a process. This is also important in large manufacturing plants where fewer steps are equated with lower cost.

The next step in the patterning process is to remove the exposed oxide areas (Fig. 2.3i). Notice that the etchant etches under the resist, causing the opening in the oxide to be larger than what was specified by the mask. Some manufacturers intentionally bloat (make larger) or shrink (make smaller) the masks as specified by the layout program. Figure 2.3j shows the cross-sectional view of the opening after the resist has been removed.

### 2.1.2 Patterning the N-well

At this point we can make an n-well by diffusing donor atoms, those with 5 valence electrons, as compared to 4 for silicon, into the wafer. Referring to our generic patterning discussion given in Fig. 2.3, we begin by depositing a layer of resist directly on the wafer, Fig. 2.3d (without oxide). This is followed by exposing the resist to light through a mask (Figs. 2.3f and g) and developing or removing the resist (Fig. 2.3h). The mask used here can be generated with the LASI program. The next step in fabricating the n-well is to expose the wafer to donor atoms. The resist will block the diffusion of the atoms, while the openings will allow the donor atoms to penetrate into the wafer. This is shown in Fig. 2.5a. After a certain amount of time, dependent on the depth of the n-well desired, the diffusion source is removed (Fig. 2.5b). Notice that the n-well "outdiffuses" under the resist; that is, the final n-well size is not the same as the mask size. Again, the foundry where the chips are fabricated may bloat or shrink the mask to compensate for this *lateral diffusion*. The final step in making the n-well is the removal of the resist (Fig. 2.5c).

## 2.2 Laying out the N-well

When we lay out the n-well, we are viewing the chip from the top. The following example illustrates how to lay out an n-well of size 10 μm square.

---

**Example 2.1**

Using the LASI program, lay out an n-well that is 10 μm square. Sketch the cross-sectional view of the layout. Assume we are using the CN20 setups given in the last chapter and using the layer table select **Layr** from the drawing screen commands and the layer "nwel." If the origin marker is not showing, type "r." Now select **Add**. The bottom of the display should show that the object is a box and that the grids are 1 μm. Place the cursor in the display area on the origin

Microchip Ex. 1055, p. 52
Microchip v. Aptiv
IPR2024-00494

and press and release the LEFT mouse button. Move the mouse until the distance shows $ux$ = 10 μm and $uy$ = 10 μm. The resulting display is shown in the top of Fig. 2.6. Notice that the drawn size, both width and length, of the n-well differs from the actual size because of the lateral diffusion. ■

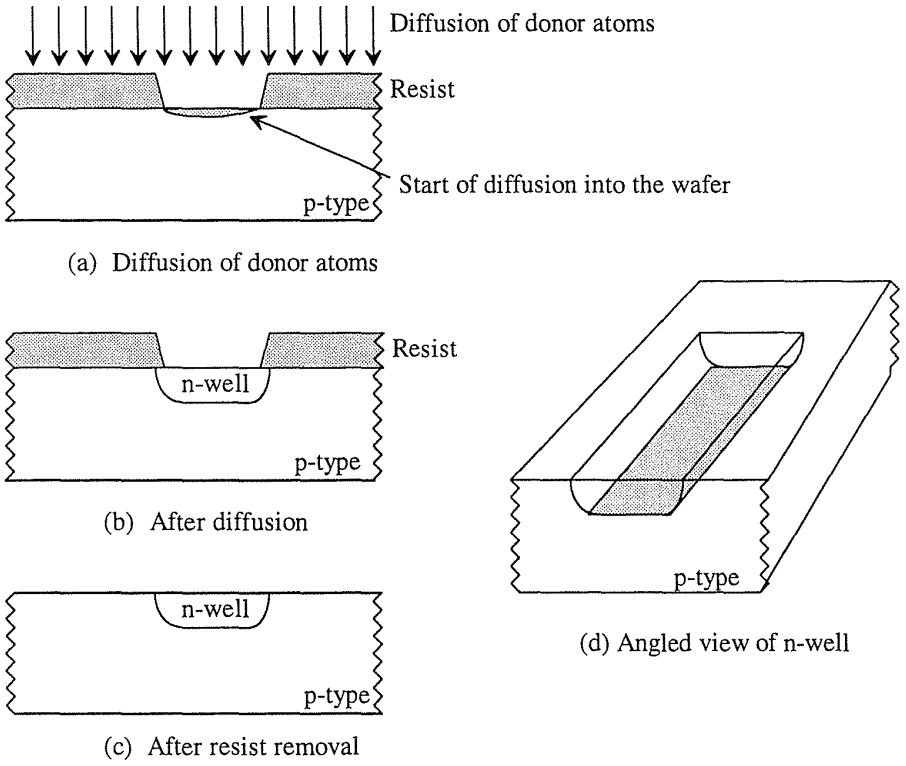



**Figure 2.5** Formation of the n-well.

### 2.2.1 Design Rules for the N-well

Now that we know how to lay out the n-well, we might ask the question, "Are there any limitations or constraints on the size and spacing of the n-wells?" That is to say, can we make the n-well 2 μm square? Can we make the distance between the n-wells 1 μm? As we might expect, there are minimum spacing and size requirements for all layers in a CMOS process. Process engineers, who design the integrated circuit process, specify the design rules. A complete listing of the CN20 design rules can be found in Appendix A.

Figure 2.7 shows the design rules for the n-well. The minimum width of any n-well is 3 μm, while the minimum spacing between different n-wells is 9 μm. As the layout becomes complicated, the need for a program that ensures that the design rules are not violated is needed.

Microchip Ex. 1055, p. 53
Microchip v. Aptiv
IPR2024-00494

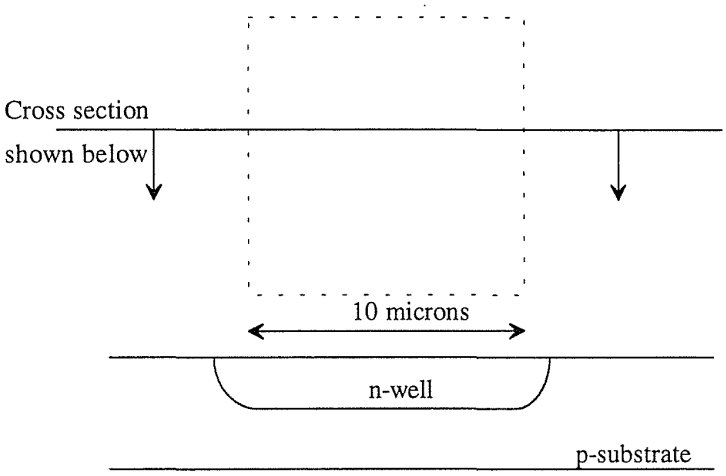



**Figure 2.6** Layout and cross-sectional view of a 10 μm n-well.

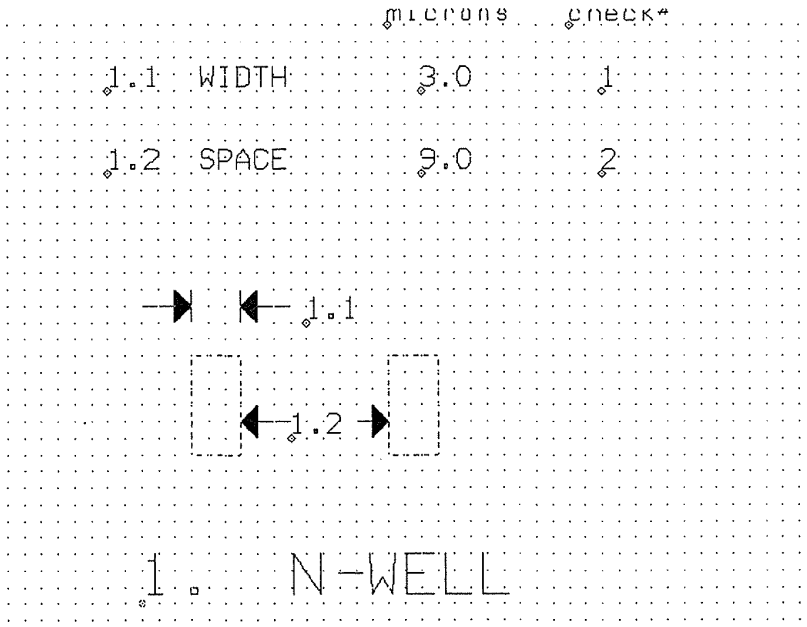



**Figure 2.7** Design rules for layout of the n-well.

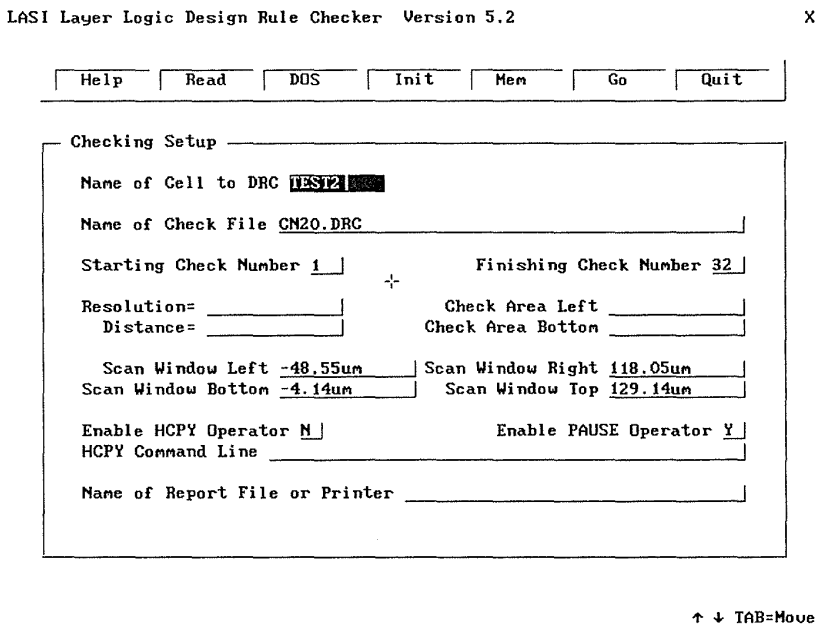
Microchip Ex. 1055, p. 54
Microchip v. Aptiv
IPR2024-00494

### 2.2.2 Using the LASIDRC Program

To perform a design rule check (DRC) of a layout, simply select the LASIDRC at the LASI system menu. Enter the name of the cell (see below) to be checked and type cn20.drc as the name of the check file. Since we only know two design rules at this point, that is, from Fig. 2.7 checks 1 and 2, the beginning check should be 1 and the ending check should be 2. To start the program, select Go on the top of the LASIDRC screen. If there is an error, a bit map of the n-well in blue will be displayed with the error in white. You can exit LASIDRC and go to LASI to fix the problem. When you return to LASIDRC, you can continue your checking at the check you failed.

*DRC Area*

The design rule check will be performed on the *section* of layout shown in the drawing display just prior to calling the LASIDRC program. To DRC the entire cell, press the **Init** command button on the LASIDRC screen shown below. This feature can be used to decrease the time it takes to perform a DRC by DRCing only the specific areas of interest.




## 2.3 Resistance Calculation

In addition to serving as a region in which to build PMOS transistors, n-wells are sometimes used to create integrated resistors. The resistance of a material is a function of the materials resistivity, ρ, and the materials dimensions. For example, the slab of material in Fig. 2.8 between the two leads has a resistance given by

Microchip Ex. 1055, p. 55
Microchip v. Aptiv
IPR2024-00494

$$R = \frac{\rho}{t} \cdot \frac{L}{W} \tag{2.2}$$

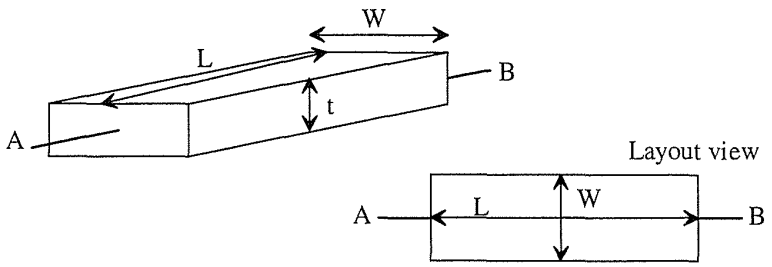



**Figure 2.8** Calculation of the resistance of a rectangular block of material.

In semiconductor processing, all of the fabricated thicknesses, such as the n-well, are constant. We only have control over $W$ and $L$. Also notice that the $W$ and $L$ are what we see from the top view, that is, the layout view. We can rewrite Eq. (2.2) as

$$R = R_{square} \cdot \frac{L}{W} \tag{2.3}$$

$R_{square}$ is the sheet resistance of the material in Ω/square.

**Example 2.2**

Calculate the resistance of an n-well that is 10 μm wide and 100 μm long.

From the Orbit electrical parameters in Appendix A for CN20 we see that the sheet resistance of the n-well varies from a minimum of 2,000 Ω/square to a maximum of 3,000 Ω/square, with a typical number of 2,500 Ω/square. So the typical resistance between the ends of the n-well is

$$R = 2,500 \cdot \frac{100\ \mu m}{10\ \mu m} = 25\ k\Omega \ \blacksquare$$

When laying out resistors using the "path" object, n-well, poly, or some other layer, LASI has a resistor calculator that will help in the calculation of nonrectangular resistances. Often, to minimize space, resistors are laid out in a serpentine pattern. The corners, that is, where the layer bends, are not rectangular. This is shown in Fig. 2.9a. All sections in Fig 2.9a are square, so the resistance of sections 1 and 3 is $R_{square}$. The equivalent resistance of section 2 between the adjacent sides, however, is approximately $0.6\ R_{square}$. The overall resistance between points A and B is therefore $2.6 \cdot R_{square}$.

The layout shown in Fig. 2.9b uses wires to connect separate sections of a resistor avoiding corners. Avoiding corners in a resistor is the generally preferred method of layout in analog circuit design where the ratio of two resistors is important.

Microchip Ex. 1055, p. 56
Microchip v. Aptiv
IPR2024-00494

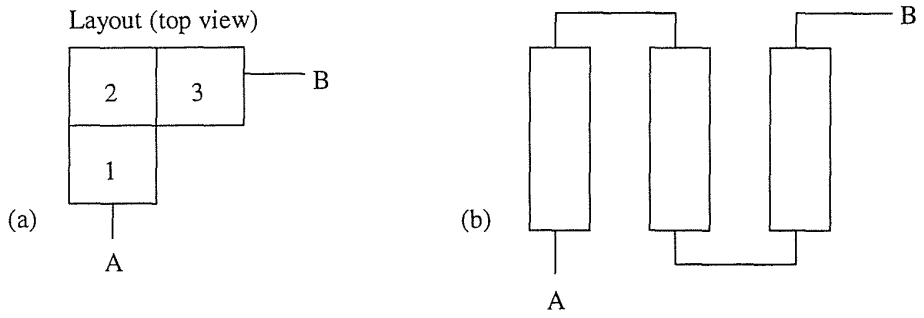



**Figure 2.9** (a) Figure for the calculation of the resistance of a corner section and (b) layout to avoid corners.

### 2.3.1 The N-well Resistor

At this point, it is appropriate to show the actual cross-sectional view of the n-well after all processing steps are completed (Fig. 2.10). The n+ and p+ implants are used to increase the threshold voltage of the field devices; more will be said on this later in Ch. 4. Notice in Appendix A, the Orbit electrical parameters, that the sheet resistance of the n-well is measured with the field implant in place, that is, with the n+ implant between the two metal connections in Fig. 2.10. Not shown in Fig. 2.10 is the connection to substrate. The field oxide (FOX; also know as ROX or recessed oxide) will be discussed in Ch. 4 when we discuss the active and poly layers.

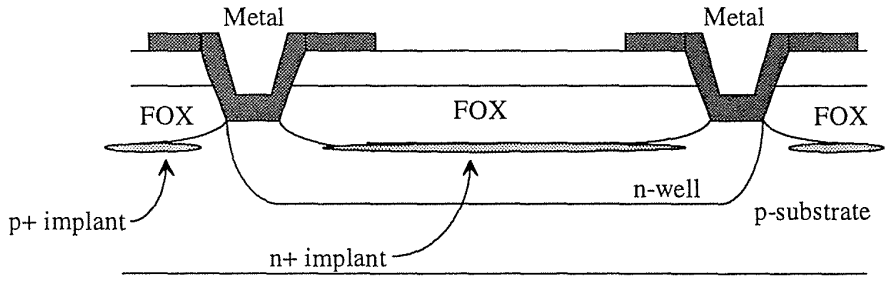



**Figure 2.10** Cross-sectional view of n-well showing field implant.

## 2.4 The N-well/substrate Diode

Placing an n-well in the p-substrate forms a diode. Therefore, it is important to understand how to model a diode for hand calculations and in SPICE simulations. In particular, let's discuss diodes using the n-well/substrate pn junction as an example [2]. The DC characteristics of the diode are given by the Shockley diode equation, or

Microchip Ex. 1055, p. 57
Microchip v. Aptiv
IPR2024-00494

$$I_D = I_S\left(e^{\frac{V_d}{nV_T}} - 1\right) \tag{2.4}$$

The current $I_D$ is the diode current; $I_S$ is the scale (saturation) current; $V_d$ is the voltage across the diode where the anode (p-type material) is assumed positive with respect to the cathode (n-type); and $V_T$ is the thermal voltage which is given by $\frac{kT}{q}$ where $k$ = Boltzmann's constant ($1.3806 \times 10^{-23}$ joules per degree kelvin), $T$ is temperature in kelvin, $n$ is the emission coefficient (a term that is related to the doping profile and affects the exponential behavior of the diode), and $q$ is the electron charge of $1.6022 \times 10^{-19}$ coulombs. The scale current and thus the overall diode current are related in SPICE by an area factor. The SPICE circuit simulation program assumes that the value of $I_s$ supplied in the model statement was measured for a device with a reference area of 1. If an area factor of 2 is supplied for a diode, then $I_S$ is doubled in Eq. (2.4).

### 2.4.1 Depletion Layer Capacitance

N-type silicon has a number of mobile electrons, while p-type silicon has a number of mobile holes (a vacancy of electrons in the valence band). Formation of a pn junction results in a depleted region at the p-n interface (Fig. 2.11). A depletion region is an area depleted of mobile holes or electrons. The mobile electrons move across the junction, leaving behind fixed donor atoms and thus a positive charge. The movement of holes across the junction, to the right in Fig. 2.11, occurs for the p-type semiconductor as well with a resulting negative charge. The fixed atoms on each side of the junction within the depleted region exert a force on the electrons or holes that have crossed the junction. This equalizes the charge distribution in the diode, preventing further charges from crossing the diode junction and also gives rise to a parasitic (depletion) capacitance.

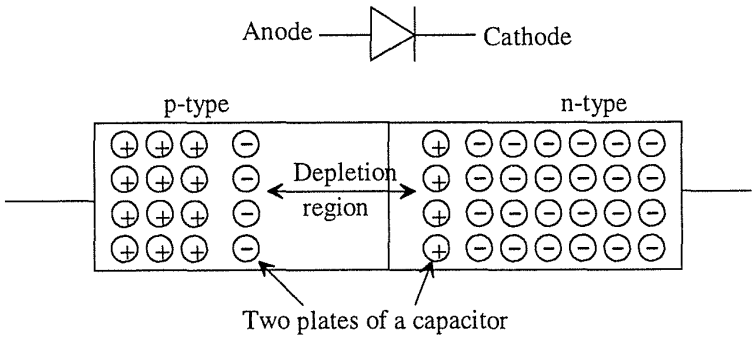



**Figure 2.11** Simple illustration of depletion region formation in a pn junction.

The depletion capacitance, $C_j$, of a pn junction is given by

$$C_j = \frac{C_{j0}}{\left[1 - \left(\frac{V_d}{\varphi_0}\right)\right]^m} \tag{2.5}$$

Microchip Ex. 1055, p. 58
Microchip v. Aptiv
IPR2024-00494

$C_{j0}$ is the zero-bias capacitance of the pn junction, that is, the capacitance when the voltage across the diode is zero. $V_d$ is the voltage across the diode, $m$ is the grading coefficient (showing how the silicon changes from n- to p-type), and $\phi_0$ is the built-in potential given by

$$\phi_0 = V_T \cdot \ln\left(\frac{N_A N_D}{n_i^2}\right) \tag{2.6}$$

where $N_A$ and $N_D$ are the dopings for the p- and n-type semiconductors, respectively, $V_T$ is the thermal equivalent voltage $\frac{kT}{q}$(26 mV @ room temperature), and $n_i$ is the intrinsic carrier concentration of silicon ($n_i = 14.5 \times 10^9$ atoms/cm$^3$).

**Example 2.3**

Sketch schematically the depletion capacitance of an n-well/p-substrate diode $100 \times 100$ μm$^2$ square given that the substrate doping is $10^{16}$ atoms/cm$^3$ and the well doping is $10^{16}$ atoms/cm$^3$. The measured zero-bias depletion capacitance of the junction is 100 aF/μm$^2$, and the grading coefficient is 0.333. Assume the depth of the n-well is 3 μm.

We can begin this problem by calculating the built-in potential using Eq. (2.6):

$$\phi_0 = (.026) \cdot \ln\frac{10^{16} \cdot 10^{16}}{(14.5 \times 10^9)^2} = 0.7 \text{ V}$$

The depletion capacitance is made up of a *bottom* component and a *sidewall* component as shown in Fig. 2.12.

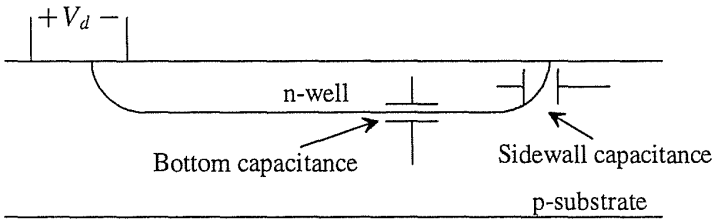



**Figure 2.12** A pn junction on the bottom and sides of the junction.

The bottom zero-bias depletion capacitance, $C_{j0b}$, is given by

$C_{j0b}$ = (Capacitance per Area) · (Bottom Area, which, for this example, is

$$C_{j0b} = (100 \text{ aF/μm}^2) \cdot (100 \text{ μm})^2 = 1 \text{ pF}$$

The sidewall zero-bias depletion capacitance, $C_{j0s}$, is given by

$C_{j0s}$ = (Capacitance per Area) · (Depth of the Well) · (Perimeter of the Well)

or

Microchip Ex. 1055, p. 59
Microchip v. Aptiv
IPR2024-00494

$$C_{j0s} = (100\ \text{aF}/\mu\text{m}^2) \cdot (3\ \mu\text{m}) \cdot (400\ \mu\text{m}) = 120\ \text{fF}$$

The total diode depletion capacitance between the n-well and the p-substrate is the parallel combination of the bottom and sidewall capacitances, or

$$C_j = \frac{C_{j0b}}{\left[1 - \left(\frac{V_d}{\varphi_0}\right)\right]^m} + \frac{C_{j0s}}{\left[1 - \left(\frac{V_d}{\varphi_0}\right)\right]^m} = \frac{C_{j0b} + C_{j0s}}{\left[1 - \left(\frac{V_d}{\varphi_0}\right)\right]^m}$$

Substituting in the numbers, we get

$$C_j = \frac{1\ \text{pF} + 0.120\ \text{pF}}{\left(1 - \left(\frac{V_d}{0.7}\right)\right)^{0.33}}$$

A sketch of how this capacitance changes with reverse potential is given in Fig. 2.13. Notice that when we discuss the depletion capacitance of a diode, it is usually with regard to a reverse bias. When the diode becomes forward-biased minority carriers, electrons in the p material and holes in the n material, injected across the junction, form a stored charge in and around the junction and give rise to a storage capacitance. This capacitance is usually much larger than the depletion capacitance. Furthermore, the time it takes to remove this stored charge can be significant. ■

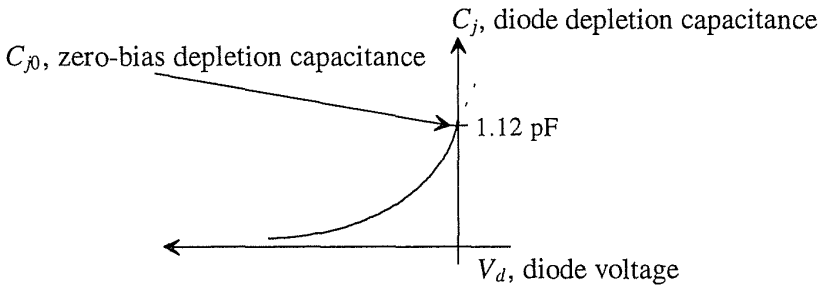



**Figure 2.13** Sketch of diode depletion capacitance against diode reverse voltage.

### 2.4.2 Storage Capacitance

Consider the charge distribution of a forward-biased diode shown in Fig. 2.14. When the diode becomes forward biased, electrons from the n-type side of the junction are attracted to the p-type side (and vice versa for the holes). After an electron drifts across the junction, it starts to diffuse toward the metal contact. If the electron recombines, that is, falls into a hole, before it hits the metal contact, the diode is called a "long base diode." The time it takes an electron to diffuse from the junction to the point the electron recombines is called the carrier lifetime. For silicon this lifetime is on the order of 10 μs. If the distance between the junction and the metal contact is short, such that the electrons make it to the metal contact before recombining, the diode is said to be a "short base diode." In either case, the time between crossing the junction and

Microchip Ex. 1055, p. 60
Microchip v. Aptiv
IPR2024-00494

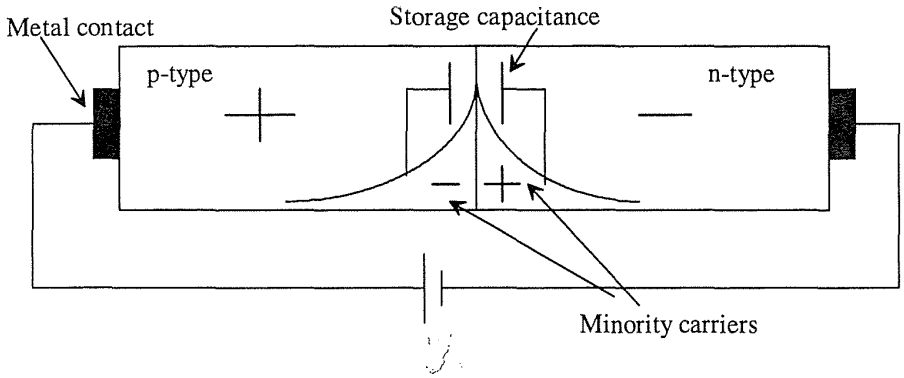



**Figure 2.14** Charge distribution in a forward-biased diode.

recombining is called the transit time, $\tau_T$. A capacitance is formed between the electrons diffusing into the p-side and the holes diffusing into the n-side, that is, formed between the minority carriers. (Electrons are the minority carriers in the p-type semiconductor.) This capacitance is often called a diffusion capacitance or storage capacitance (due to the presence of the stored minority carriers around the junction).

We can characterize the storage capacitance, $C_S$, in terms of the minority carrier lifetime. Under DC operating conditions, the storage capacitance is given by

$$C_S = \frac{I_D}{nV_T} \cdot \tau_T \tag{2.7}$$

$I_D$ is the DC current flowing through the forward-biased junction given by Eq. (2.4). Looking at the diode capacitance in this way is very useful for analog AC small-signal analysis. However, for digital applications we are more interested in the large-signal switching behavior of the diode. It should be pointed out that in general, for a CMOS process, it is undesirable to have a forward-biased pn junction. If we do have a forward-biased junction, it usually means there is a problem, for example, electrostatic protection, capacitive feedthrough possibly causing latch-up, and so on. These topics are discussed in more detail later in this chapter.

Consider Fig. 2.15. In the following diode switching analysis, we will assume that $V_F >> 0.7$, $V_R < 0$ and that the voltage source has been at $V_F$ long enough to reach steady-state condition; that is, the minority carriers have diffused out to an equilibrium condition. At the time $t_1$ the input voltage source makes an abrupt transition from $V_F$ to $V_R$, causing the current to change from $\frac{V_F}{R}$ to $\frac{V_R}{R}$. The diode voltage remains at 0.7 V since the diode contains a stored charge that must be removed . At time $t_2$ the stored charge is removed. At this point, the diode basically looks like a voltage-dependent capacitor that follows Eq. (2.5). In other words for $t > t_2$ the diode depletion capacitance is charged through $R$ until the current in the circuit goes to zero and the voltage across the diode is $V_R$. This accounts for the exponential decay of the current and voltage shown in Fig. 2.15.

Microchip Ex. 1055, p. 61
Microchip v. Aptiv
IPR2024-00494

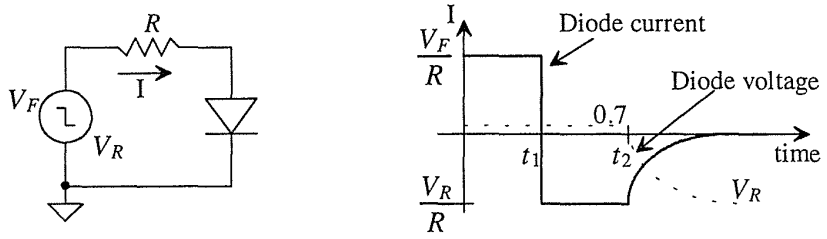



**Figure 2.15** Diode test circuit.

The diode storage time, the time it takes to remove the stored charge, $t_S$, is simply the difference in $t_2$ and $t_1$, or

$$t_S = t_2 - t_1 \tag{2.8}$$

This time is also given by

$$t_S = \tau_T \cdot \ln \frac{i_F - i_R}{-i_R} \tag{2.9}$$

where $\frac{V_F}{R} = i_F$ and $\frac{V_R}{R} = i_R$ = a negative number in this discussion. Note that it is quite easy to determine the minority carrier lifetime using this test setup.

Defining a time $t_3$, where $t_3 > t_2$, when the current in the diode becomes 10 percent of $\frac{V_R}{R}$, we can define the diode reverse recovery time, or

$$t_{rr} = t_3 - t_1 \tag{2.10}$$

### 2.4.3 SPICE Modeling

The SPICE (simulation program with IC emphasis) diode model parameters are listed in Table 2.1. The series resistance, $R_s$, deserves some additional comment. This resistance results from the finite resistance of the semiconductor used in making the diode and the contact resistance, the resistance resulting from a metal contact to the semiconductor. At this point, we are only concerned with the resistance of the semiconductor. For a reverse-biased diode, the depletion layer width changes, increasing for larger reverse voltages (decreasing both the capacitance and series resistance, of the diode). However, when we model the series resistance, we use a constant value. In other words, SPICE will not show us the effects of a varying $R_S$.

**Example 2.4**

Using SPICE, explain what happens when a diode with a carrier lifetime of 30 ns is taken from the forward-biased region to the reverse-biased region. Use the circuit shown in Fig. Ex2.4 to illustrate your understanding.

We will assume a zero-bias depletion capacitance of 1 pF. The SPICE netlist for this circuit is shown below.

Microchip Ex. 1055, p. 62
Microchip v. Aptiv
IPR2024-00494

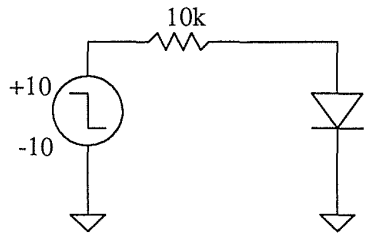



**Figure Ex2.4**

```
Diode storage time.
D1      1 0 TRR
R1      3 1 10k
Vin     3 0     DC 0 PULSE(10 -10 50n .1n .1n 50n 100n)
.Model TRR  D
+ IS=1.0E-15 TT=30E-9 CJO=1E-12 VJ=.7 M=0.33
.probe
.tran 1n 100n
.end
```

Figure 2.16 shows the current through the diode (scaled), the input voltage step, and the voltage across the diode. One of the interesting things to notice about this circuit is that current actually flows through the diode in the negative direction, even though the diode is forward biased. During this time, the stored minority carrier charge is removed from the junction. The storage time is given by

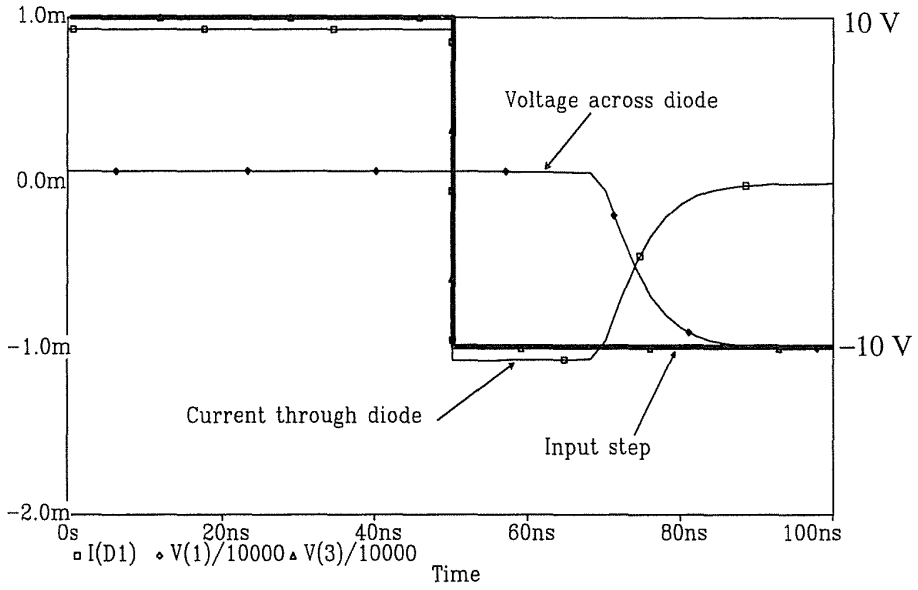
$$t_S = 30 \text{ ns} \cdot \ln \frac{0.93 \text{ mA} + 0.107 \text{ mA}}{0.107 \text{ mA}} = 18.8 \text{ ns}$$

which is close to the simulation results. Note that the input pulse doesn't change until 50 ns after the simulation starts. This ensures a steady-state condition when the input changes from 10 to –10 V. ■

| Name | SPICE | |
|---|---|---|
| $I_S$ | IS | Saturation current |
| $R_S$ | RS | Series resistance |
| $n$ | N | Emission coefficient |
| $V_{bd}$ | BV | Breakdown voltage |
| $I_{bd}$ | IBV | Current which flows during $V_{bd}$ |
| $C_{j0}$ | CJ0 | Zero-bias pn junction capacitance |
| $\varphi_0$ | VJ | Built-in potential |
| $m$ | M | Grading coefficient |
| $\tau_T$ | TT | Carrier transit time |

**Table 2.1** SPICE parameters related to diode.

Microchip Ex. 1055, p. 63
Microchip v. Aptiv
IPR2024-00494




## 2.5 The RC Delay Through an N-well

At this point, we know that the n-well can be used as a resistor and as a diode when used with the substrate. Figure 2.17a shows the parasitic capacitance and resistance associated with the n-well. Since there is a depletion capacitance from the n-well to the substrate, we could sketch the equivalent symbol for the n-well resistor as shown in Fig. 2.17b. This is the basic form of an RC transmission line. If we put a step into one side of the n-well resistor a finite time later, called the delay time and measured at the 50 percent points of the pulses, the pulse will appear.

The delay can be calculated by knowing the resistance, $r$, per unit length, the capacitance, $c$, per unit length, and $l$, the number of unit lengths using the following [3]:

$$t_d = 0.35rcl^2 \tag{2.11}$$

**Example 2.5**

Estimate the delay through a 250 kΩ resistor made using an n-well with a width of 3 μm and a length of 300 μm. Verify your answer with SPICE.

If we divide the resistor up into 100 squares each 3 μm wide and 3 μm long, we can define the number of unit lengths, $l$, as 100. The resistance of one of these squares is 2.5 kΩ (= $r$) per unit length. Now we are faced with determining the capacitance to substrate of one of these squares. Since the capacitance is a function of the voltage, we can get a worst-case estimate by selecting the zero-bias depletion capacitance (see Fig. 2.13). From Ex. 2.3 the zero-bias

Microchip Ex. 1055, p. 64
Microchip v. Aptiv
IPR2024-00494

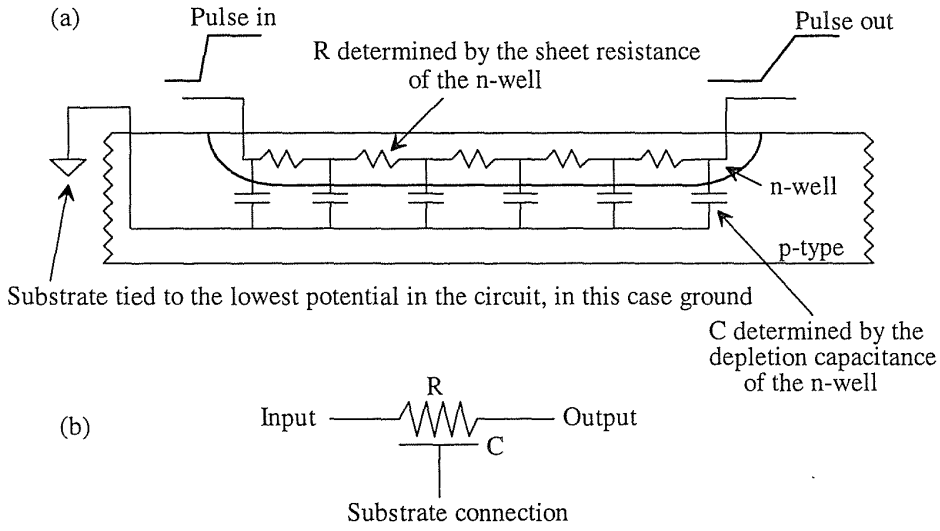



**Figure 2.17** (a) Parasitic resistance and capacitance of the n-well and (b) schematic symbol.

depletion capacitance between the n-well and substrate is 100 aF/μm². The capacitance per unit length is the sum of the bottom and sidewall capacitances. However, since each square, except for the first and last squares, has only two sides contributing to the depletion capacitance and it is desirable to keep the number of calculations to a minimum in an estimate, we will neglect the sidewall capacitance. The capacitance per unit length is given by

$$c = C_{j0b} = 100\frac{\text{aF}}{\mu\text{m}^2}(3 \cdot 3)\mu\text{m}^2 = 900\ a\text{F}$$

The delay is now estimated by

$$t_d = 0.35 \cdot rcl^2 = 0.35 \cdot 2.5\text{k} \cdot 900\ \text{aF} \cdot 100^2 = 7.88\ \text{ns}$$

The SPICE netlist and the resulting output are shown in Fig. 2.18. Note that this is a SPICE3 netlist and not a PSPICE netlist. Also note that the nodes were labeled with names (i.e., Vin and Vout) rather than numbers. ■

We can simplify Eq. (2.11) by realizing that the products $r \cdot l$ and $c \cdot l$ are the total resistance and capacitance to substrate of the n-well resistor. Using this result on the previous example gives $R = r \cdot l = 2{,}500 \cdot 100 = 250\ \text{k}\Omega$ and $C = c \cdot l = 900\ \text{aF} \cdot 100 = 90\ \text{fF}$ and therefore

$$t_d = 0.35 \cdot RC = 0.35 \cdot 250\text{k} \cdot 90\ \text{fF} = 7.88\ \text{ns} \qquad (2.12)$$

which is the same result given the example. The important thing to notice here is that we can totally avoid the unit length parameter $l$. The resistance $R$ is the resistance of

Microchip Ex. 1055, p. 65
Microchip v. Aptiv
IPR2024-00494

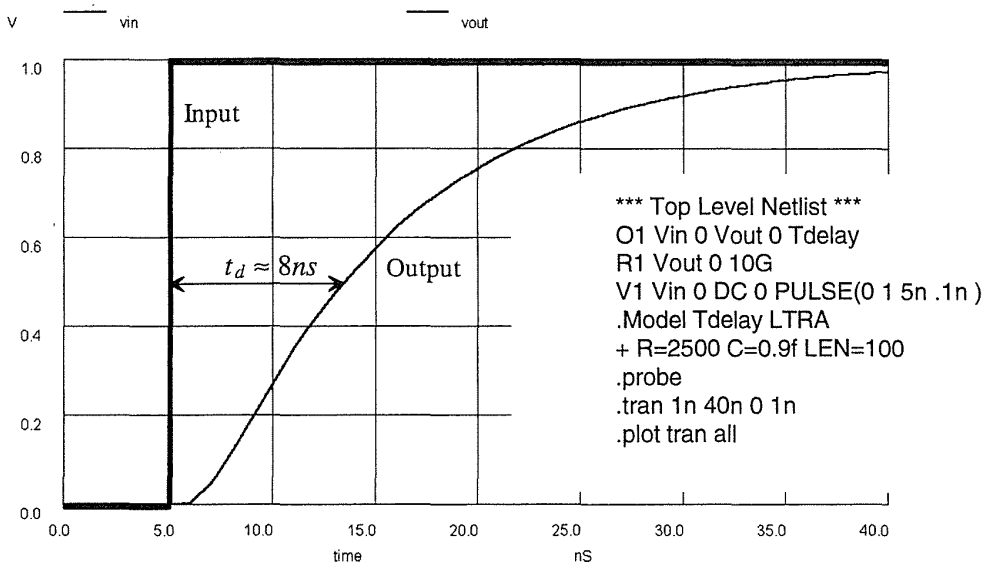



**Figure 2.18** Simulation results from Example 2.5.

the resistor ( = 250 kΩ above), and the capacitance $C$ is simply the product of the bottom area of the resistor with the zero-bias depletion capacitance (= 3 · 300 ·100 aF = 90 fF).

## REFERENCES

[1] D. E. Boyce, *LASI Users Manual,* available by pressing F1 while running LASI or by running the MANUAL.EXE program included with LASI.

[2] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits,* McGraw-Hill Publishing Company, 2nd ed., 1988. ISBN 0-07-029158-6.

[3] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design,* Addison-Wesley, 2nd ed., 1993. ISBN 0-201-53376-6.

## PROBLEMS

**2.1** Figure P2.1 is a section of n-well laid out using the path object with a width of 4 μm. Sketch the cross-sectional view (see Ex. 2.1, at the positions indicated in the figure). Copy this layout using the LASI program. Using the **Res** command (before using the **Res** command, use **fget** on the path) with a sheet resistance of 2,500 Ω/square (the typical sheet resistance of the n-well) and an end correction of 0.6, determine the resistance of the section. How does this compare with the $2.6R_{square}$ used for the layout in Fig. 2.9a?

Microchip Ex. 1055, p. 66
Microchip v. Aptiv
IPR2024-00494

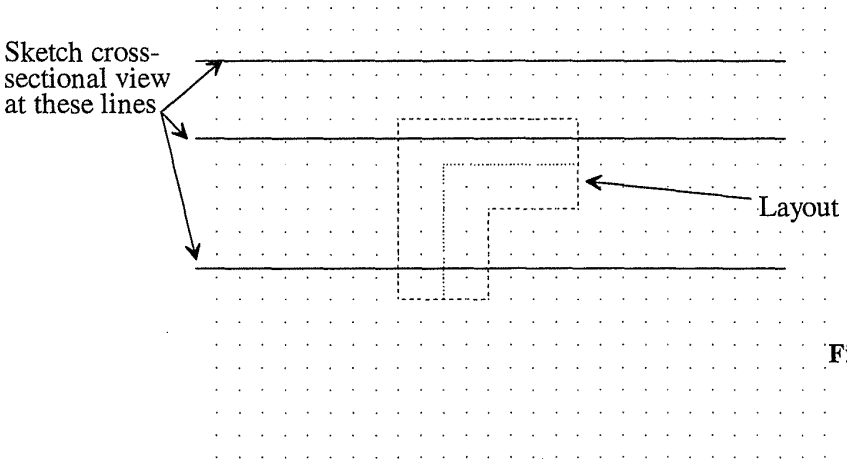



**Figure P2.1**

**2.2** Add two boxes to the layout of Problem 1 (see Fig. P2.2). Using the LASIDRC program with checks 1 and 2, show the design rule violations in the layout.

**2.3** Lay out a nominally 250 kΩ resistor using the n-well and the serpentine pattern shown in Fig. P2.3. Assume that the maximum length of a segment is 100 μm. Also design rule check the finished resistor.

**2.4** Assuming the n-well depth is 3 μm in the CN20 process, what are the minimum, typical, and maximum values of the n-well resistivity?

**2.5** Normally, the scale current of a pn junction is specified in terms of a scale current density, $J_s$ (A/m$^2$) and the width and length of a junction (i.e., $I_s = J_s \cdot L \cdot W$ neglecting the sidewall component). Estimate the scale current for the diode of Ex. 2.3 if $J_s = 10^{-8}$ A/m$^2$.

**2.6** Repeat Problem 5 including the sidewall component ($I_s = J_s \cdot L \cdot W + J_s \cdot (2L+2W) \cdot$depth).

**2.7** Using the diode of Ex. 2.3 in the following circuit, estimate the frequency of the input signal when the AC component of $v_{out}$ is 707 μV (i.e., estimate the 3 dB frequency of the $|v_{out}/v_{in}|$).

**2.8** Verify the answer given in problem 7 with SPICE.

**2.9** Using SPICE, show that a diode can conduct current from its cathode to its anode when the diode is forward biased.

**2.10** Estimate the delay through a 1 MΩ resistor (5 μm by 2,000 μm) made in the CN20 process using the n-well. Verify with SPICE.

**2.11** If one end of the resistor in Problem 10 is tied to +5 V and the other end is tied to the substrate that is tied to ground, estimate the depletion capacitance (F/m$^2$) between the n-well and the substrate in the middle of the resistor. Assume that the resistance does not vary with position along the resistor.

Microchip Ex. 1055, p. 67
Microchip v. Aptiv
IPR2024-00494

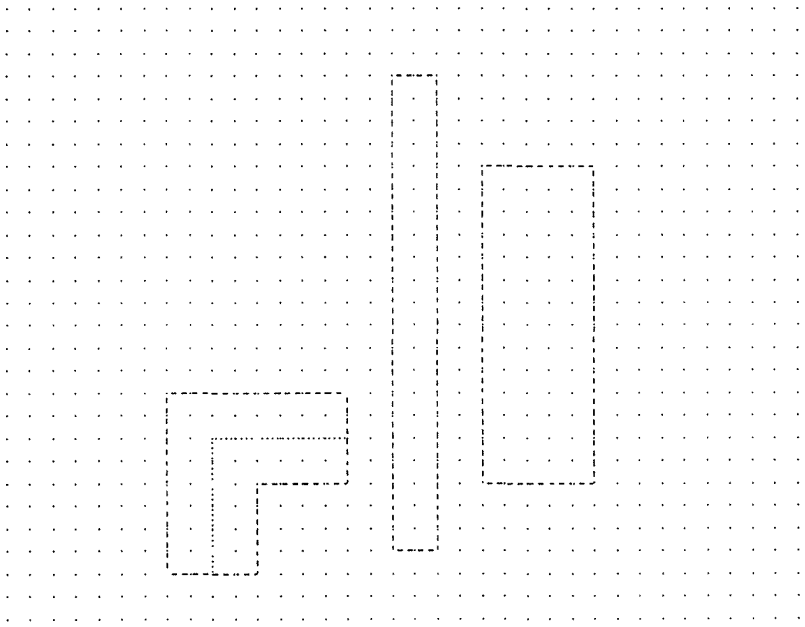


**Figure P2.2**

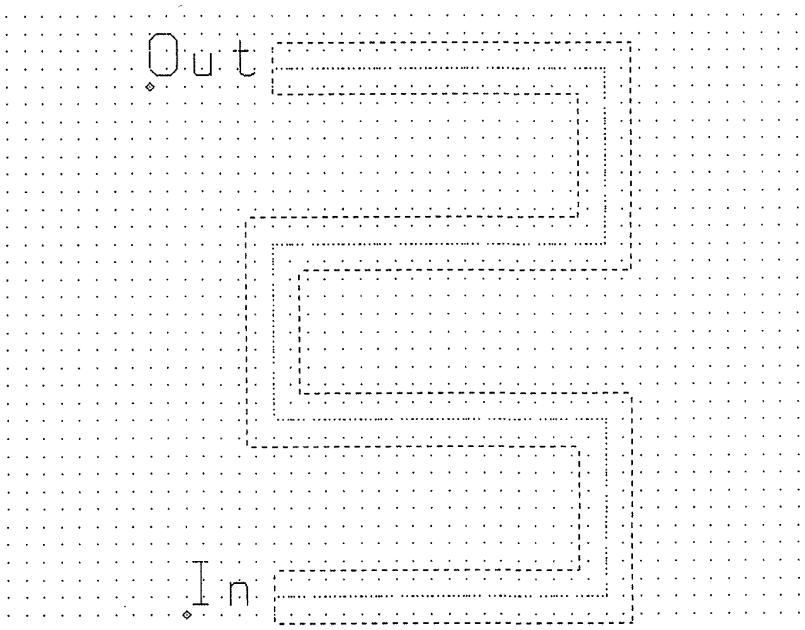



**Figure P2.3**

Microchip Ex. 1055, p. 68
Microchip v. Aptiv
IPR2024-00494

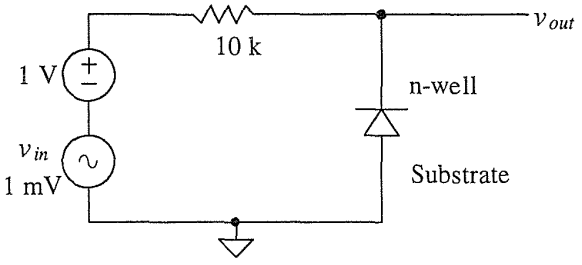



**Figure P2.7**

**2.12** The diode reverse breakdown current, that is, the current that flows when $|V_d| < BV$ (breakdown voltage), is modeled in SPICE by

$$I_D = IBV \cdot e^{-(V_d + BV)/V_T}$$

Assuming 10 μA of current flows when the junction starts to break down at 100 V, simulate, using a SPICE DC sweep, the reverse breakdown characteristics of the diode. (The breakdown voltage, $BV$, is a positive number. When the diode starts to break down $-BV = V_d$. For this diode, breakdown occurs when $V_d = -100$ V.)

**2.13** Repeat Ex. 2.3 if the n-well/p-substrate diode is 50 μm square and the acceptor doping concentration is changed to $10^{15}$ atoms/cm$^3$.

**2.14** Estimate the storage time, that is, the time it takes to remove the stored charge in a diode (see Fig. 2.15), when $\tau_T = 5$ ns, $V_F = 5$ V, $V_R = -5$ V, $C_{j0} = 0.5$ pF, and $R = 1$k. Verify your results with SPICE.

Microchip Ex. 1055, p. 69
Microchip v. Aptiv
IPR2024-00494

Chapter

3

# The Metal Layers

Now that we are familiar with the well, let's discuss the metal layers. In particular, for the CN20 process there are two levels of metal. These levels are named metal1 and metal2. The metal layers are used to connect the circuit together. In this chapter, we look at the bonding pad, the design rules for the metal layers, capacitances associated with the metal layers, crosstalk, sheet resistance, and metal migration.

## 3.1 The Bonding Pad

The bonding pad is at the interface between the die and the package or the outside world. One side of a wire is soldered to the pad, while the other side of the wire is connected to a lead frame. (The lead frame, in part, is the actual pins we see in a packaged integrated circuit.) At this point we will not concern ourselves with electro static discharge (ESD) protection, which is an important design consideration when designing the pad.

### 3.1.1 Laying out the Pad

The basic size of the bonding pad specified by MOSIS is a square 100 μm x 100 μm. For a probe pad[1], used to probe the circuit with a microprobe station, the size should be greater than 6 μm x 6 μm. A pad that uses metal2 is shown in Fig. 3.1. Notice, in the cross-sectional view, the layers of insulator ($SiO_2$ in most cases) under and above the metal2. These layers are used for isolation between the other layers in the CMOS process.

Before proceeding any further, we might ask the question, "What is the capacitance from this metal2 box to the substrate?" This is important because we have

---

[1] The minimum size of a probe pad is set by the minimum overglass size of 6 μm. In general, and if possible, probe pads should be a square with a side measuring 75 μm.

Microchip Ex. 1055, p. 70
Microchip v. Aptiv
IPR2024-00494

to drive this capacitance in order to get a signal off the chip. From the data sheets for the process, the Orbit electrical parameters given in Appendix A, the specification of metal2 to substrate plate capacitance is 13 to 15 aF/μm$^2$. If the measured plate capacitance is 14 aF/μm$^2$, then the capacitance of the metal2 bonding pad to substrate is 0.14 pF. This illustrates that when designing high-performance CMOS digital or analog circuits, the design engineer must be aware of the limitations, in this case the capacitances, inherent in the process being used. Layer 12, in the CN20 setups, is used to specify the metal2 layer. The metal layers are sometimes referred to as wires because these layers are used to connect circuits, resistors, MOSFETs, and capacitors, together.

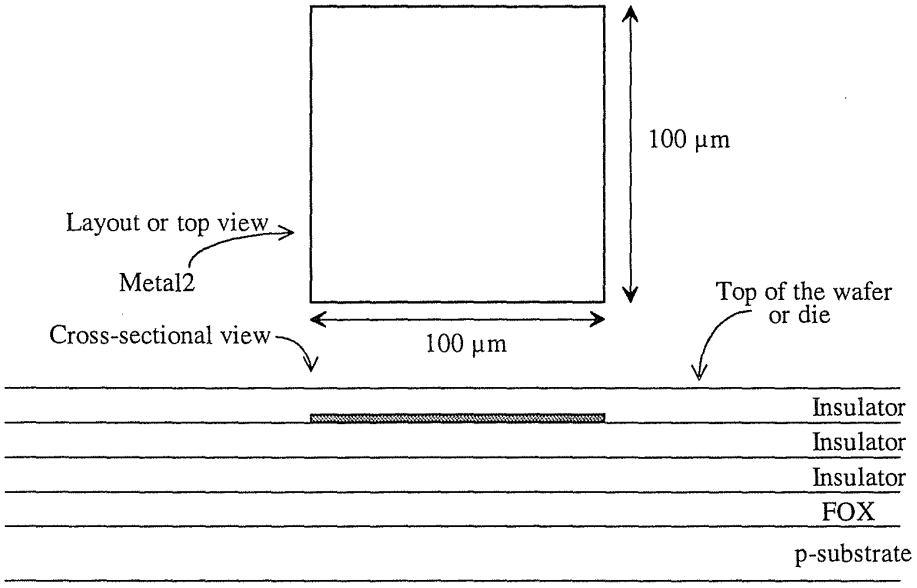



**Figure 3.1** Layout of metal2 used for bonding pad with associated cross-sectional view.

Since an insulator is covering the pad, we cannot bond (connect a wire) to the pad. To specify an opening or cut in the glass, we use the PAD or overglass layer, LASI layer 13. The top layer insulator on the chip is also called passivation. The passivation helps protect the chip from contamination. Openings for bonding pads are called cuts in the passivation. Orbit specifies 5 microns between the edge of metal2 and the PAD layer. A complete pad using metal2 with the pad layer is shown in Fig. 3.2.

Often metal1, layer 10, is placed under metal2 when laying out the pads. This is so either metal layer from the circuit can be connected to the pad. Metal2 is connected to metal1 by the layer via, layer 11. On die the only layer that can be connected to metal2 is metal1, the layer directly below metal2. Metal1 can connect to metal2, poly, or active (n+ or p+). We will discuss this in more detail in the next chapter. The via must be at least 3 μm inside metal2. A bonding pad with both metal1 and 2 is shown in Fig. 3.3. Notice how the via has the effect of removing the glass under metal2. When

Microchip Ex. 1055, p. 71
Microchip v. Aptiv
IPR2024-00494

metal2 is deposited on the wafer, it will make physical contact with metal1 where there is a via. Figure 3.4 shows an expanded corner view of a pad using both metals1 and 2 drawn with LASI.

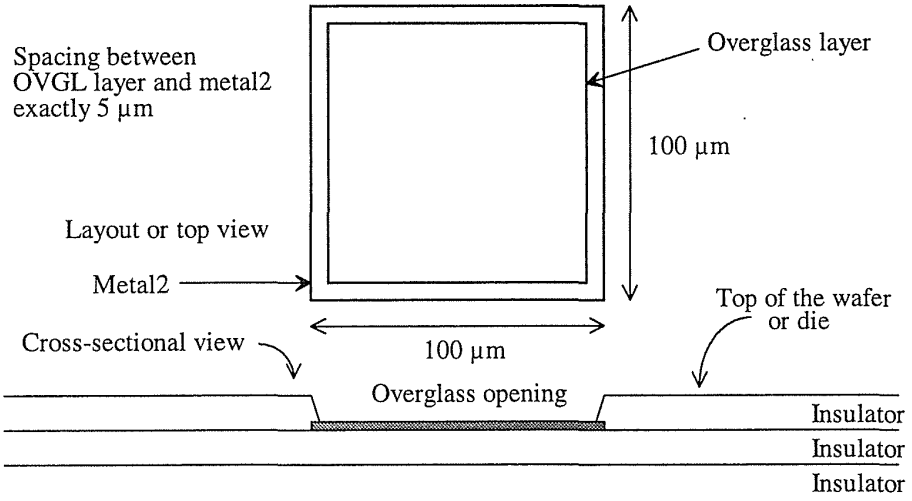



**Figure 3.2** Layout of a metal2 pad with pad opening for bonding connection.

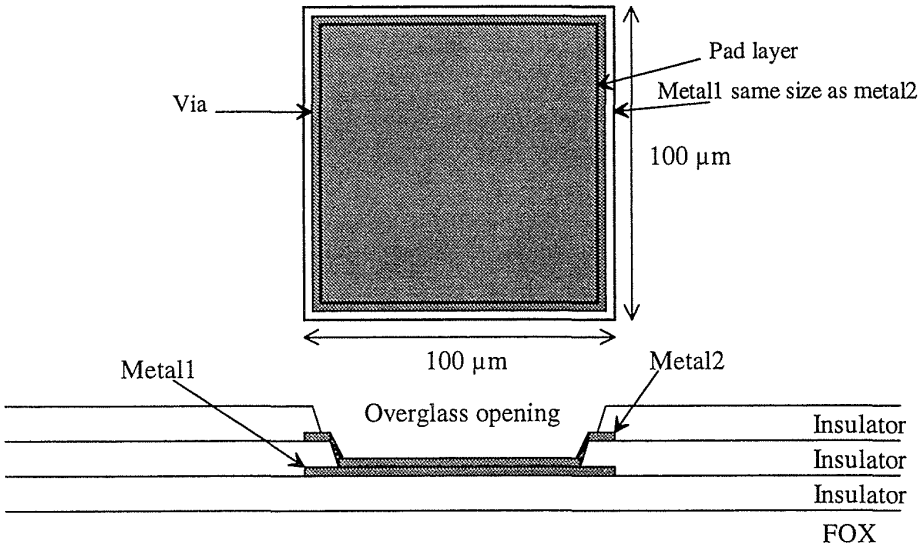



**Figure 3.3** A bonding pad using metal1 and metal2.

Microchip Ex. 1055, p. 72
Microchip v. Aptiv
IPR2024-00494

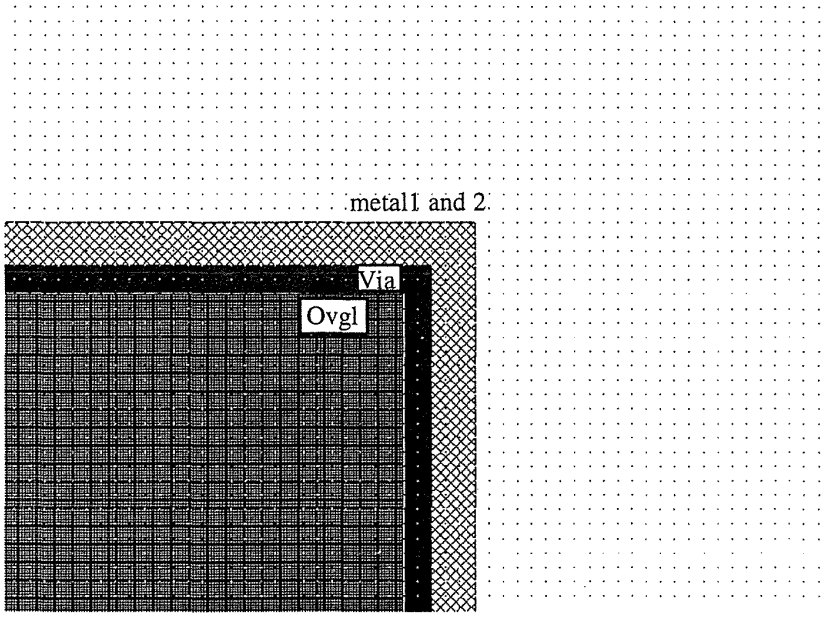



**Figure 3.4** Expanded corner view of a pad.

### 3.1.2 Design Rules for Pads

Figure 3.5 shows the design rules for the bonding pads. The bonding pad size we will use is 100 μm by 100 μm. The ovgl layer (cuts in the passivation) should be a square, 90 μm on a side, centered in the pad, that is, 5 μm from the metal edge. The pad-to-pad spacing must be at least 75 μm. Note that the bonding pad design rules are not checked by LASIDRC.

---

**Example 3.1**
Lay out a 40-pad frame for a MOSIS tiny chip.

The size of the tiny chip, from Ch. 1, is 2.2 mm by 2.2 mm. If we assume that ten pads are on each of the four sides and the corners are open, that is, contain no pads (see Fig. 3.6), then we can divide the length of a side by the sum of the number of pads and the corner areas to get the pad cell size, or

$$\text{Cell size} = \frac{2.2 \times 10^{-3}}{12} = 183\ \mu\text{m} \overset{\text{rounded to}}{\rightarrow} 180\ \mu\text{m}$$

The chip, or die size, that we will have using a pad cell size of 180 μm is 2160 μm by 2160 μm. Figure 3.7 shows the pad placed in a box, made using the outline layer (layer 58), of 180 μm square. The outline layer has no effect on fabrication. This layer is used to help align the cells when we place them in a higher ranking cell. Figure 3.6 shows the entire padframe. ■

Microchip Ex. 1055, p. 73
Microchip v. Aptiv
IPR2024-00494

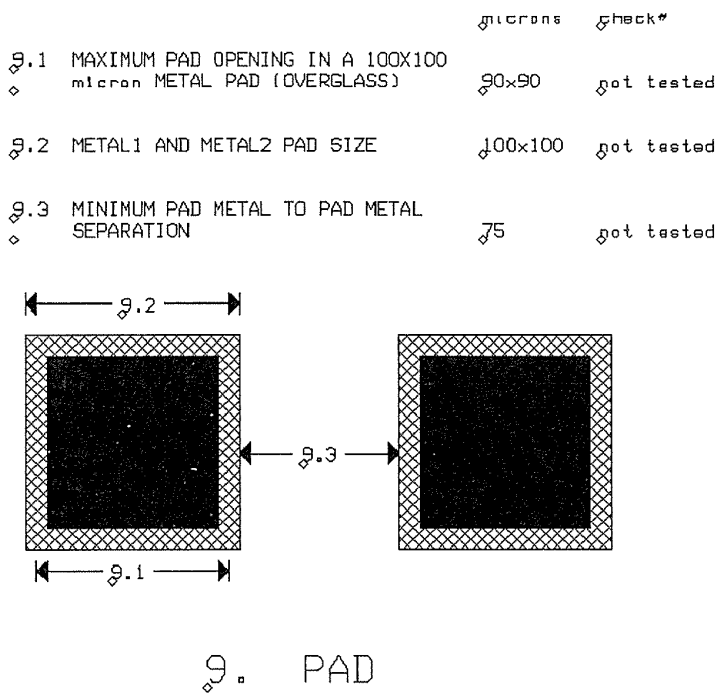



**Figure 3.5** Design rules for the bonding pads.

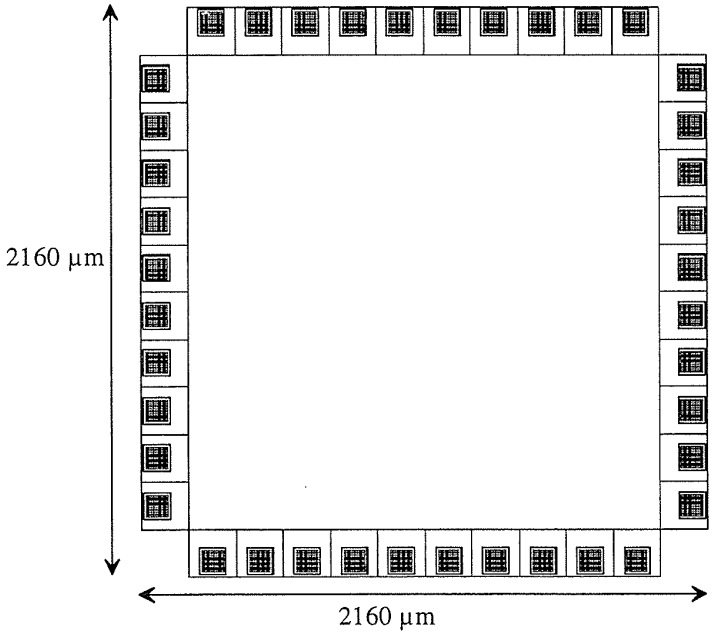



**Figure 3.6** Layout of a 40-pad padframe. Final dimensions 2160 µm by 2160 µm.

Microchip Ex. 1055, p. 74
Microchip v. Aptiv
IPR2024-00494

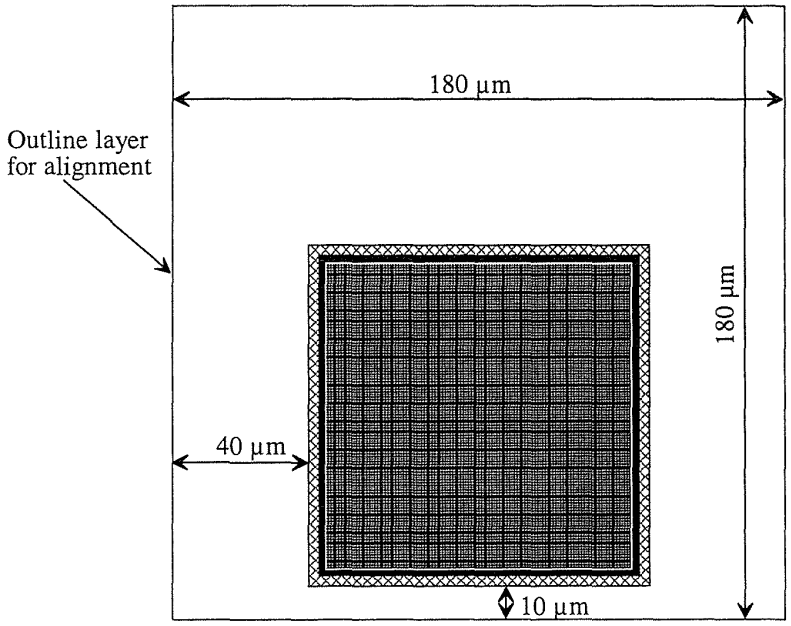



**Figure 3.7** Layout of a pad using the outline layer.

## 3.2 Design and Layout Using the Metal Layers

As mentioned earlier, the metal layers are used to connect the resistors, capacitors, and MOSFETs in a CMOS integrated circuit. We had a taste of the metal1 and 2 layers in the last section. In this section, we begin by discussing the design rules for the metal layers. The parasitic resistances and capacitances of these metal layers are then discussed.

### 3.2.1 Design Rules for the Metal Layers

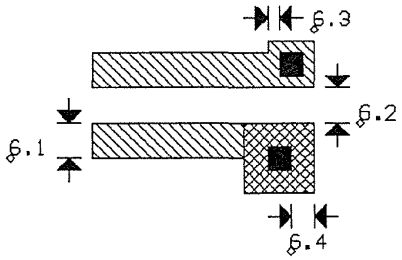
The metal and via design rules are shown in Figs. 3.8 through 3.10. The complete set of design rules is given in Appendix A. The minimum width and spacing of metals1 and 2 is 3 µm. Also shown in these figures is the contact layer. This layer connects metal1 to p+, n+ or the poly layers. At this point, we will concern ourselves with metal1, metal2, and the via layers.

### 3.2.2 Parasitics Associated with the Metal Layers

The basic parasitic resistances and capacitances associated with the metal layers can be calculated from the information given in Appendix A. The main parameters we are interested in are the sheet resistances of the layers, the capacitance between the metal layers and active, poly, substrate, and between one another. Also, there is a finite contact resistance of the via. The following examples illustrate some of the unwanted parasitics associated with these layers.

Microchip Ex. 1055, p. 75
Microchip v. Aptiv
IPR2024-00494

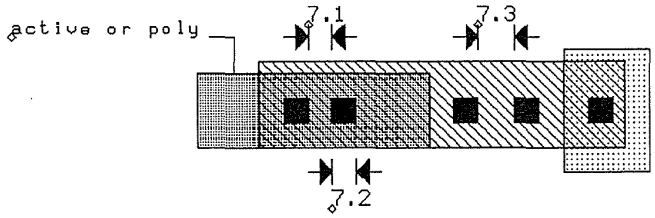
| | | microns | check* |
|---|---|---|---|
| 6.1 | WIDTH | 3.0 | 23 |
| 6.2 | SPACING | 3.0 | 24 |
| 6.3 | OVERLAP OF CONTACT | 1.0 | 25 |
| 6.4 | OVERLAP OF VIA | 2.0 | 26 |




6. METAL 1

**Figure 3.8** Metal1 design rules.

| | | microns | check* |
|---|---|---|---|
| 7.1 | SPACE TO CONTACT | 2.0 | 27 |
| 7.2 | SIZE (EXCEPT FOR PADS) | 2.0x2.0 | 28 |
| 7.3 | SPACING | 3.0 | 29 |

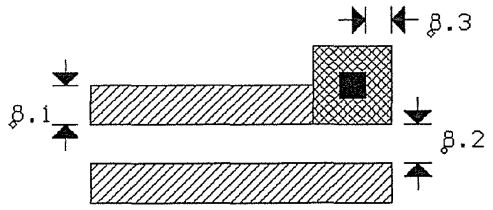



7. VIA

**Figure 3.9** Via design rules.

Microchip Ex. 1055, p. 76
Microchip v. Aptiv
IPR2024-00494

| | | microns | check# |
|---|---|---|---|
| 8.1 | WIDTH | 3.0 | 30 |
| 8.2 | SPACE | 3.0 | 31 |
| 8.3 | METAL2 OVERLAP OF VIA | 2.0 | 32 |




8. METAL 2

**Figure 3.10** Metal2 design rules.

**Example 3.2**

Estimate the resistance of a 1 mm long piece of metal1 with a width of 3 μm. Also estimate the delay through this piece of metal, treating the metal line as an RC transmission line. Assume that no other layers are present under the metal run.

First, we need to calculate the resistance of the metal line. From Appendix A we see that the typical sheet resistance of metal1 is 0.06 Ω/square. The overall resistance of the line is then given by

$$R = 0.06 \cdot \frac{1{,}000\ \mu\text{m}}{3\ \mu\text{m}} = 20\ \Omega$$

The capacitance to substrate (we use this value on the data sheets because there are no other layers under metal1 in this example, and we are neglecting the fringing capacitance) is given in the data sheets as 26 aF/μm² (max). The overall capacitance of the line is then

$$C = \frac{26 \times 10^{-18} F}{\mu\text{m}^2} \cdot (1{,}000\ \mu\text{m}) \cdot (3\ \mu\text{m}) = 78\ \text{fF}$$

If we treat the metal line as an RC transmission line, the delay can be estimated as $0.35RC$ or 0.55 ps, a negligible and unrealistic delay. In general, unloaded

Microchip Ex. 1055, p. 77
Microchip v. Aptiv
IPR2024-00494

metal lines must include inductive effects. An unloaded metal line exhibits a delay of 5.4 ps/mm [2]. ■

**Example 3.3**

Estimate the capacitance between a 10 μm square piece of metal1 and an equal-size piece of metal2 placed exactly above the metal1 piece. Sketch the layout and the cross-sectional view. Also sketch the symbol of a capacitor on the cross-sectional view.

The plate capacitance, from Appendix A, between metal1 and metal2 is at most 38 aF/μm², while the fringe capacitance is at most 104 aF/μm. The two layers form a parallel plate capacitor. The capacitance between the plates is given by the sum of the plate capacitance and the fringe capacitance, or

$$C_{12} = \frac{38\times10^{-18}\text{F}}{\mu\text{m}^2}\cdot\left[\overbrace{10\ \mu\text{m}}^{area}\right]^2 + \frac{104\times10^{-18}\text{F}}{\mu\text{m}}\cdot\overbrace{(40\ \mu\text{m})}^{perimeter} = 8\ \text{fF}$$

The layout and cross-sectional view are shown in Fig. 3.11. ■

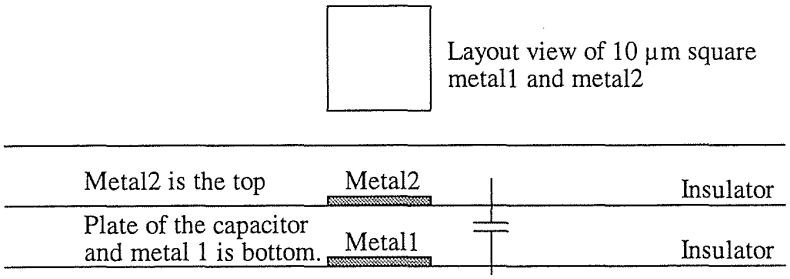



**Figure 3.11** Capacitance between metal1 and metal2.

**Example 3.4**

In the previous example, estimate the voltage change on metal1 when metal2 changes potential from 0 to 5 V.

The capacitance from metal2 to metal1 was calculated as 8 fF. The capacitance from metal1 to substrate is given by

$$C_{1sub} = \frac{26\times10^{-18}F}{\mu\text{m}^2}\cdot\left[\overbrace{10\ \mu\text{m}}^{area}\right]^2 + \frac{82\times10^{-18}\text{F}}{\mu\text{m}}\cdot\overbrace{(40\ \mu\text{m})}^{perimeter} = 5.9\ \text{fF}$$

The equivalent schematic is shown in Fig. 3.12. The voltage on $C_{1sub}$ is given by

Microchip Ex. 1055, p. 78
Microchip v. Aptiv
IPR2024-00494

$$\Delta V_{metal1} = 5 \cdot \frac{\frac{1}{j\omega C_{1sub}}}{\frac{1}{j\omega C_{1sub}} + \frac{1}{j\omega C_{12}}} = 5 \cdot \frac{C_{12}}{C_{12} + C_{1sub}} = 5 \cdot \frac{8}{8+5.9} = 2.9 \text{ V}$$

A displacement current flows through the capacitors, causing the potential on metal1 to change 2.9 V. This may seem significant at first glance. However, one must remember that most metal lines in a CMOS circuit are being driven from a low-impedance source; that is, the metal is not floating but is being held at some potential. This is not the case in some dynamic circuits or in circuits with high-impedance nodes or long metal runs. ■

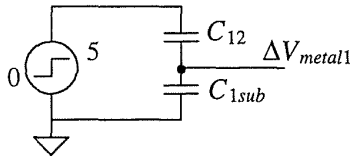



**Figure 3.12** Equivalent circuit used to calculate the change in metal1 voltage.

### 3.2.3 Current Carrying Limitations

Now that we have some familiarity with the metal layers, we need to answer the question, "How much current can we carry on a given width or length of metal?" The factors that limit the amount of current on a metal wire or bus are metal electromigration, and the maximum voltage drop across the wire or buss due to the resistance of the metal layer.

Metal electromigration results from a conductor carrying too much current. This effect is similar to the erosion that occurs when a river carries too much water. The result is a change in the conductor dimensions, causing spots of higher resistance and eventually failure. If the current density is kept below the metal migration threshold current density, $J_{Al}$, metal migration will not occur. Typically, for aluminum, which is what metal1 and metal2 are made of, the current threshold for migration $J_{Al}$ is $1 \rightarrow 2 \frac{\text{mA}}{\mu\text{m}}$.

---

**Example 3.5**

Estimate the maximum current a piece of metal1 3 μm wide can carry. Also estimate the maximum current a bonding pad can receive from a bonding wire.

Assuming that $J_{AL} = 1 \frac{\text{mA}}{\mu\text{m}}$, the maximum current on a 3 μm wide aluminum conductor is given by

$$I_{max} = J_{Al} \cdot W = 10^{-3} \cdot 3 = 3 \text{ mA}$$

The maximum current through a bonding pad is then 100 mA. ■

Microchip Ex. 1055, p. 79
Microchip v. Aptiv
IPR2024-00494

**Example 3.6**

Estimate the voltage drop across the conductor discussed in the previous example when the length of the conductor is 1 cm and the current flowing in the conductor is 3 mA ($I_{max}$).

The sheet resistance of metal1 is 0.06 Ω/square. The voltage drop across a metal1 wire that is 3 μm wide and 10,000 μm (1 cm) long carrying 3 mA is

$$V_{drop} = (0.06\Omega/\text{square}) \cdot \frac{10,000}{3} \cdot 3 \text{ mA} = 0.6 \text{ V}$$

or a significant voltage drop. If this conductor was used for power, we would want to increase the width significantly; however, if the conductor is used to route data, the size may be fine. ■

In general, metal2 should be used for power routing. Metal2 is approximately twice as thick as metal1 and therefore has a lower sheet resistance. When routing power the more metal that is used the fewer problems, in general, that will be encountered. If possible, a ground or power plane should be used across the entire die. The more capacitance between the power and ground busses, the harder it is to induce a voltage change on the power plane; that is, the DC voltages will not vary.

### 3.2.4 Parasitics Associated with the Via

The via layer is used to make a connection between metal1 and metal2. The size of a via is a square with a side measuring exactly 2 μm (except for the bonding pad). The via layer specifies where to remove glass so that when metal2 is deposited it makes contact with metal1. Associated with this connection is a contact resistance. This resistance arises because the metal2 thins as it is deposited in the hole or as a result of the contact potential difference between two different materials. Minimum contact resistance for a via, given in Table A.8 in Appendix A, is 0.05 Ω, while the maximum contact resistance is 0.08 Ω. Also associated with the via is a metal migration limitation of typically 0.4 mA/contact. Because of the thinning of the metal2, the metal migration threshold becomes 0.05 mA/μm of via perimeter.

**Example 3.7**

Sketch the equivalent schematic for the layout depicted in Fig. 3.13a showing the via contact resistance. Estimate the voltage drop across the contact resistance of the via when 10 mA flows through the via. Comment on the reliability of the via. Suggest a method, by changing the layout, of lowering the effective contact resistance and current density through the via. What is the maximum current that can flow from metal2 to metal1 after the layout change?

The equivalent schematic is in Fig. Ex3.7. If the via contact resistance is 0.08 Ω and 10 mA flows through the via, a voltage drop of 0.8 mV will result. The reliability of the single via is poor. The maximum current that should flow through the via is 0.4 mA. Forcing 10 mA through the via will cause the via to

Microchip Ex. 1055, p. 80
Microchip v. Aptiv
IPR2024-00494

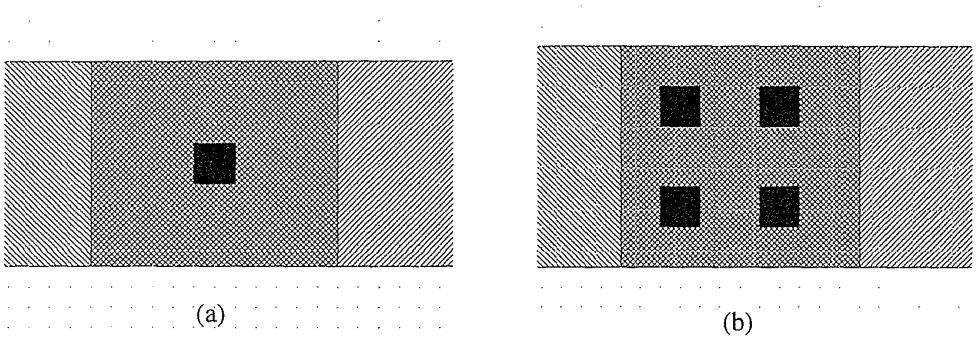



**Figure 3.13** Layouts used in Example 3.7.

eventually open or become a high-resistance connection. Figure 3.13b shows that adding as many vias as the design rules will allow lowers the contact resistance. The four vias shown in this figure give an effective contact resistance of 0.08/4 or 0.02 Ω due to the contact resistances of each of the vias being in parallel. The maximum current, set by metal electromigration considerations, through the four contacts is 1.6 mA. Increasing the overlap and the number of vias will increase the maximum current to a limitation set by the width of the metal lines. ■

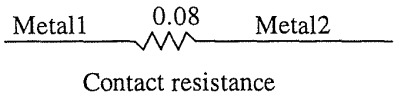



**Figure Ex3.7**

## 3.3 Crosstalk and Ground Bounce

Crosstalk is a term used to describe an unwanted interference from one conductor to another. Between two conductors there exists mutual capacitance and inductance, which give rise to signal feedthrough.

Consider the two metal wires shown in Fig. 3.14. A signal voltage propagating on one of the conductors couples current onto the conductor. This current can be estimated using

$$I_m = C_m \frac{dV_A}{dt} \tag{3.1}$$

where $C_m$ is the mutual capacitance, $I_M$ is the coupled current, and $V_A$ is the signal voltage on the source conductor [2]. Treating the capacitance between the two conductors in this simple manner is useful in most cases. If the coupled noise voltage is less than 10 percent of signal voltage, this approximation is accurate to one decimal

Microchip Ex. 1055, p. 81
Microchip v. Aptiv
IPR2024-00494

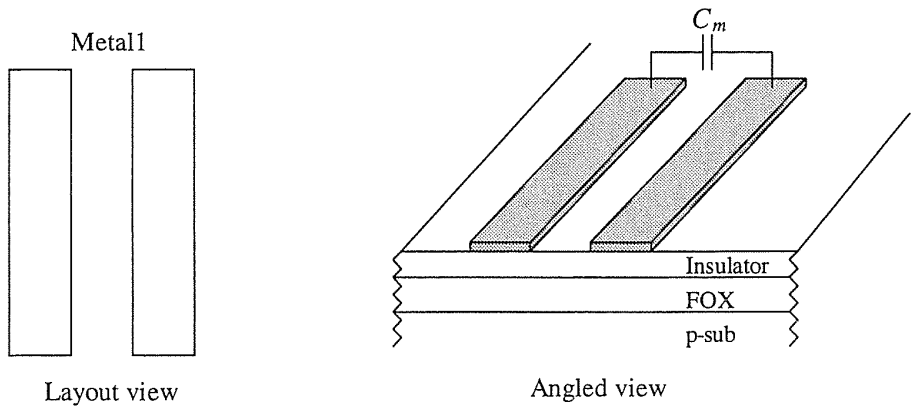



**Figure 3.14** Conductors used to illustrate crosstalk.

place [2]. Determining $C_m$ experimentally proceeds by applying a step voltage to one conductor while measuring the coupled voltage on the adjacent conductor. Since we know the capacitance of any conductor to substrate (see Appendix A) we can write

$$\Delta V = V_A \cdot \frac{C_m}{C_m + C_{1s}} \tag{3.2}$$

where $\Delta V$ is the coupled noise voltage to the adjacent conductor and $C_{1s}$ is the capacitance of the adjacent conductor to ground.

The adjacent metal lines shown in Fig. 3.14 also exhibit a mutual inductance. The effect can be thought of as connecting a miniature transformer between the two conductors. A current flowing on one of the conductors induces a voltage on the other conductor. Measuring the mutual inductance begins by injecting a current into one of the conductors. The voltage on the other conductor is measured. The mutual inductance is determined using

$$V_m = L_m \frac{dI_A}{dt} \tag{3.3}$$

where $I_A$ is the injected current, $V_m$ is the induced voltage, and $L_m$ is the mutual inductance.

At the risk of stating the obvious, crosstalk can be reduced by increasing the distance between adjacent conductors. In many applications (e.g., DRAM), the design engineer has no control over the spacing (pitch) between conductors.

### 3.3.1 Ground Bounce

Consider the single metal wire shown in Fig. 3.15. The inductance of this wire can be estimated [2], assuming $w > h$, using

$$L\ (\text{nH/mm}) = \frac{1.25}{\frac{w}{h} + 1.393 + 0.667 \cdot \ln\left[\frac{w}{h} + 1.44\right]}\ (\text{nH/mm}) \tag{3.4}$$

Microchip Ex. 1055, p. 82
Microchip v. Aptiv
IPR2024-00494

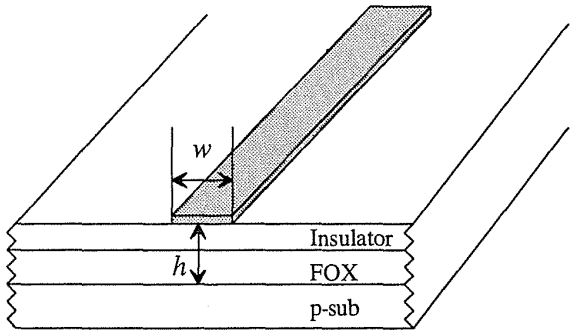



**Figure 3.15** Conductors used in calculation of inductance.

Here we are assuming that the thickness of the wire is small compared to the width and that the p-substrate acts like a constant potential or reference ground plane. For the CN20 process, Appendix A, the thickness between metal1 and substrate is 1.5 μm. For a 3 μm wide piece of metal1, the inductance is

$$L\ (\text{nH/mm}) = \frac{1.25}{\frac{3}{1.5} + 1.393 + 0.667 \cdot \ln\left(\frac{3}{1.5} + 1.44\right)} = 0.3\ \text{nH/mm}$$

Next consider the circuit shown in Fig. 3.16. A circuit is powered through two wires made using metal1, each 10 mm long and 3 μm wide. Ideally, point A in the figure is held at *VDD*, while point B is fixed at ground potential. Neglecting resistive losses, we find this is true when *I* is constant. However, if *I* changes, the lengths of metal behave as inductors.

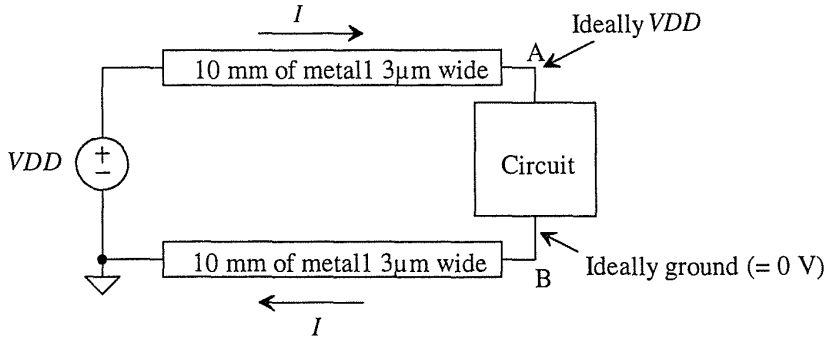



**Figure 3.16** Block diagram used to illustrate ground bounce.

Consider a circuit that sources 50 mA in 3 ns (a common requirement in an output driver). The inductance of either metal wire is 3 nH. The change in voltage at point B (keeping in mind point B is supposed to be "ground") is given by

Microchip Ex. 1055, p. 83
Microchip v. Aptiv
IPR2024-00494

$$V_B = L\frac{dI}{dt} = 3\text{ nH} \cdot \frac{50\text{ mA}}{3\text{ ns}} = 50\text{ mV}$$

This means the "ground" of the circuit actually jumps 50 mV when the circuit needs this current. This noise or ground bounce generated on the ground conductor can feed into other circuits on the chip and cause problems. What's worse is when ten circuits on chip pull this current through the same conductor at the same time.

Ground bounce can be reduced by increasing the width, and thus decreasing the inductance, of the conductors supplying power to a circuit (the wider the metal the better). Also, increasing the capacitance between the conductor supplying current and the conductor returning current helps reduce ground bounce. The simplest method of increasing the capacitance is to lay the conductors out side by side. An on-chip decoupling capacitor between the circuit *VDD* and ground will help as well.

In the previous analysis, we neglected the resistance of the metal wires. In most practical situations, the inductance of the wire is negligible. The ground bounce is dominated by the resistive drop across the wire. Calculating the ground bounce due to the resistive component in Fig. 3.16 begins by calculating the resistance of a wire. The sheet resistance of metal1 is 0.06 Ω/square. The resistance of the wire is 200 Ω. When $I = 0$, the potential at point B is 0. However, when $I = 5$ mA, the potential at point B is 1 V! Again increasing the width of the conductors, and thus decreasing the resistance, helps to reduce the ground bounce.

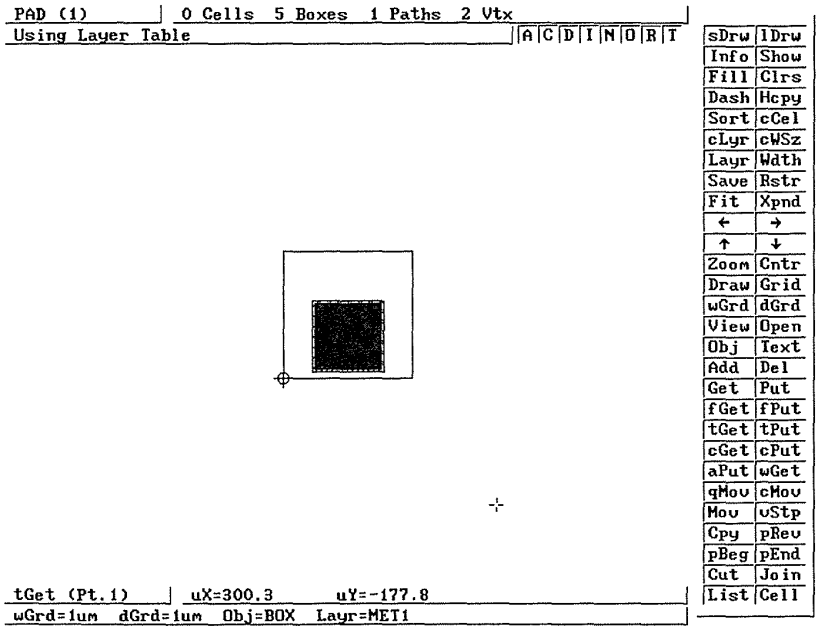
## 3.4 Layout Using Cell Hierarchy

*Do not lay out a chip unless the material in this section is understood!*

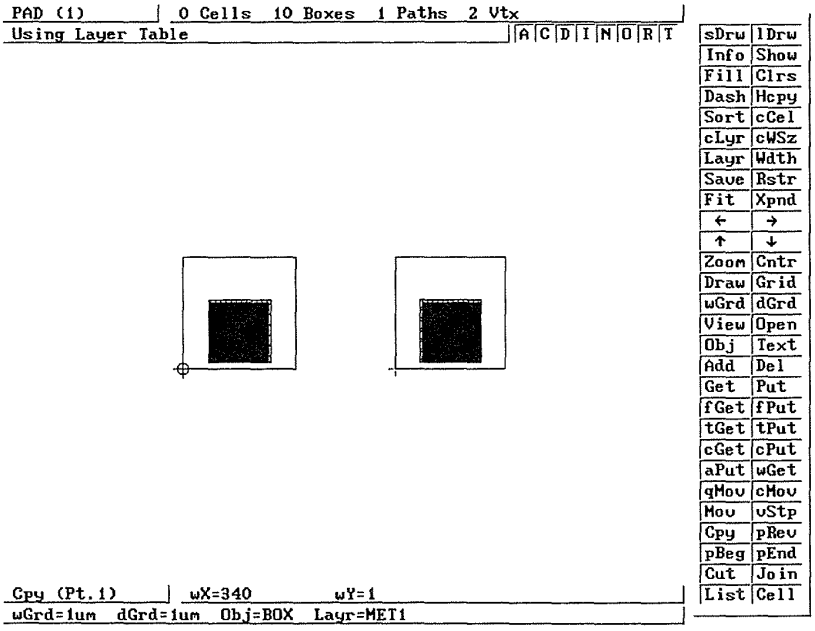
Using cell hierarchy, that is, ranking and nesting of cells, is important to keep the size of the design file (the final GDS file sent to the fab-house) from getting too large. Consider the pad cell shown in Fig. 3.5. Let's assume this cell is called PAD and has a rank of 1. We can generate the layout of Fig. 3.6 in two ways: the right way and the wrong way. Let's consider the wrong way first. Working within the pad cell shown in Fig. 3.17a, use the copy command, **Cpy**, to copy the pad layout Fig. 3.17b, until the number of pads needed is copied into the cell. This is the wrong way to do layout. Each time the cell is copied, boxes or polygons are generated at the new location in the cell, resulting in a significant increase in the size of the design file. In addition, a change in the pad, for example, changing the size of metal2 or adding objects to the cell, must be made to each of the copied pads. This can result in a waste of time. The **Cpy** command should be used as little as possible when laying out a chip.

The correct procedure for laying out the padframe of Fig. 3.6 begins by making a cell called PADFRAME with a rank of 2 using the **Cell** command button. Once in the padframe cell, the object is changed to PAD, the cell we want to use 40 times in the padframe cell using the **Obj** command button. Notice that the PAD cell has a lower ranking than the PADFRAME cell. Figure 3.18 shows use of the **Add** command to add the PAD cell to the PADFRAME cell. Each time the PAD cell is added to the

Microchip Ex. 1055, p. 84
Microchip v. Aptiv
IPR2024-00494




(a)




(b)

**Figure 3.17** How not to lay out a padframe, (a) basic pad cell with a rank of 1 and (b) using the Cpy command to lay out the padframe of Fig. 3.6.

Microchip Ex. 1055, p. 85
Microchip v. Aptiv
IPR2024-00494

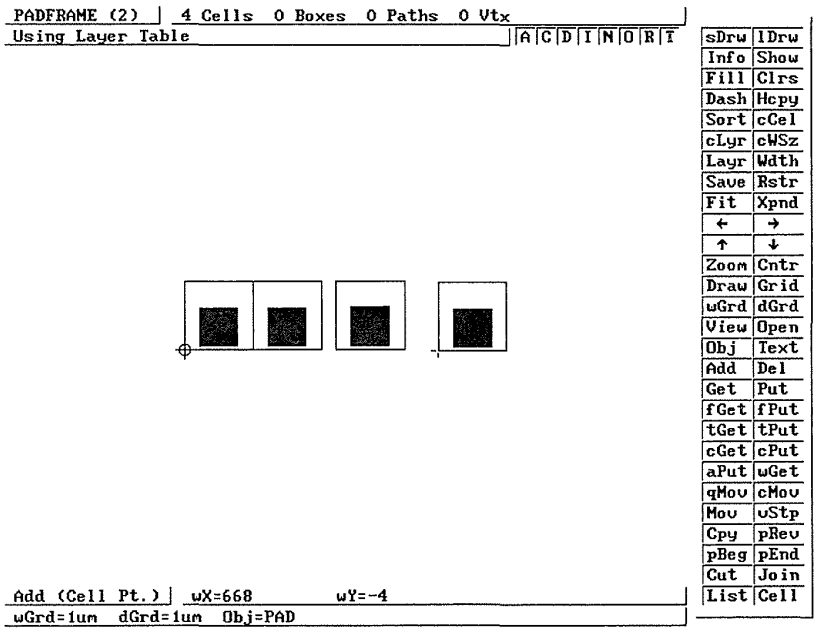



**Figure 3.18** Correct method of laying out the PADFRAME cell.

PADFRAME cell a single specifying line, used to reference the added cell and the location, is added to the design file. The result is a significantly smaller design file. Also, if a change is made to the PAD cell, the change is automatically propagated through the hierarchy to the other higher ranking cells using the PAD cell. Figure 3.19 shows the cells of Fig. 3.18 drawn in outlines using the **Outl** command. This can speed up the redraw time.

Cell hierarchy can be extended to every aspect of layout. For example, we know that the via size, used in the CN20 process, is exactly 2 μm by 2 μm square. Instead of drawing a via this size each time we need to make a connection between metal1 and metal2, we can lay out a cell called VIA with a rank of 1 that is a box on the via layer measuring 2 μm by 2 μm square. Now each time we need to make a connection between metal1 and metal2 we simply use the via cell as an object (time is saved since the size of the via is fixed in the VIA cell). The same idea can be used for the active and poly contacts that will be discussed in the next chapter.

When designing VLSI digital circuits, it is common to use minimum-size MOSFETs. A minimum-size MOSFET can be laid out in a cell called NMIN with a rank of one. Each time a MOSFET is needed in a circuit, the NMIN cell can be used. This idea can and should also be used in analog VLSI circuits. Note that an added cell cannot be edited from a higher ranking cell. For example, to edit the PAD cell of Fig. 3.18 we would first have to leave the PADFRAME cell and return to the PAD cell.

Microchip Ex. 1055, p. 86
Microchip v. Aptiv
IPR2024-00494

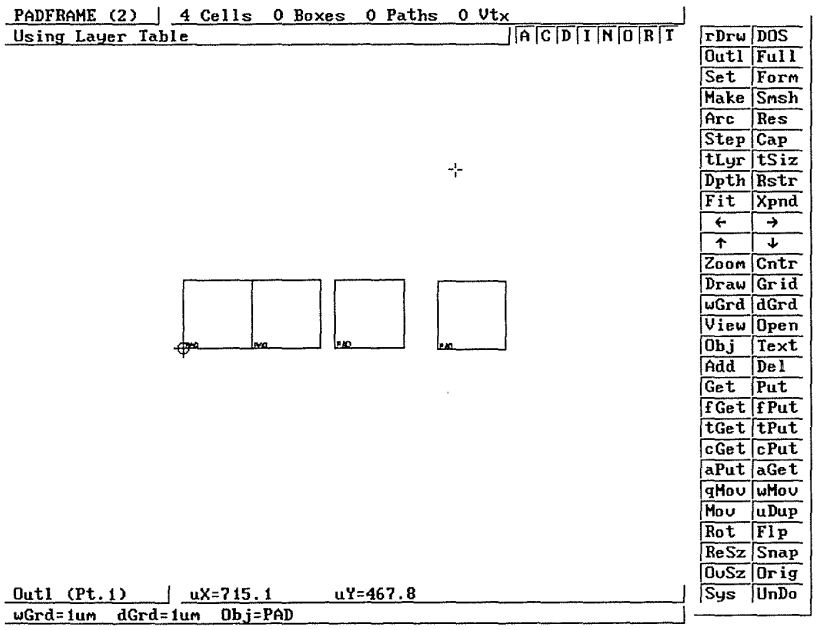



**Figure 3.19** Drawing cells in the outline mode can speed up redraw time.

## REFERENCES

[1] W. Tanner, *MOSIS User Manual,* Release 4.0, August 1994.

[2] H. W. Johnson and M. Graham, *High Speed Digital Design: A Handbook of Black Magic,* Prentice-Hall Publishing Company, 1993. ISBN 0-13-395724-1.

## PROBLEMS

**3.1** Lay out and DRC the pad given in Fig. 3.5. Also lay out the padframe shown in Fig. 3.6. The pad cell should be a rank of 1. The padframe should have a rank of 2. Use unique names for the cells. Keep these cells backed up. We will add ESD protection in the next chapter. This is basically the beginnings of the chip you will submit to MOSIS.

**3.2** Sketch the cross-sectional view of the layout shown in Fig. P3.2.

**3.3** Sketch the cross-sectional view across the center of the pad in Fig. 3.5 without the via layer present. Does the outline layer, layer 58, have any fabrication significance? Why would the outline layer be used?

**3.4** What would be the result of submitting a chip without the OVGL layer drawn in the pad cell?

**3.5** Explain how to label the pin numbers on the padframe cell of Problem 1 using the **Text** command. Use a Text size of at least 20 μm.

Microchip Ex. 1055, p. 87
Microchip v. Aptiv
IPR2024-00494

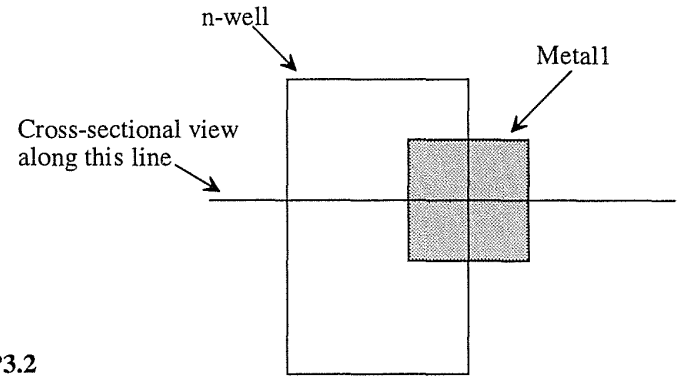



**Figure P3.2**

**3.6** Estimate the change in voltage on metal1 from a change in voltage on metal2 directly above and the same size. Assume the change in metal2 is 1 V. Consider only the plate capacitance, not the fringe capacitance.

**3.7** What is the maximum current that a 5 μm wide piece of metal2 can carry? How many vias will be needed to connect this piece of metal2 to metal1 at the maximum current?

**3.8** What is the minimum width of a metal2 wire used to supply 20 mA of current to a circuit? How many vias are needed to connect the metal2 wire to a metal1 wire?

**3.9** Estimate the inductance of a 4 μm wide piece of metal2 in nH/mm.

**3.10** Sketch the cross-sectional views of the layouts shown in Fig. P3.10 at the three locations.

**3.11** Estimate the voltage change on metal1, in the layout of Fig. P3.11, if metal2's potential changes from 0 to 5 V.

Microchip Ex. 1055, p. 88
Microchip v. Aptiv
IPR2024-00494

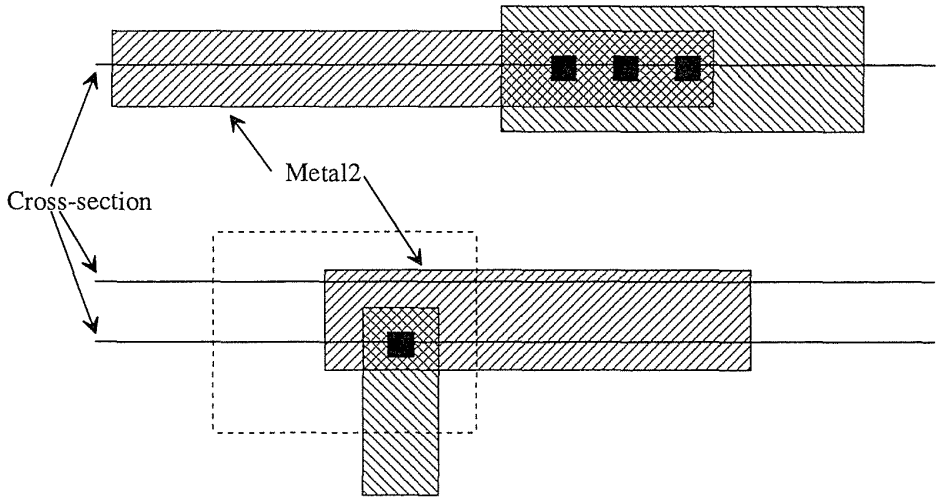



**Figure P3.10**

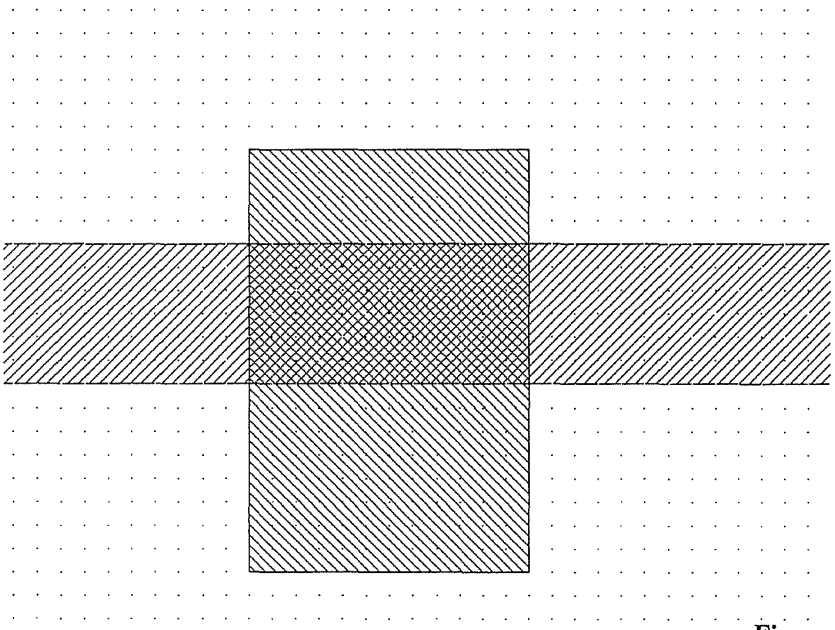


**Figure P3.11**

Microchip Ex. 1055, p. 89
Microchip v. Aptiv
IPR2024-00494

Chapter

4

# The Active and Poly Layers

In this chapter we discuss the active layers, both n+ and p+, and the poly layers. The active layers are used to make the source and drain regions of the MOSFETs. The active layers are also used to connect metal1 to the substrate or the well.

The poly layer is used to form the gate of the MOSFETs. Poly is a short name for polysilicon. Polysilicon is made up of small crystalline regions of silicon. Therefore, in the strictest sense poly is not amorphous silicon,[1] and it is not crystalline silicon such as the wafer.

The poly layer can also be used to connect MOSFETs together. The main limitation when using the poly layer for interconnection is its sheet resistance. As we saw in the last chapter, the sheet resistance of the metal layers is less than 0.1 ohms/square. The sheet resistance of the doped poly can be on the order of 20 Ω/square. The capacitance to substrate is also larger for poly simply because it is closer to the substrate. Therefore, the delay through a poly line can be considerably longer than the delay through a metal line.

The first section of this chapter develops the design rules for the active, poly1, poly2, and contact layers. We use these design rules to lay out a standard cell frame. The standard cell frame allows us to have a fixed area to lay out the n- and p-channel transistors without worrying about power supply routing and substrate/well tie downs. Circuits designed using the standard cell frame can be placed into another cell end to end. This makes layout easier; however, the cost for this is a larger layout. For the most area-efficient use of the silicon, full custom design must be used.

The remaining sections of this chapter discuss the fabrication sequence used to form n+, p+, and poly layers. The chapter ends with a discussion of parasitics.

---

[1] Amorphous silicon is made up of randomly organized Si atoms, while crystalline Si is formed by silicon atoms organized in an orderly fashion.

Microchip Ex. 1055, p. 90
Microchip v. Aptiv
IPR2024-00494

## 4.1 Design Rules

Before we start discussing the design rules for the active area, let's discuss an example layout. Consider the actual layout of an n-well resistor shown in Fig. 4.1. When we discussed the layout of the n-well resistor before, we did not discuss how to make contact to the outside world. In Fig. 4.1 we see the basic n-well resistor. We also see an n+ (heavily doped n-type) region in the n-well. This is used to make an "ohmic" contact to metal1. If metal1 is connected directly to the n-well, a rectifying or schottky diode is formed. These metal1 leads can be connected to the outside world through the pads.

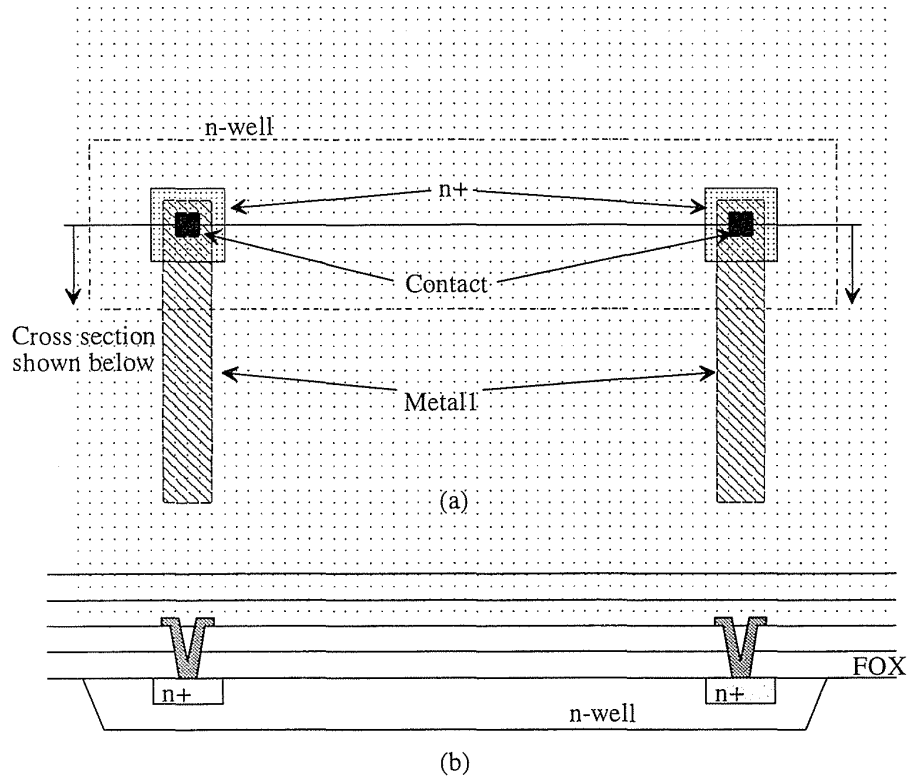



**Figure 4.1** Layout of an n-well resistor with active contacts to metal1 (a) Actual layout using LASI, (b) cross-sectional view across the contacts.

### 4.1.1 Design Rules for the n+/p+ Active Layers

The active design rules apply to both the n+ and p+ active area. A layout interpretation of the design rules is shown in Fig. 4.2. Note that the location of the active area; that is, whether it is in the substrate or the well affects the rule applied to the active area. In general, n+ active in the well is used to keep the potential of the well fixed, except in the case of the resistor when current flows through the n-well, and p+ in the substrate is used to set the potential of the substrate.

Microchip Ex. 1055, p. 91
Microchip v. Aptiv
IPR2024-00494

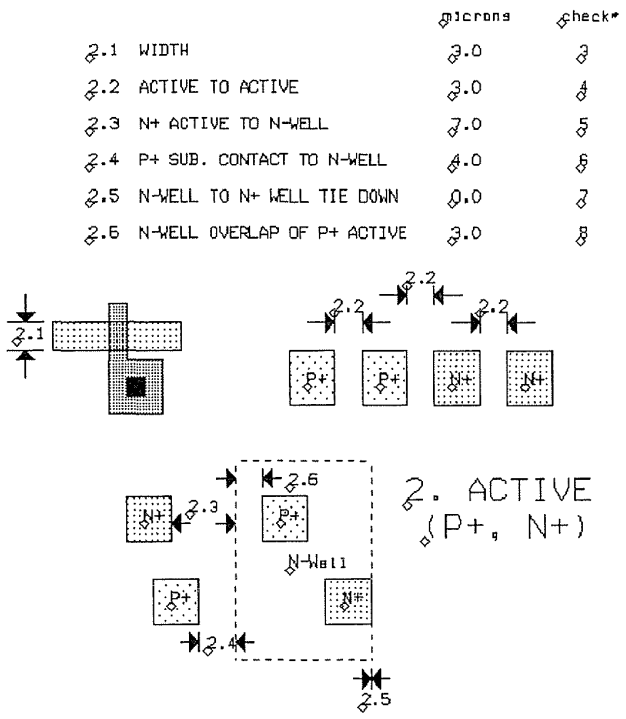



**Figure 4.2** Active area design rules.

**Example 4.1**

Lay out a diode that is 10 μm by 10 μm using the n-well as the cathode. Sketch a cross-sectional view of the diode. If the substrate is held at ground potential, what is the minimum voltage allowable on the cathode (the n-well) of the diode. Why?

The layout and cross-sectional view are shown in Fig. 4.3. If the substrate is at ground the n-well potential should be greater than –0.6 V to keep the substrate/n-well diode from becoming forward biased. ■

The contact design rules are shown in Fig. 4.4. The contact layer is used to connect metal1 to n+, p+, and the poly layers. The size of the contact is exactly 2 μm by 2 μm.

One useful function of the diode is in electrostatic discharge protection. Since the input impedance of a MOSFET is capacitive, small amounts of charge placed on a bonding pad, from something or someone external to the chip connected to a gate, can cause the gate oxide of the MOSFET to break down. To avoid this situation, protection schemes are used as shown in Fig. 4.5. These schemes rely on a diode becoming forward biased and providing a low-impedance path to pull the excessive charge away from the MOSFET gate. The scheme shown in Fig. 4.5b uses the n+ as a resistor to

Microchip Ex. 1055, p. 92
Microchip v. Aptiv
IPR2024-00494

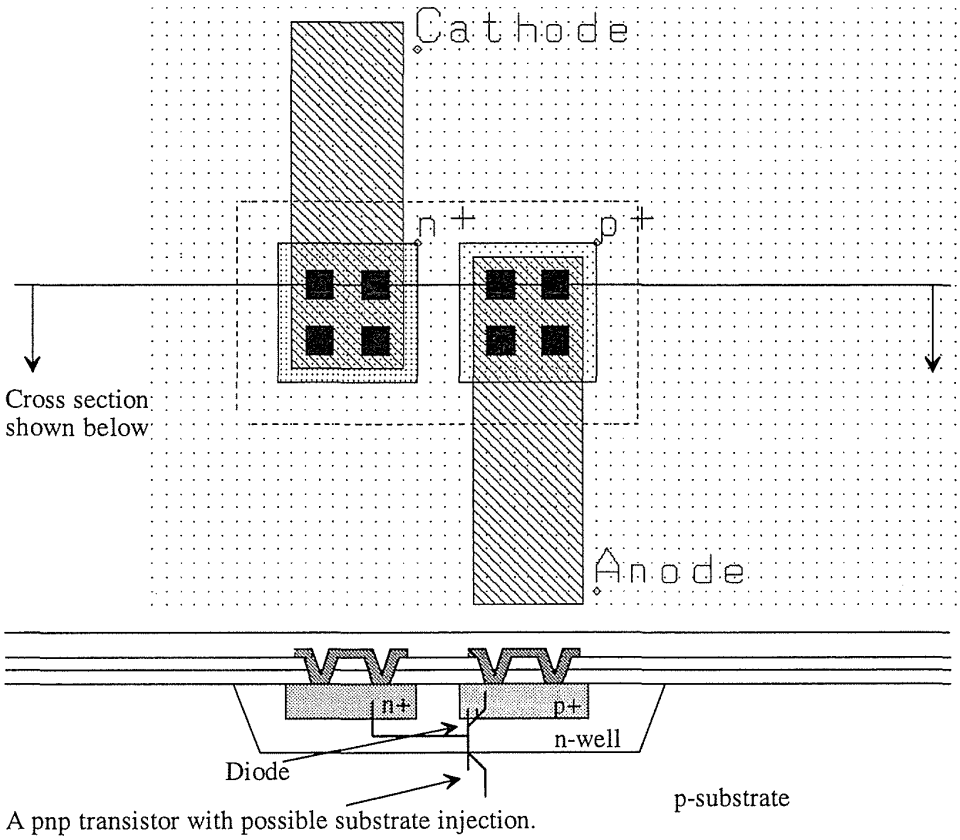



**Figure 4.3** Layout of a 10 um by 10 um diode using the n-well as the cathode.

| | | microns | sheet |
|---|---|---|---|
| 5.1 | CONTACT SIZE (EXACTLY) | 2X2 | 17 |
| 5.2 | SPACING | 2.0 | 18 |
| 5.3 | POLY OVERLAP | 2.0 | 19 |
| 5.4 | ACTIVE OVERLAP | 2.0 | 20 |
| 5.5 | POLY CONTACT TO ACTIVE EDGE | 3.0 | 21 |
| 5.6 | ACTIVE CONTACT TO GATE | 3.0 | 22 |

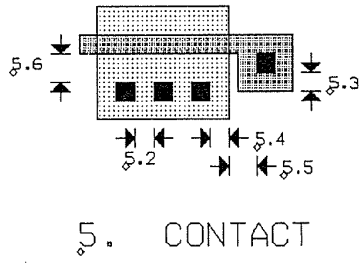



**Figure 4.4** Contact design rules.

Microchip Ex. 1055, p. 93
Microchip v. Aptiv
IPR2024-00494

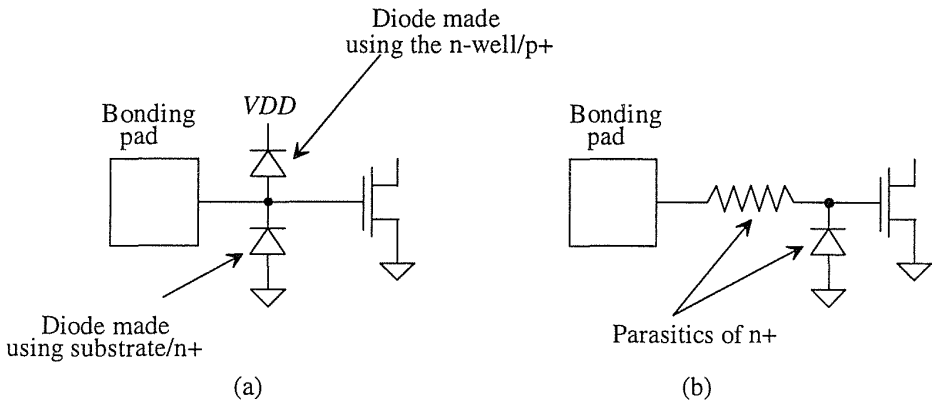



**Figure 4.5** Two methods of protecting against ESD damage.

limit the current through the pad and the n+/substrate diode to protect against negative transients and the breakdown voltage of the diode to protect against positive transients. Positive transient protection can also be achieved by adding the p+/n-well diode of Fig. 4.5a. This scheme can be thought of as a distributed resistor/diode, difficult to analyze analytically. Normally, a combination of both schemes are used together with good power and ground busses to provide a low-resistance path for removal of the unwanted charge.

**Example 4.2**

Modify the pad discussed in Ex. 3.1 to include the diode protection given in Fig. 4.5a. Also include power (*VDD*) and ground (*VSS*) busses on the pad.

The completed pad is shown in Fig. 4.6. Note that *VSS* is connected to the substrate via p+ and that *VDD* is connected to the n-well under part of the pad by way of n+. Also notice that the n+ and p+ areas extend beyond the boundary of the outline of the cell. This is used when the cells are abutted together. The n+ and p+ areas overlap, giving a continuous connection around the die. At least two of the bonding pads must be connected to the *VDD* and *VSS* busses. ■

### 4.1.2 Design Rules for Poly1

The design rules for poly1 are shown in Fig. 4.7. Poly1 is used in making the MOSFET gates and with poly2 capacitors. As we will see in Chapter 7, there is an additional layer of poly named poly2. This layer is used for making capacitors with poly1. Poly2 can also be used to form MOSFETs. However, in order to keep the design rule checks to a minimum, we will assume that only poly1 can be used to make MOSFETs and that poly2 use is limited to capacitors.

Microchip Ex. 1055, p. 94
Microchip v. Aptiv
IPR2024-00494

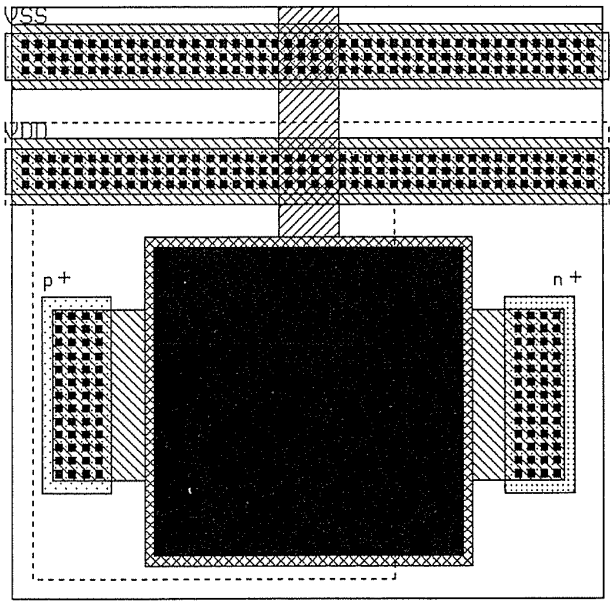
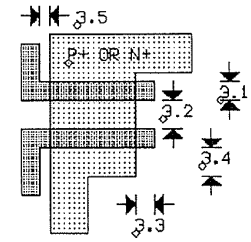



**Figure 4.6** Pad with ESD diodes and power busses.

| | | | |
|---|---|---|---|
| 3.1 | WIDTH | 2.0 | 3 |
| 3.2 | SPACE | 3.0 | 10 |
| 3.3 | GATE OVERLAP OF ACTIVE | 2.0 | 11 |
| 3.4 | ACTIVE OVERLAP OF GATE | 3.0 | not tested |
| 3.5 | FIELD POLY1 TO ACTIVE | 1.0 | not tested |




3. POLY1

**Figure 4.7** Design rules for poly1.

Microchip Ex. 1055, p. 95
Microchip v. Aptiv
IPR2024-00494

Consider the layout shown in Fig. 4.8. Poly1 is drawn over the n+ active area. As we will see later in the chapter, the poly will block n+ diffusion, keeping the area under the poly (gate of the MOSFET) from becoming n+. The width of the poly intersecting with the active area is called the drawn width, while the length of the poly over active is the drawn length. The drawn width and length are the *W* and *L* used to specify the size of a MOSFET. Notice that the n+ regions are implanted through the thin gate oxide (GOX). Also note that the poly over the field oxide (FOX) forms a MOSFET called the field MOSFET, an unwanted parasitic. The FOX is the field MOSFETs gate oxide. An implant is used (see Fig. 2.10) to increase the threshold voltage of the field device so that it cannot turn on.

## 4.2 Layout of a Standard Cell Frame

When laying out n-channel and p-channel transistors, as shown above, it is easy to forget about the bulk connection, that is, the p-substrate for the n-channel transistors and the n-well for the p-channel transistors. Complicating things is the need for *VDD* and *VSS* connections throughout the circuits. Using a standard cell frame can solve both of these layout requirements. The frame provides *VDD* and *VSS* busses in addition to tying down the substrate and well. The drawback of the standard cell frame is increased layout size.

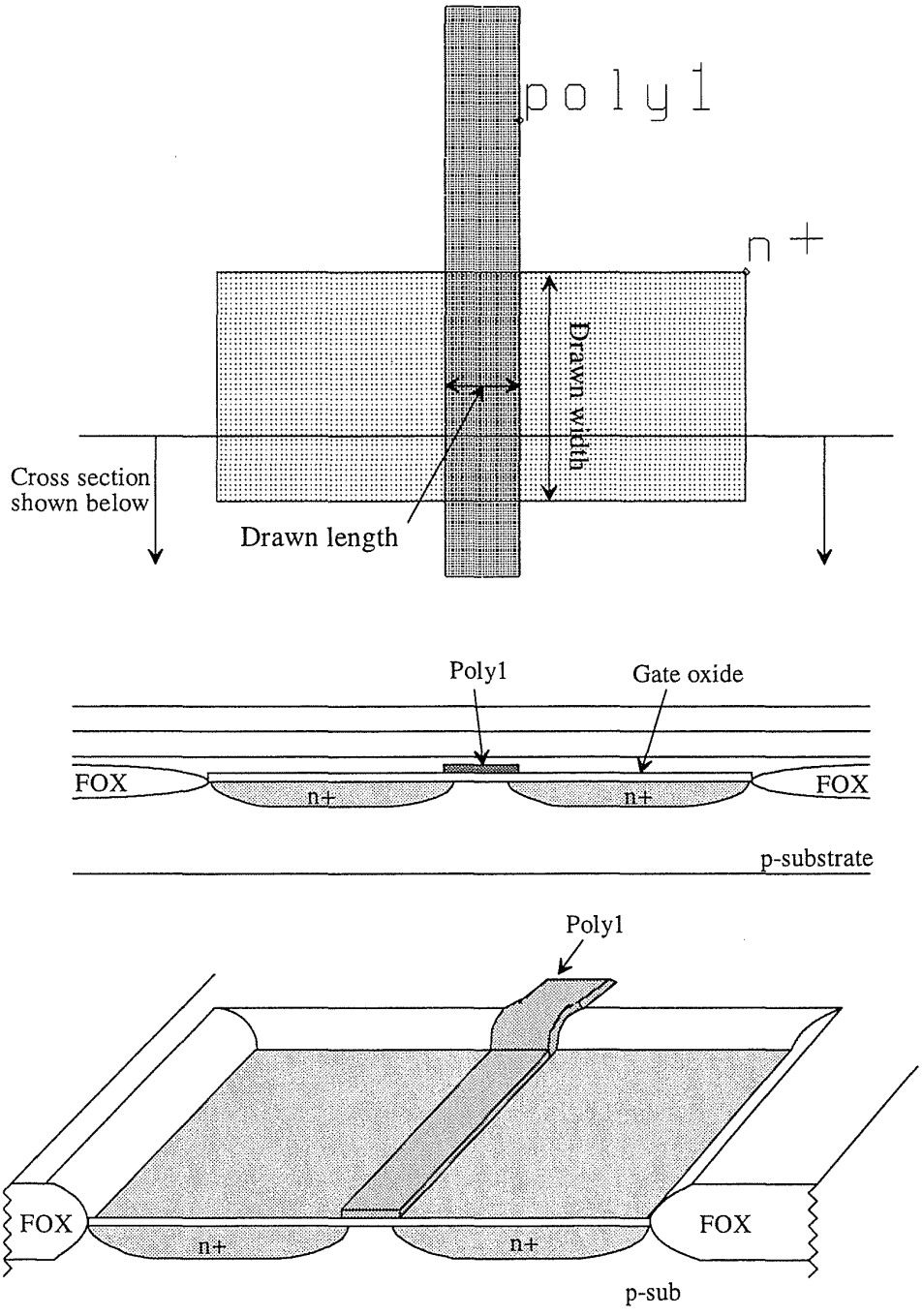
Figure 4.9 shows a standard cell frame that can be used for layout. The width of the outline in this figure is 60 μm, while the height is 120 μm. The metal1 labeled *VSS* is connected to p+ (under metal1), which makes an ohmic contact to the p-substrate. This connection sets the substrate potential to *VSS*. Using the standard cell frame, we won't have to concern ourselves with the substrate connection of the n-channel transistors. The n-well in the top portion of the standard frame cell is connected to *VDD* by the n+ under the metal1 labeled *VDD*. The p-channel transistors are drawn in the n-well, while the n-channel transistors are drawn in the bottom portion of the frame. If, while doing layout, the width of the frame is too narrow, we simply add the frames end to end. Similarly, when laying out a system, we can place the cells end to end, which automatically routes *VDD* and *VSS*.

## 4.3 Patterning the Active Layers

In Ch. 2 we described how to pattern the n-well. This can be thought of as making the substrate for the p-channel transistors (unless we are using the n-well as a resistor or as part of a diode). The region outside the active regions as defined by the n+ and p+ masks is called the field region. The field region is used to isolate the active regions. The oxide in this region is called Field OXide (FOX) or Recessed OXide (ROX). Using oxide to isolate the active regions is called LOCal Oxidation of Silicon, or LOCOS for short.

Figure 4.10 shows the general patterning steps used in creating the active areas and MOSFETs. We assume that the n-well locations on the die have already been

Microchip Ex. 1055, p. 96
Microchip v. Aptiv
IPR2024-00494




Angled view without other insulating layers.

**Figure 4.8** Layout of a MOSFET, cross-sectional and angled view.

Microchip Ex. 1055, p. 97
Microchip v. Aptiv
IPR2024-00494

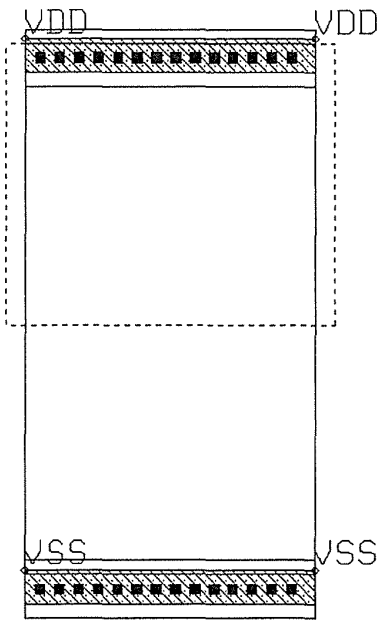


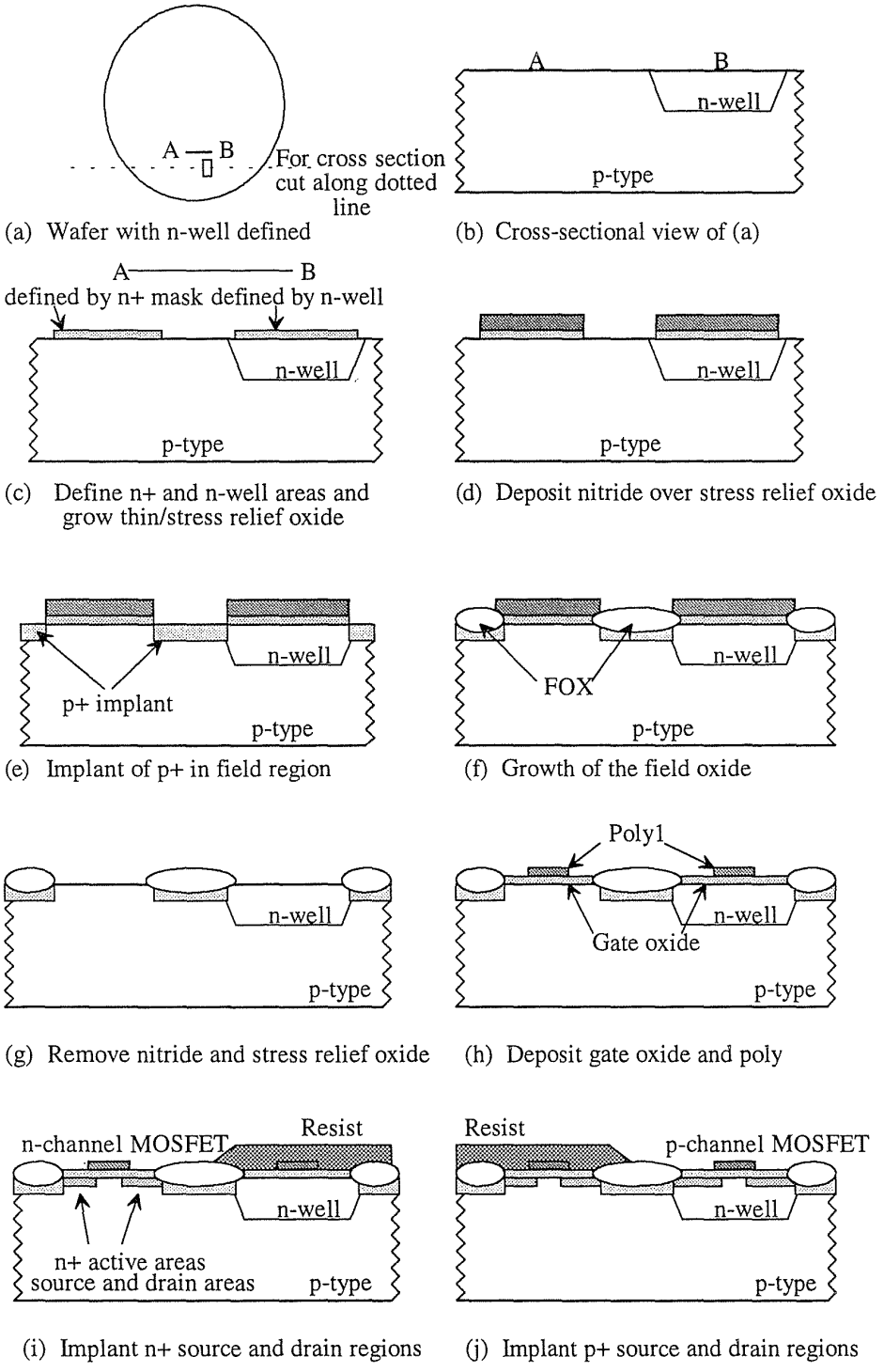

**Figure 4.9** Standard cell frame.

processed (Figs. 4.10a and b). The first step is to cover the n+ active areas and the n-well with a stress relief oxide. Nitride, $SiN_3$, is deposited on top of the stress relief oxide (Figs. 4.10c and d). This thin oxide acts like a cushion for the nitride. The areas not covered by the nitride, that is, the p-substrate areas, are implanted with a p field implant, (Fig. 4.10e), for increasing the threshold voltage of the field devices. This implant is used to increase the threshold voltage of the field MOSFETs, the devices formed by poly over the FOX. Not shown in Fig. 4.10 are the steps involved in implanting the n-type field implant used to raise the threshold voltage of the field MOSFETs in the n-well region.

After the field implants are in place, the FOX is grown (Fig. 4.10f), and the nitride and stress relief oxides are removed. At this point we should point out an unwanted characteristic of LOCOS, namely, bird's beak. Consider Fig. 4.11a. This figure shows the layout of a 10 μm by 10 μm n+ active area with a poly gate. Figure 4.11b shows the cross-sectional area with field oxide in place. (This is a 90° rotation from the cross-sectional view shown in Fig. 4.8.)

Notice that the oxide has encroached upon the area defined by the n+ active mask. The shape of this encroachment is sometimes referred to as bird's beak. We will refer to this attribute of LOCOS as simply oxide encroachment. This encroachment will affect the width of the MOSFETs. In other words, the width of the MOSFET will be decreased by two times the length of the oxide encroachment. The BSIM SPICE model discussed in Ch. 6 gives the parameter *DW*, or delta width, to describe the oxide

Microchip Ex. 1055, p. 98
Microchip v. Aptiv
IPR2024-00494




(a) Wafer with n-well defined

(b) Cross-sectional view of (a)

(c) Define n+ and n-well areas and grow thin/stress relief oxide

(d) Deposit nitride over stress relief oxide

(e) Implant of p+ in field region

(f) Growth of the field oxide

(g) Remove nitride and stress relief oxide

(h) Deposit gate oxide and poly

(i) Implant n+ source and drain regions

(j) Implant p+ source and drain regions

**Figure 4.10** Sequence of events used in MOSFET formation.

Microchip Ex. 1055, p. 99
Microchip v. Aptiv
IPR2024-00494

encroachment. If the drawn width of the MOSFET gate is W, then the effective width is given by

$$W_{eff} = W_{drawn} - DW \tag{4.1}$$

The parameter *DW* is equal to twice the oxide encroachment. The BSIM SPICE model will automatically subtract *DW* from the drawn width. (The drawn width is the *W* specified by the engineer in the SPICE simulation.)

**Example 4.3**

Assuming the drawn width of the MOSFET in Fig. 4.11 is 10 μm, estimate the effective width after oxide encroachment.

From the BSIM model parameters for the n-channel MOSFET in Appendix A, *DW* (dw) = 0.1355 μm; therefore, the actual or effective width of the MOSFET shown in Fig. 4.11 is 9.87 μm. Notice that the smaller the drawn width of the device, the larger effect the oxide encroachment has on the effective width. ■

Let's return to the sequence of events in Fig. 4.10. The next step in the process is to grow the thin gate oxide and deposit the poly (Fig. 4.10h). This is followed by depositing resist, to isolate the n+ areas, and n+ implant. The result is shown in Fig. 4.10i. This is our basic MOSFET structure. From a circuits point of view, several important events occur during these last steps of MOSFET fabrication. When the poly is deposited, it is doped "in situ" to decrease its resistivity. For CN20 the gate is an n+ doping. When the n+ implant is applied to the wafer, the poly, which is used as a mask to keep the n+ from going under the gate region[2], is doped further. The n+ poly gate also acts as a mask for the p-channel transistors. The doping introduced by the p+ implant is not sufficient to significantly change the doping levels in the polysilicon gate. Typically, sheet resistances for poly are on the order of 20 Ω/square. In order to lower the sheet resistance of the poly, a refractory metal and poly mixture called silicide is deposited to the top of the polysilicon. This silicide/poly sandwich is referred to as a polycide gate. Typical sheet resistances of polycide gates are 2-3 Ω/square. The Orbit process (CN20) we are using does not silicide the gate regions.

Another important thing happens when we implant n+ and p+. The dopant atoms diffuse under the gate of the MOSFET. This effect is shown in Fig. 4.12. As can be seen in this figure, the lateral diffusion (*LD*) subtracts from the drawn gate length. The level 2 SPICE model uses *LD* to define the lateral diffusion. The BSIM model defines *DL* (dl), or delta length, to be two times LD. The effective channel length is given by

$$L_{eff} = L_{drawn} - 2 \cdot LD = L_{drawn} - DL \tag{4.2}$$

[2] This is termed a "self-aligned" CMOS process because the gate automatically becomes aligned to the drain and the source regions of the MOSFET.

Microchip Ex. 1055, p. 100
Microchip v. Aptiv
IPR2024-00494

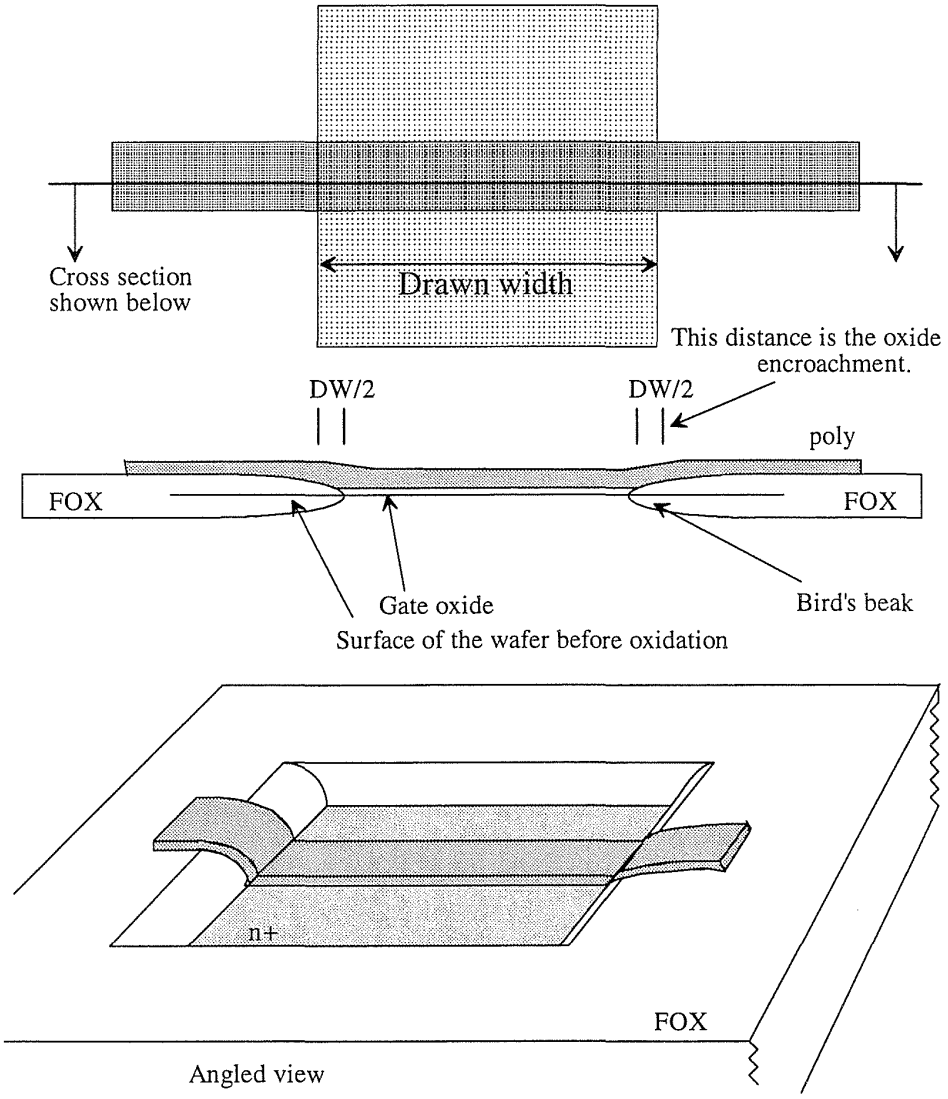



**Figure 4.11** Layout used to illustrate oxide encroachment.

**Example 4.4**

Assuming the drawn length of the MOSFET in Fig. 4.12 is 2 μm, estimate the effective length of the MOSFET after lateral diffusion.

From the BSIM model parameters for the n-channel MOSFET given in Appendix A, $DL$ = 0.64 μm and therefore the effective length of the device is 1.36 μm. ■

Microchip Ex. 1055, p. 101
Microchip v. Aptiv
IPR2024-00494

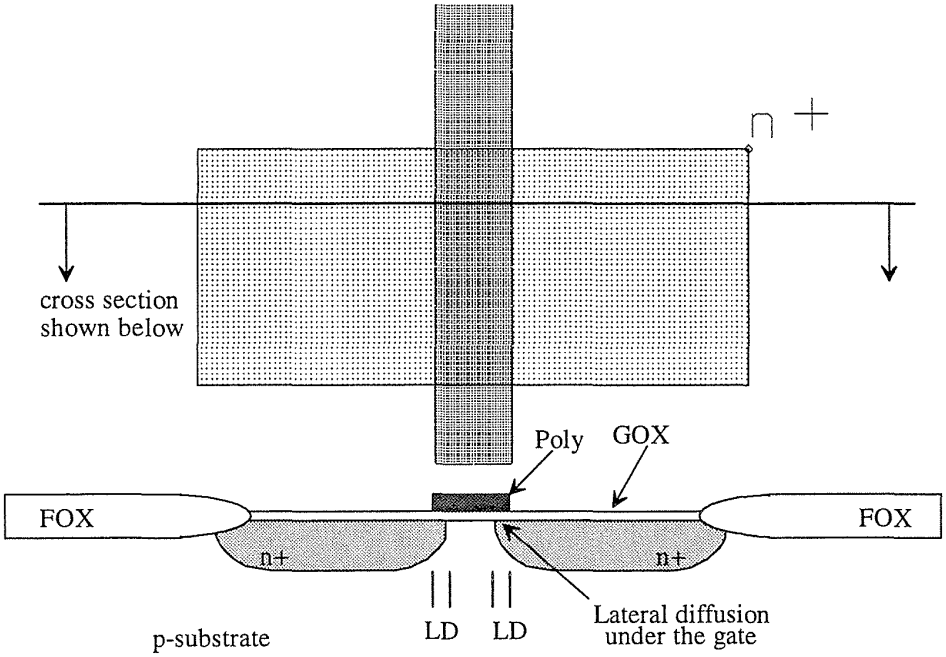



**Figure 4.12** Layout used to illustrate lateral diffusion.

The BSIM model will automatically calculate the effective width and length of the MOSFET when given the drawn lengths and widths. The level 2 model will only calculate the effects due to lateral diffusion. ($W_{eff}$ must be given by the designer.)

## 4.4 Layout of the MOSFET

The layout of the MOSFET without contacts (connections to a circuit) was shown in Figs. 4.8, 4.11, and 4.12. Consider the layout shown in Fig. 4.13 of a MOSFET with connections to the source, drain, and gate. Note that the source and drain of the MOSFET are interchangeable. The substrate connection of the MOSFET is not shown in this figure. Placing the MOSFET in the bottom of the standard cell frame of Fig. 4.9 provides the substrate connection. Using the pad of Fig. 4.6 in a padframe anchors the substrate to *VSS* around the die. P-channel MOSFETs, which are laid out in the n-well, must have a well tie down, that is, an n+ region connected to *VDD* or the source of the MOSFET.

### 4.4.1 Parasitics Associated with the Active Layers

Associated with the source and drain implants (n+ and p+) is a sheet resistance. We can get the sheet resistance from the tables in Appendix A or directly from the BSIM model parameters. The n-channel transistor has n+ source and drain regions, while the p-channel transistor has p+ source and drain regions. The sheet resistance of the n+ implant is given in the n-channel BSIM model by "rsh" (= 27.9 Ω).

Microchip Ex. 1055, p. 102
Microchip v. Aptiv
IPR2024-00494

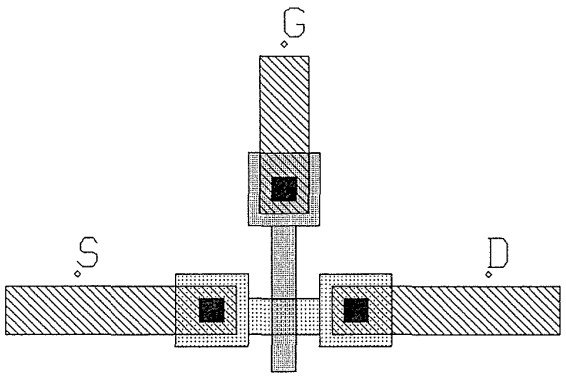



**Figure 4.13** Layout of a 2 μm (= L) by 3 μm (= W) n-channel MOSFET with metal1 wire connections. Substrate connection not shown.

**Example 4.5**

Estimate the resistance of the following layout (3 μm wide and 100 μm long) made using the n+ implant. Neglect corner effects.

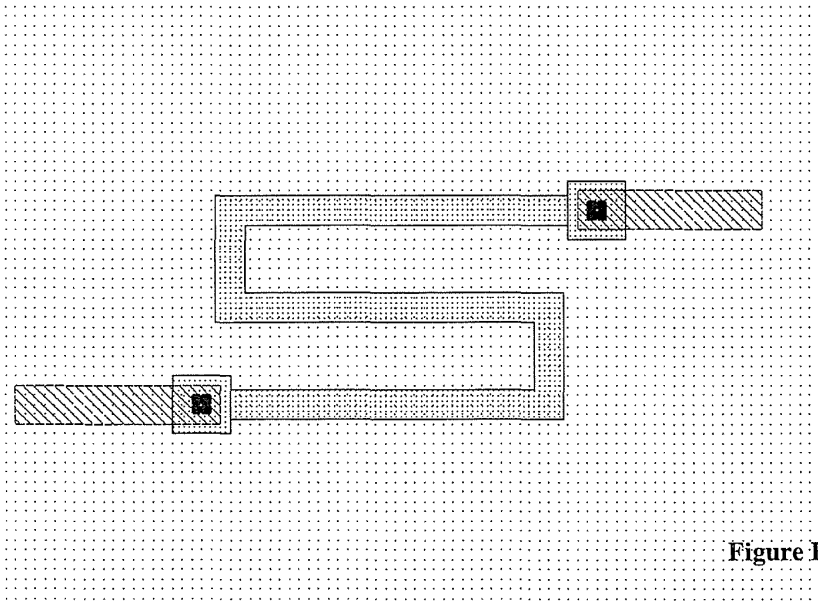


**Figure Ex4.5**

This layout is an n+ resistor with a value of $R = 27.9 \cdot \frac{100}{3} \approx 930\ \Omega$. ■

The depletion capacitance of the n+ and p+ regions consists of a bottom component and a sidewall component (see Fig. 4.14). The zero-bias depletion

Microchip Ex. 1055, p. 103
Microchip v. Aptiv
IPR2024-00494

capacitances are given in the BSIM model by *cj* (bottom, $F/m^2$) and *cjsw* (sidewall, *F/m*). The built-in potential of the bottom and the sidewall is *pb* and *pbsw* respectively, and the grading coefficients of the bottom and sidewall are *mj* and *mjsw*, respectively. These parameters are given in the BSIM model. The depletion capacitance can be calculated knowing the area of the drain (or source) and the perimeter, for the n+ to bulk, by

$$C_{jdep} = \frac{cj \cdot A_D}{\left(1 + \frac{V_{DB}}{pb}\right)^{mj}} + \frac{cjsw \cdot P_D}{\left(1 + \frac{V_{DB}}{pbsw}\right)^{mjsw}} \tag{4.3}$$

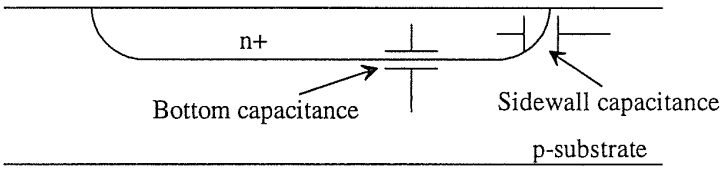



**Figure 4.14** Depletion capacitances of n+ to p-substrate.

**Example 4.6**

Estimate the depletion capacitance between the drain of the MOSFET shown in Fig. 4.13 and the substrate.

The size of the n+ implant shown in this figure is approximately 6 μm by 6 μm, neglecting the small active area between the poly and the contact. This gives $A_D$ = 36 x $10^{-12}$ $m^2$ (36 $\mu m^2$) and $P_D$ = 24 μm. The capacitance as a function of $V_{DB}$ (= drain voltage minus the substrate voltage) is given by

$$C_{jdep} = \frac{1.04\times10^{-4}\frac{F}{m^2}\cdot 36\times10^{-12}m^2}{(1+\frac{V_{DB}}{0.8})^{.66}} + \frac{2.2\times10^{-10}\frac{F}{m}\cdot 24\times10^{-6}m}{(1+\frac{V_{DB}}{0.8})^{0.18}}$$

$$C_{jdep} = \frac{3.75\text{ fF}}{(1+\frac{V_{DB}}{0.8})^{.66}} + \frac{5.3\text{ fF}}{(1+\frac{V_{DB}}{0.8})^{0.18}}$$

This result shows how the depletion capacitance changes with drain-substrate potential. ■

In many situations, during the course of designing circuits we will neglect the depletion capacitance voltage dependence and simply use the zero-bias capacitance. In the above example, we would use 9 fF as the capacitance between the drain and substrate.

Microchip Ex. 1055, p. 104
Microchip v. Aptiv
IPR2024-00494

## REFERENCES

[1] G. Massobrio and P. Antognetti, *Semiconductor Device Modeling with SPICE,* 2nd ed., McGraw-Hill, 1993. ISBN 0-07-002469-3.

[2] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits,* McGraw-Hill Publishing Company, 2nd ed., 1988. ISBN 0-07-029158-6.

[3] R. S. Muller and T. I. Kamins, *Device Electronics for Integrated Circuits,* 2nd ed., John Wiley and Sons, 1986. ISBN 0-471-88758-7.

## PROBLEMS

**4.1** Lay out a nominally 250 kΩ resistor with metal1 wire connections. DRC your layout. What would happen if the layout did not include n+ under the contacts?

**4.2** Lay out and DRC the pad of Fig. 4.6. Comment on the layout; that is, describe the operation and layout of the power busses and protection diodes.

**4.3** Sketch the cross-sectional views across the *VDD* and *VSS* power busses of the standard cell frame of Fig. 4.9.

**4.4** How much does the oxide encroach upon the width of the p-channel MOSFET in the CN20 process? Does the oxide encroachment affect the length of the MOSFET? Sketch the location of the oxide encroachment on the layout (top view) of a MOSFET.

**4.5** How much does the lateral diffusion extend under the gate of a p-channel MOSFET in the CN20 process? Does the lateral diffusion affect the width of the MOSFET? Sketch the location of the lateral diffusion on the layout of the MOSFET.

**4.6** Lay out an n-channel MOSFET, similar to Fig. 4.13, with an $L$ and $W$ of 5 μm. Include a p+ connection to the substrate for the fourth terminal of the MOSFET. DRC the layout.

**4.7** Using the standard frame layout, both a p- and n-channel MOSFET with $L = W = 5$ μm. Is the substrate and well connected to *VSS* and *VDD,* respectively? How?

**4.8** What is the maximum depletion capacitance of an n+ implant that measures 10 μm by 10 μm? (*Hint*: The maximum capacitance is simply the zero-bias depletion capacitance, assuming the junction is zero or reverse biased.) If the implant is held at a constant potential and the substrate potential is reduced, what happens to the depletion capacitance?

**4.9** Estimate the delay through a 1 mm length, 2 μm wide run of poly1 loaded with 100 *fF* capacitors every 10 μm.

**Microchip Ex. 1055, p. 105**
**Microchip v. Aptiv**
**IPR2024-00494**

**4.10** Sketch the cross-sectional views across the lines shown in Fig. P4.10.

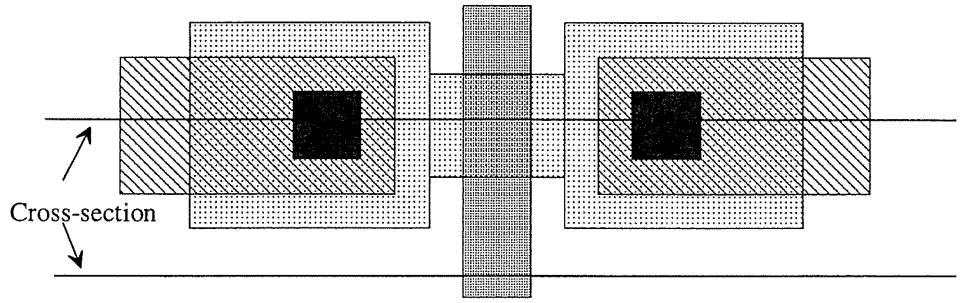



**Figure P4.10**

**4.11** Sketch the cross-sectional views across the lines shown in Fig. P4.11.

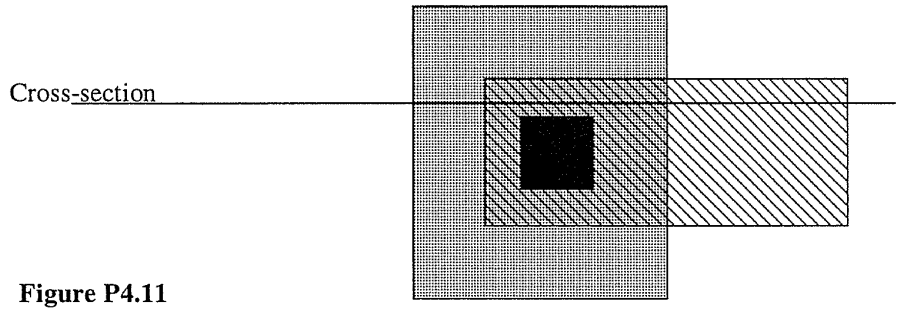
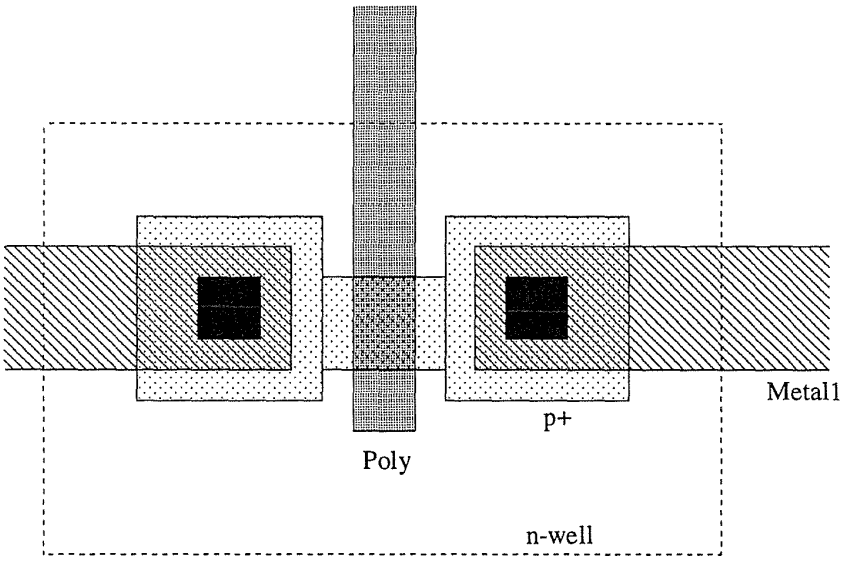



**Figure P4.11**

Microchip Ex. 1055, p. 106
Microchip v. Aptiv
IPR2024-00494

**4.12** Lay out a diode using the n+ diffusion and the p-substrate. Assume the diode area is a square where the side length is 20 μm. Show connections to the anode and cathode of the diode. Sketch the cross-sectional view of the diode. Specify the SPICE statement and model for the diode (*Hint*: to determine $I_S$ of the diode, use "js" given in the n-channel BSIM model of Appendix A and the area of the diode). Are there any restrictions on the diode anode voltage?

**4.13** The layout of a p-channel MOSFET shown in Fig. P4.13 is incorrect. What is the problem?




What is the fatal error associated with this layout?

**Figure P4.13**

Microchip Ex. 1055, p. 107
Microchip v. Aptiv
IPR2024-00494

Chapter

5

# The MOSFET

At this point we should have some appreciation for the parasitics, that is, capacitances and resistances, associated with a CMOS process. In this chapter we discuss the MOSFET operation. To begin let's define the symbols used to denote the n-channel and p-channel MOSFETs (see Fig. 5.1). When the substrate is connected to *VSS* and the well is tied to *VDD*, we will use the simplified models shown at the bottom of the figure. It is important to keep in mind that the MOSFET is a four-terminal device and that the source and drain of the MOSFET are interchangeable.

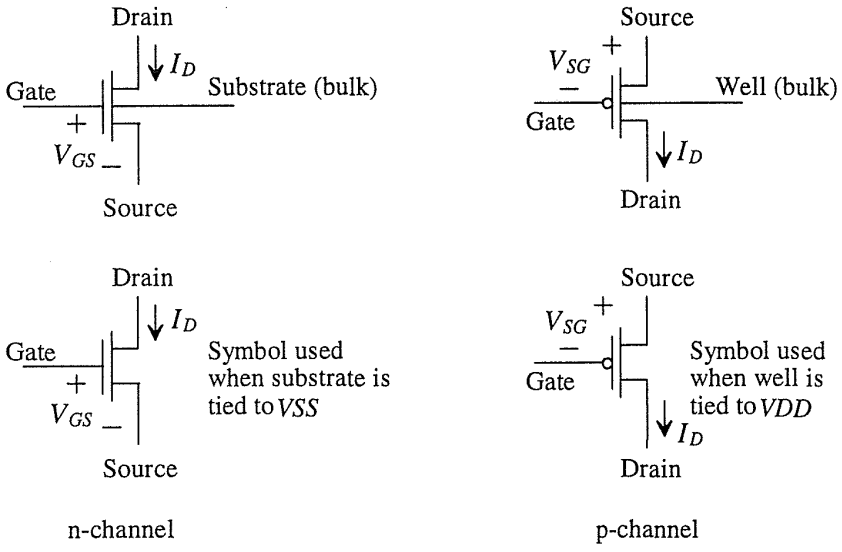



**Figure 5.1** Symbols used for n- and p-channel MOSFETs.

Microchip Ex. 1055, p. 108
Microchip v. Aptiv
IPR2024-00494

## 5.1 The MOSFET Capacitances

Consider the MOSFET shown in Fig. 5.2 and its associated cross-sectional view. Associated with the drain and source regions to the substrate is a depletion capacitance that was discussed in the previous chapter. In this section we will concentrate on the capacitances associated with the gate electrode, that is, the capacitance from the gate to ground in Fig. 5.2.

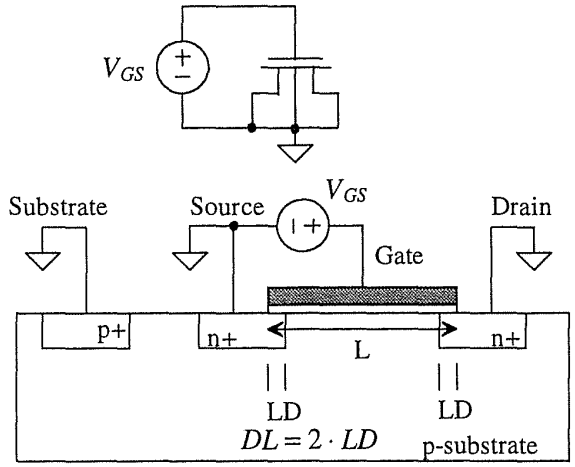



**Figure 5.2** Cross-sectional view of MOSFET used to calculate capacitances.

### 5.1.1 Case I: Accumulation

Let's first consider the case when $V_{GS} < 0$ (Fig. 5.3). Under this condition, mobile holes from the substrate are attracted under the gate oxide. The thickness of the oxide in the SPICE MOSFET model is given by the parameter *TOX*. The capacitance between the *gate* electrode and the *substrate* electrode is given by

$$C_{gb} = \frac{\varepsilon_{ox} \cdot \overbrace{(L - 2 \cdot LD)}^{L_{eff}} \cdot W}{TOX} \tag{5.1}$$

where $\varepsilon_{ox}$ (= 3.97 · 8.85 aF/μm) is the dielectric constant of the gate oxide, $W$ is the drawn width (neglecting oxide encroachment), and $L - 2 \cdot LD$ is the effective channel length. The capacitance between the gate and drain or source is given by

$$C_{gd,s} = \frac{\varepsilon_{ox} \cdot LD \cdot W}{TOX} = \text{Gate-drain (or source) overlap capacitance} \tag{5.2}$$

neglecting oxide encroachment. The gate-drain overlap capacitance is present in a MOSFET regardless of the biasing conditions. This capacitance is specified in SPICE MOSFET models by the variables *CGDO* and *CGSO* with units of farads/meter. Estimation of $C_{gd}$ or $C_{gs}$ using the measured BSIM model parameters uses

Microchip Ex. 1055, p. 109
Microchip v. Aptiv
IPR2024-00494

$$C_{gd} = CGDO \cdot W = \frac{\varepsilon_{ox} \cdot LD}{TOX} \cdot W \tag{5.3}$$

and

$$C_{gs} = CGSO \cdot W \qquad \text{(farads)} \tag{5.4}$$

The total capacitance, independent of the width and length of the MOSFET, between the gate and ground in the circuit of Fig. 5.2 is the sum of $C_{gd}$, $C_{gs}$, and $C_{gb}$ and is given by

$$C'_{ox} = \frac{\varepsilon_{ox}}{TOX} \qquad \text{(farads/meter}^2\text{)} \tag{5.5}$$

The term $C'_{ox}$ is called the oxide capacitance, which for the CN20 process is approximately 800 aF/μm$^2$. Knowing the width and length of a MOSFET gives a total capacitance from the gate of the MOSFET in Fig. 5.2 to ground of

$$C_{ox} = C'_{ox} \cdot W \cdot L \qquad \text{(farads)} \tag{5.6}$$

There is a significant resistance in series with $C_{gb}$ in Fig. 5.3 from the resistivity of the p-substrate. The resistivity of the n+ source and drain regions tends to be small enough to neglect in most circuit design applications.

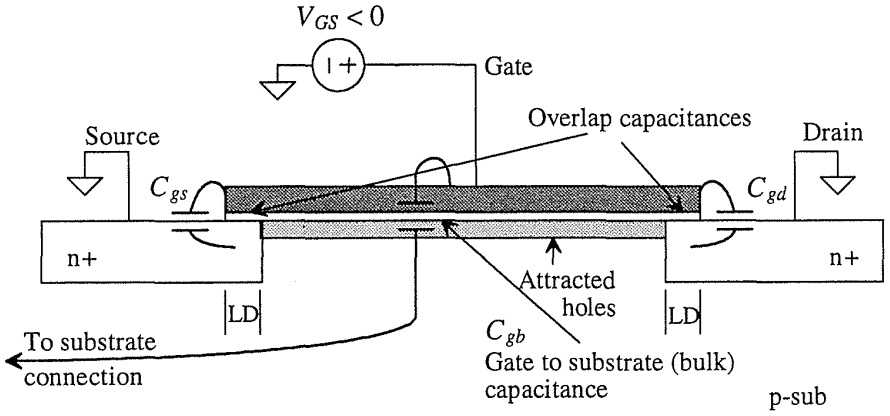



**Figure 5.3** MOSFET in accumulation.

### 5.1.2 Case II: Depletion

Referring again to Fig. 5.2, let's consider the case when $V_{GS}$ is not negative enough to attract a large number of holes under the oxide and not positive enough to attract a large number of electrons. Under these conditions, the surface under the gate is said to be depleted. Consider Fig. 5.4. As $V_{GS}$ is increased from some negative voltage, holes will be displaced under the gate, leaving immobile acceptor ions that contribute a negative charge. We see that as we increase $V_{GS}$ a capacitance between the gate and the induced channel under the oxide exists. Also, a depletion capacitance between the induced channel and the substrate is formed. The capacitance between the gate and the

Microchip Ex. 1055, p. 110
Microchip v. Aptiv
IPR2024-00494

source/drain is simply the overlap capacitance, while the capacitance between the gate and the substrate is the oxide capacitance in series with the depletion capacitance. The depletion layer shown in Fig. 5.4 is formed between the substrate and the induced channel. The MOSFET operated in this region is said to be in weak inversion or the subthreshold region because the surface under the oxide is not heavily n+.

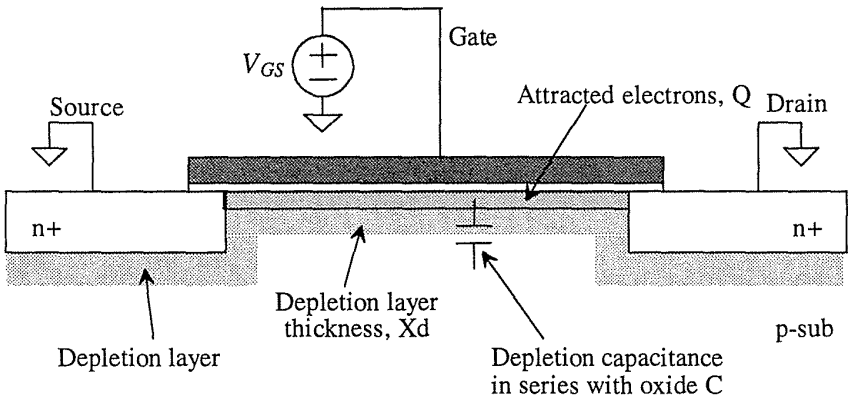



**Figure 5.4** MOSFET in depeletion.

### 5.1.3 Case III: Strong Inversion

When $V_{GS}$ is sufficiently large (> $V_{THN}$, the threshold voltage of the n-channel MOSFET) so that a large number of electrons are attracted under the gate, the surface is said to be inverted, that is, no longer p-type. Figure 5.5 shows how the capacitance from the gate to ground changes as $V_{GS}$ changes for the MOSFET configuration of Fig. 5.2. This figure can be misleading. Remember that when the MOSFET is in the accumulation region the majority of the capacitance to ground, $C_{gb}$, runs through the large parasitic

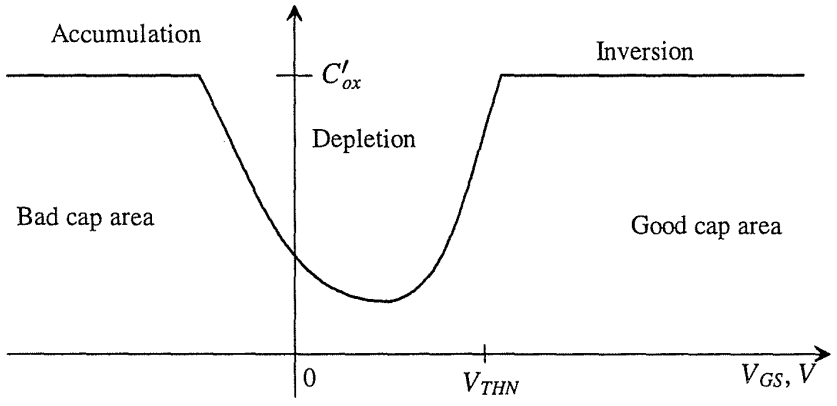



**Figure 5.5** Capacitance to ground at the gate terminal of the circuit shown in Fig. 5.2 plotted against the gate-source voltage.

Microchip Ex. 1055, p. 111
Microchip v. Aptiv
IPR2024-00494

resistance of the substrate. Also note that the MOSFET makes a very good capacitor when $V_{GS} > V_{THN}$ + a few hundred mV. We will make a capacitor in this fashion many times while we are designing circuits.

**Example 5.1**

Suppose the following MOSFET configuration is to be used as a capacitor. If the width and length of the MOSFET are both 100 μm, estimate the capacitance. Are there any restrictions on the voltages we can use across the capacitor?

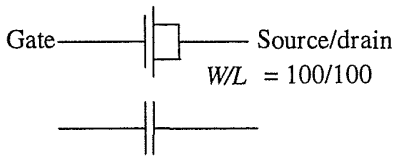



**Figure Ex5.1**

Since the MOSFET is to be used as a capacitor, we require operation in the strong inversion region, with the gate potential always at $V_{THN}$ + 100 mV above the source/drain potentials. The capacitance between the gate and the source/drain is then $C_{tot} = C'_{ox} \cdot W \cdot L$. For the CN20 process this results in

$$C_{tot} = (800 \text{ aF}/\mu\text{m}^2)(100 \ \mu\text{m})(100 \ \mu\text{m}) = 8 \text{ pF}$$

Note that we did not concern ourselves with the substrate connection. Since we are assuming strong inversion, the bulk (substrate) connection will only affect the capacitances from the drain/source to substrate (those from the source/drain implant regions). We will see, however, that the connection of the substrate will significantly affect the threshold voltage of the devices. ■

### 5.1.4 Summary

Figure 5.6 shows our MOSFET symbol with capacitances. The capacitance $C_{gb}$ is associated with the gate poly over the field region. The gate-drain capacitance, $C_{gd}$ and the gate-source capacitance $C_{gs}$ are determined by the region of operation (see Table 5.1).

| Name | Off | Triode | Saturation |
|---|---|---|---|
| $C_{gd}$ | $CGDO \cdot W$ | $\frac{1}{2} \cdot W \cdot L \cdot C'_{ox}$ | $CGDO \cdot W$ |
| $C_{db}$ | $C_{jdep}$ | $C_{jdep}$ | $C_{jdep}$ |
| $C_{gb}$ | $C'_{ox} WL_{eff} + CGBO \cdot L$ | $CGBO \cdot L$ | $CGBO \cdot L$ |
| $C_{gs}$ | $CGSO \cdot W$ | $\frac{1}{2} \cdot W \cdot L \cdot C'_{ox}$ | $\frac{2}{3} \cdot W \cdot L \cdot C'_{ox}$ |
| $C_{sb}$ | $C_{jdep}$ | $C_{jdep}$ | $C_{jdep}$ |

**Table 5.1** MOSFET capacitances.

Microchip Ex. 1055, p. 112
Microchip v. Aptiv
IPR2024-00494

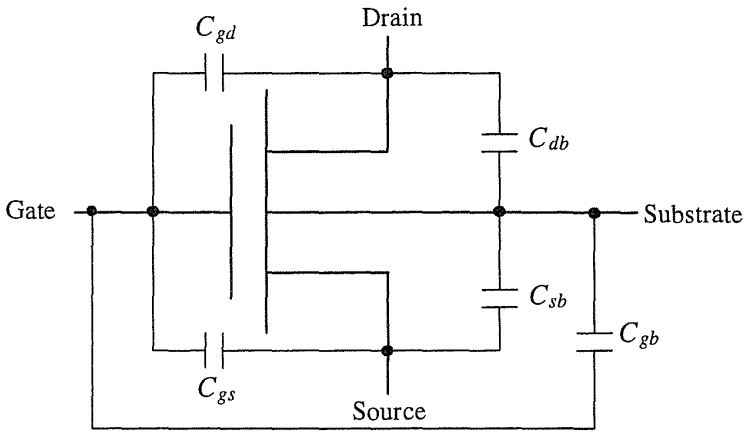



**Figure 5.6** MOSFET capacitances.

## 5.2 The Threshold Voltage

In the last section we said that the semiconductor/oxide surface is inverted when $V_{GS}$ is greater than the threshold voltage $V_{THN}$. Under these conditions, a depletion region exists between the inverted channel and the substrate. The thickness of the depletion region (Fig. 5.4) is given from pn junction theory by

$$X_d = \sqrt{\frac{2\varepsilon_{si}\phi}{qN_A}} = \sqrt{\frac{2\varepsilon_{si}|\phi_s - \phi_F|}{qN_A}} \tag{5.7}$$

where $N_A$ is the number of acceptor atoms in the substrate, $\phi_s$ is the electrostatic potential at the oxide-silicon interface, and the electrostatic potential of the p-type substrate is given by

$$\phi_F = -\frac{kT}{q}\ln\frac{N_A}{n_i} \tag{5.8}$$

where $n_i$ is the intrinsic carrier concentration of silicon (= $14.5 \times 10^9$ atoms/cm$^3$). The depletion region in the p-type semiconductor is void of mobile holes. The absence of holes in this region leaves a net negative charge due to the immobile acceptor atoms that remain behind. This charge is equal to the charge attracted under the gate. The charge/unit area is given by

$$Q_b' = qN_A X_d = \sqrt{2\varepsilon_{si}qN_A|\phi_s - \phi_F|} \tag{5.9}$$

If the surface electrostatic potential at the oxide interface, $\phi_s$, is simply the same as the bulk electrostatic potential $\phi_F$ (i.e., $\phi_s = \phi_F$ and then $Q_b' = 0$), the MOSFET is operating in the accumulation mode, or the MOSFET is OFF in circuit terms. At this point both $\phi_s$ and $\phi_F$ are negative numbers. The number of holes at the oxide-semiconductor surface is $N_A$, the same concentration as the bulk.

Microchip Ex. 1055, p. 113
Microchip v. Aptiv
IPR2024-00494

As $V_{GS}$ is increased, the surface potential becomes more positive. When $\phi_s = 0$, the surface under the oxide has become depleted (the carrier concentration is $n_i$). When $\phi_s = -\phi_F$ (a positive number), the channel is inverted (electrons are pulled under the oxide forming a channel), and the electron concentration at the semiconductor-oxide interface is equal to the substrate doping concentration. The value of $V_{GS}$ when $\phi_s = -\phi_F$is arbitrarily defined as the threshold voltage, $V_{THN}$. Note that the surface potential changed a total of $2\phi_F$ between the strong inversion and depletion cases.

For $V_{GS} = V_{THN}$ ($\phi_s = -\phi_F$), the fixed negative charge in the depletion region is given by

$$Q'_{bo} = \sqrt{2qN_A\varepsilon_{si}|-2\phi_F|} \tag{5.10}$$

with units of coulombs/m$^2$. Up to this point we have assumed that the substrate and source were tied together to ground. If the source of the n-channel MOSFET is at a higher potential than the substrate, the potential difference is given by $V_{SB}$; the negative charge present in the depletion region then becomes

$$Q'_b = \sqrt{2qN_A\varepsilon_{si}|-2\phi_F+V_{SB}|} \tag{5.11}$$

**Example 5.2**

For a substrate doping of $10^{15}$ atoms/cm$^3$, $V_{GS} = V_{THN}$ and $V_{SB} = 0$, estimate the electrostatic potential in the substrate region and at the oxide-semiconductor interface, the depletion layer width, and the charge contained in the depletion region and thus the inverted region under the gate.

The electrostatic potential of the substrate is

$$\phi_F = -\frac{kT}{q}\ln\frac{N_A}{n_i} = -26\text{ mV}\cdot\ln\frac{10^{15}}{14.5\times10^9} = -290\text{ mV}$$

and therefore the electrostatic potential at the oxide semiconductor interface ($V_{GS} = V_{THN}$), $\phi_s$, is 290 mV. The depletion layer width is given by

$$X_d = \sqrt{\frac{2\cdot11.7\cdot(8.85\times10^{-18}\text{F/}\mu\text{m})(2\cdot0.29\text{V})}{(1.6\times10^{-19}\frac{\text{C}}{\text{atom}})(10^{15}\frac{atoms}{\text{cm}^3})(\frac{\text{cm}^3}{10^{12}\mu\text{m}^3})}} = 0.866\ \mu\text{m}$$

and the charge contained in this region, from Eq. (5.10) or (5.9) with $\phi_s = -\phi_F$ by

$$Q'_{bo} = qN_AX_d = \left(1.6\times10^{-19}\frac{\text{C}}{\text{atom}}\right)\left(10^{15}\frac{\text{atoms}}{\text{cm}^3}\right)\left(\frac{\text{cm}^3}{10^{12}\mu\text{m}^3}\right)(0.866\mu\text{m})$$

$$= 139\ \frac{\text{aC}}{\mu\text{m}^2}$$

which is the charge contained in the depletion region or oxide-semiconductor interface. Note that this is true only when $V_{GS} = V_{THN}$. ■

Microchip Ex. 1055, p. 114
Microchip v. Aptiv
IPR2024-00494

Consider the MOSFET shown in Fig. 5.7. We will assume that the applied $V_{GS} = V_{THN}$ so that the preceding discussions and assumptions hold. The potential across the gate-oxide capacitance, $C'_{ox}$, is simply

$$V_{BC} = \frac{Q'_b}{C'_{ox}} \tag{5.12}$$

The surface potential *change*, $V_C$ (= $\Delta\phi_s$), from the equilibrium case is $2\phi_F$. (The absolute voltage of the channel is 0 V; that is, the source, drain, and channel are at ground.) The potential needed to change the surface potential and overcome the depletion layer charge is given by

$$V_B = \frac{Q'_b}{C'_{ox}} - 2\phi_F \tag{5.13}$$

An additional positive charge exists at the silicon-oxide interface due to imperfections during growth of the gate oxide or as the result of an ion implantation used to adjust the threshold voltage of the MOSFET. We will call this positive charge $Q'_{ss}$ with units of coulombs/area. Equation (5.13) may be rewritten including these surface charges as

$$V_B = \frac{Q'_b - Q'_{ss}}{C'_{ox}} - 2\phi_F \tag{5.14}$$

The final component needed to determine the threshold voltage is the contact potential between point D (the bulk) and point A (the gate material) in Fig. 5.7. The potential difference between the gate and bulk (p-substrate) can be determined by summing the difference between the materials in the MOS system shown in Fig. 5.8.

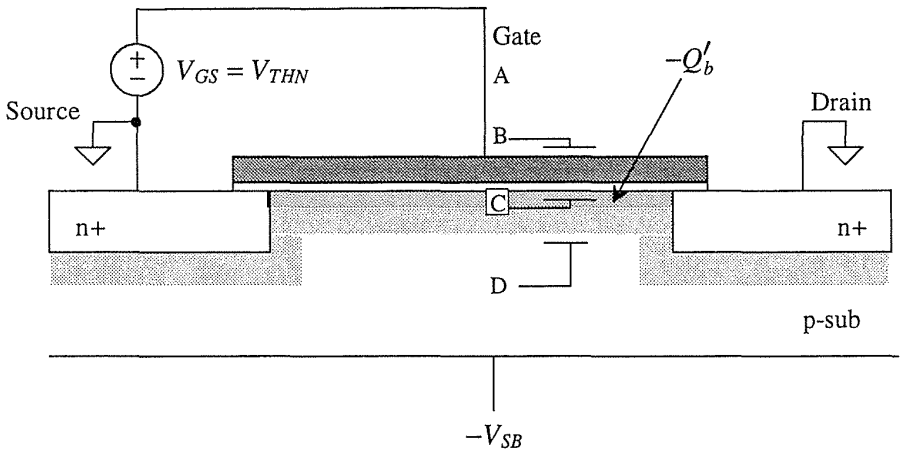



**Figure 5.7** Calculation of the threshold voltage.

Microchip Ex. 1055, p. 115
Microchip v. Aptiv
IPR2024-00494

Adding the contact potentials, we get $(\phi_G - \phi_{ox}) + (\phi_{ox} - \phi_F) = \phi_G - \phi_F$. The contact potential between the bulk and the gate poly, we will assume n+ poly with doping concentration $N_{D,\,poly}$, is given by

$$\phi_{ms} = \phi_G - \phi_F = \frac{kT}{q}\ln\left[\frac{N_{D,\,poly}}{n_i}\right] + \frac{kT}{q}\ln\frac{N_A}{n_i} \tag{5.15}$$

The threshold voltage, $V_{THN}$, is given by

$$V_{THN} = \frac{Q'_b - Q'_{ss}}{C'_{ox}} - 2\phi_F - \phi_{ms} \tag{5.16}$$

$$= -\phi_{ms} - 2\phi_F + \frac{Q'_{bo} - Q'_{ss}}{C'_{ox}} - \frac{Q'_{bo} - Q'_b}{C'_{ox}} \tag{5.17}$$

$$= -\phi_{ms} - 2\phi_F + \frac{Q'_{bo} - Q'_{ss}}{C'_{ox}} + \frac{\sqrt{2q\varepsilon_{si}N_A}}{C'_{ox}}\left[\sqrt{|2\phi_F| + V_{SB}} - \sqrt{|2\phi_F|}\right] \tag{5.18}$$

When the source is shorted to the substrate, $V_{SB} = 0$, we can define a zero substrate bias or simply zero-bias threshold voltage as

$$V_{THN0} = \phi_{ms} - 2\phi_F + \frac{Q'_{bo} - Q'_{ss}}{C'_{ox}} \tag{5.19}$$

We can define a body effect coefficient or body factor by

$$\gamma = \frac{\sqrt{2q\varepsilon_{si}N_A}}{C'_{ox}} \tag{5.20}$$

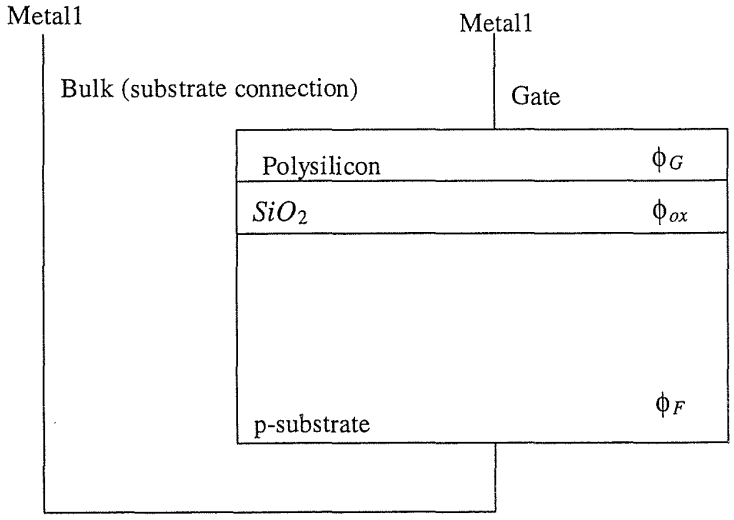



**Figure 5.8** Determining the contact potential between poly and substrate.

Microchip Ex. 1055, p. 116
Microchip v. Aptiv
IPR2024-00494

Equation (5.18) can now be written as

$$V_{THN} = V_{THN0} + \gamma\left(\sqrt{|2\phi_F| + V_{SB}} - \sqrt{|2\phi_F|}\right) \tag{5.21}$$

It is interesting to note that a voltage, called the flatband voltage $V_{FB}$, must be applied for the oxide-semiconductor interface surface potential, $\phi_s$, to become the same potential as the bulk surface potential, $\phi_F$. The flatband voltage is given by

$$V_{FB} = -\phi_{ms} - \frac{Q'_{ss}}{C'_{ox}} \tag{5.22}$$

The zero-bias threshold voltage may then be written in terms of the flatband voltage as

$$V_{THN0} = V_{FB} - 2\phi_F + \frac{Q'_{bo}}{C'_{ox}} \tag{5.23}$$

These equations describe how the threshold voltage of the MOSFET is affected by substrate doping, oxide thickness, source/substrate bias, gate material, and surface charge density.

**Example 5.3**

Assuming $N_A = 10^{15}$ atoms/cm$^3$ and $C'_{ox}$= 800 aF/μm$^2$ (CN20), estimate γ (GAMMA). Compare this estimate to the value of γ given in the SPICE level 2 n-channel model presented in Appendix A.

From Eq. (5.20) the calculated γ is

$$\gamma = \frac{\sqrt{2 \cdot 1.6\times10^{-19}\frac{\text{col}}{\text{atom}} \cdot 11.7 \cdot 8.85\frac{\text{aF}}{\mu\text{m}} \cdot 10^{15}\frac{\text{atoms}}{\text{cm}^3} \cdot \frac{\text{cm}^3}{10^{12}\mu\text{m}^3}}}{800\frac{\text{aF}}{\mu\text{m}^2}} = 0.228\ \text{V}^{1/2}$$

while the γ given in the level 2 SPICE model is 0.4179 (= GAMMA). We assume that the doping concentration of the substrate was too low. A more accurate assumption may be 3.4 × 10$^{15}$ atoms/cm$^3$. ■

**Example 5.4**

Estimate the zero-bias threshold voltage for the MOSFET of Ex. 5.2. Assume that the poly doping level is $10^{20}$ atoms/cm$^3$. What happens to the threshold voltage if sodium contamination causes an impurity of 40 aC/μm$^2$ at the oxide-semiconductor interface?

The electrostatic potential between the gate and substrate is given by

$$-\phi_{ms} = \phi_F - \phi_G = -290\ \text{mV} - 26\ \text{mV} \cdot \ln\frac{10^{20}}{14.5\times10^{9}} = -879\ \text{mV}$$

$$-2\phi_F = 580\ \text{mV}$$

Microchip Ex. 1055, p. 117
Microchip v. Aptiv
IPR2024-00494

$$\frac{Q'_{bo}}{C'_{ox}} = \frac{139\frac{\text{aC}}{\mu\text{m}^2}}{800\frac{\text{aF}}{\mu\text{m}^2}} = 173 \text{ mV}$$

$$\frac{Q'_{ss}}{C'_{ox}} = 50 \text{ mV}$$

The threshold voltage, from Eq. (5.19) without the sodium contamination, is –126 mV; with the sodium contamination the threshold voltage is –176 mV. ■

These threshold voltages would correspond to depletion devices, that is, MOSFETs that conduct when the $V_{GS}$ = 0. In CMOS applications this is highly undesirable. To compensate or adjust the value of the threshold voltage, the channel, the area under the gate poly, can be implanted with p+ ions. This will effectively increase the value of the threshold voltage by $Q'_c/C'_{ox}$ where $Q'_c$ is the charge density/unit area due to the implant. If $N_I$ is the ion implant dose in atoms/unit area, then we can write

$$Q'_c = q \cdot N_I \tag{5.24}$$

and the threshold voltage by

$$V_{THN0} = -\phi_{ms} - 2\phi_F + \frac{Q'_{bo} - Q'_{ss} + Q'_c}{C'_{ox}} \tag{5.25}$$

**Example 5.5**
Estimate the dose required to change the threshold voltage, in Ex. 5.4 without sodium contamination , to 1 V.

From Eqs. (5.24) and (5.25) and the results of Ex. 5.4

$$V_{THN0} = -126 \text{ mV} + \frac{qN_I}{C'_{ox}} = 1 \text{ V}$$

This gives $N_I$ = 563 x $10^9$ atoms/cm$^2$. ■

These calculations lend some insight into how the threshold voltage is affected by the different process parameters. In practice, the results of these calculations do not exactly match the measured threshold voltage. From a circuit design engineer's point of view, the threshold voltage and the body factor are measured in the laboratory. Measurements are performed when the BSIM model parameters (discussed in the next chapter) are extracted. The design engineer then has measured data available in the form of the BSIM model parameters.

## 5.3 IV Characteristics of MOSFETs

Now that we have some familiarity with the factors influencing the threshold voltage of a MOSFET, let's derive the large-signal IV (current/voltage) characteristics of the MOSFET, namely operation in the triode and the saturation regions. The following

Microchip Ex. 1055, p. 118
Microchip v. Aptiv
IPR2024-00494

derivation is sometimes referred to as the gradual-channel approximation. The charge distribution in the channel is assumed to be essentially constant.

### 5.3.1 MOSFET Operation in the Triode Region

Consider Fig. 5.9 where $V_{GS} > V_{THN}$, so that the surface under the oxide is inverted and $V_{DS} > 0$, causing a drift current to flow from the drain to the source. In our initial analysis, we will assume that $V_{DS}$ is sufficiently small so that the threshold voltage and the depletion layer width are approximately constant.

Initially, we must find the charge stored on the oxide capacitance $C'_{ox}$. The voltage, with respect to the source of the MOSFET, of the channel a distance $y$ away from the source is labeled $V(y)$. The potential difference between the gate electrode and the channel is then $V_{GS} - V(y)$. The charge/unit area in the inversion layer is given by

$$Q'_{ch} = C'_{ox} \cdot [V_{GS} - V(y)] \tag{5.26}$$

However, we know that a charge $Q'_b$ is present in the inversion layer from the application of the threshold voltage, $V_{THN}$, necessary for conduction between the drain and the source. This charge is given by

$$Q'_b = C'_{ox} \cdot V_{THN} \tag{5.27}$$

The total charge available in the channel, for conduction of a current between the drain and the source, is given by the difference in these two equations, or

$$Q'_I(y) = C'_{ox} \cdot (V_{GS} - V(y) - V_{THN}) \tag{5.28}$$

where $Q'_I$ is the charge in the inverted channel.

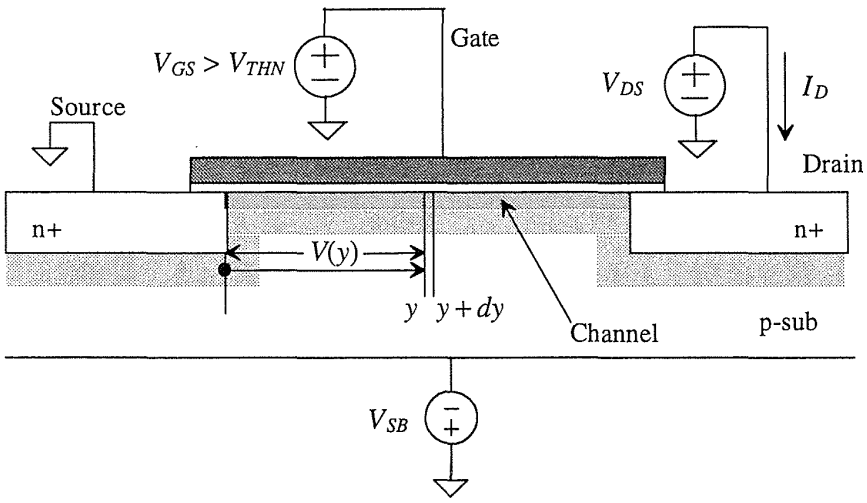



**Figure 5.9** Calculation of the large-signal behavior of the MOSFET in the triode (ohmic) region.

Microchip Ex. 1055, p. 119
Microchip v. Aptiv
IPR2024-00494

The differential resistance of the channel region with a length $dy$ and a width $W$ is given by

$$dR = \overbrace{\frac{1}{\mu_n Q_I'(y)}}^{\text{eff. sheet Res.}} \cdot \frac{dy}{W} \tag{5.29}$$

where $\mu_n$ is the average electron mobility through the channel with units of $cm^2/V \cdot sec$. The mobility is simply a ratio of the electron (or hole) velocity cm/sec to the electric field, V/cm. This parameter is used, in the BSIM model parameters, to fit the IV curves of the SPICE models to the measured results. Also, for short channel devices, the mobility decreases when the velocity of the carriers starts to saturate. This causes the effective sheet resistance in Eq. (5.29) to increase, resulting in a lowering of the drain current.

The differential voltage drop across this differential resistance is given by

$$dV(y) = I_D \cdot dR = \frac{I_D}{W\mu_n Q_I'(y)} \cdot dy \tag{5.30}$$

or substituting Eq. (5.28) and rearranging

$$I_D \cdot dy = W\mu_n C_{ox}'(V_{GS} - V(y) - V_{THN}) \cdot dV(y) \tag{5.31}$$

At this point, let's define the transconductance parameter for a MOSFET. For an n-channel MOSFET, this parameter is given by

$$KP_n = \mu_n \cdot C_{ox}' = \mu_n \cdot \frac{\varepsilon_{ox}}{TOX} \tag{5.32}$$

and for a p-channel MOSFET, it is given by

$$KP_p = \mu_p \cdot C_{ox}' = \mu_p \cdot \frac{\varepsilon_{ox}}{TOX} \tag{5.33}$$

where $\mu_p$ is the mobility of the holes in a PMOS transistor. Typical values of $KP$ in the CN20 process are 50 $\mu A/V^2$ and 17 $\mu A/V^2$ for n- and p-channel transistors, respectively.

The current can be obtained by integrating the left side of Eq. (5.31) from the source to the drain, that is, from 0 to $L$ and the right side from 0 to $V_{DS}$. This is shown below:

$$I_D \int_0^L dy = W \cdot KP_n \cdot \int_0^{V_{DS}} (V_{GS} - V(y) - V_{THN}) \cdot dV(y) \tag{5.34}$$

or

$$I_D = KP_n \cdot \frac{W}{L} \cdot \left[ (V_{GS} - V_{THN})V_{DS} - \frac{V_{DS}^2}{2} \right] \text{ for } V_{GS} \geq V_{THN} \text{ and } V_{DS} \leq V_{GS} - V_{THN} \tag{5.35}$$

This equation is valid when the MOSFET is operating in the triode (linear) region. This is the case when the induced channel extends from the source to the drain.

Microchip Ex. 1055, p. 120
Microchip v. Aptiv
IPR2024-00494

Furthermore, we can rewrite Eq. (5.35) by defining the following transconductance parameter:

$$\beta = KP_n \cdot \frac{W}{L} \tag{5.36}$$

or

$$I_D = \beta \cdot \left[ (V_{GS} - V_{THN})V_{DS} - \frac{V_{DS}^2}{2} \right] \tag{5.37}$$

The equivalent equation for the p-channel MOSFET operating in the triode region is

$$I_D = KP_p \cdot \frac{W}{L} \cdot \left[ (V_{SG} - V_{THP})V_{SD} - \frac{V_{SD}^2}{2} \right] \text{ for } V_{SG} \geq V_{THP} \text{ and } V_{SD} \leq V_{SG} - V_{THP} \tag{5.38}$$

where the threshold voltage of the p-channel MOSFET is assumed to be positive (see Appendix A). In fact, all voltages in Eqs. (5.35) and (5.38) are positive.

### 5.3.2 The Saturation Region

The voltage $V(L)$ in Eq. (5.28) is simply $V_{DS}$. In the previous subsection, we said that $V_{DS}$ is always less than $V_{GS} - V_{THN}$ so that at no point along the channel is the inversion charge zero. When $V_{DS} = V_{GS} - V_{THN}$, the inversion charge under the gate at $L$ (the drain-channel junction) is zero. This drain-source voltage is called $V_{DS,sat}$ ($= V_{GS} - V_{THN}$), and it indicates when the channel charge becomes *pinched* off at the drain-channel interface. Further increases in $V_{DS}$ do not cause an increase in the drain current[1]. Figure 5.10 shows that the depletion region between the drain and substrate increases, causing the channel to pinch off. If $V_{DS}$ is increased until the drain-substrate depletion region extends from the drain to the source, the device is said to be *punched* through. Large currents can flow under these conditions, causing device failure. The maximum voltage, for near minimum-size channel lengths, that can be applied between the drain and source of a MOSFET is set by the "punchthrough" voltage (see Appendix A). For long channel lengths, the maximum voltage is set by the breakdown voltage of the drain (n+) implant/substrate diode, also specified in Appendix A.

When a MOSFET is operated with its channel pinched off, that is, $V_{DS} \geq V_{GS} - V_{THN}$ and $V_{GS} \geq V_{THN}$, it is operating in the saturation region. Substitution of $V_{DS,sat}$ into Eq. (5.35) yields

$$I_D = \frac{KP_n}{2} \cdot \frac{W}{L} \cdot (V_{GS} - V_{THN})^2 = \frac{\beta}{2}(V_{GS} - V_{THN})^2$$
$$\text{for } V_{DS} \geq V_{GS} - V_{THN} \text{ and } V_{GS} \geq V_{THN} \tag{5.39}$$

[1] We will see that this is not entirely true. An effect called channel length modulation causes the drain current to increase with increasing drain-source voltage.

Microchip Ex. 1055, p. 121
Microchip v. Aptiv
IPR2024-00494

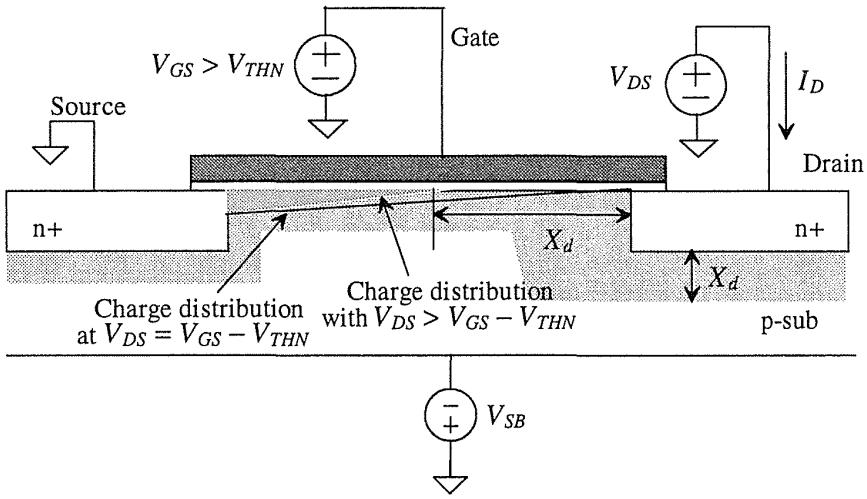



**Figure 5.10** The MOSFET in saturation (pinched off).

We can define an electrical channel length of the MOSFET as the difference between the drawn channel length, neglecting lateral diffusion, and the depletion layer width, $X_d$, between the drain n+ and the channel under the gate oxide by

$$L_{elec} = L_{drawn} - X_d \tag{5.40}$$

Substituting this into Eq. (5.39), we obtain a better representation of the drain current

$$I_D = \frac{KP_n}{2} \cdot \frac{W}{L_{elec}} (V_{GS} - V_{THN})^2 \tag{5.41}$$

Qualitatively, this means that since the depletion layer width increases with increasing $V_{DS}$ the drain current will increase as well. This effect is called *channel length modulation*. To determine the change in output current with drain-source voltage, we simply take the derivative of Eq. (5.41) with respect to $V_{DS}$, or

$$\frac{\partial I_D}{\partial V_{DS}} = -\frac{KP_n}{2} \frac{W}{L_{elec}^2} (V_{GS} - V_{THN})^2 \cdot \frac{dL_{elec}}{dV_{DS}} = I_D \cdot \left[ \frac{1}{L_{elec}} \frac{dX_d}{dV_{DS}} \right] \tag{5.42}$$

or if

$$\lambda_c = \left[ \frac{1}{L_{elec}} \cdot \frac{dX_d}{dV_{DS}} \right] \tag{5.43}$$

Typical values for $\lambda_c$, called the channel length modulation parameter, range from greater than 0.1 for short channel devices to 0.01 for long channel devices. Equation 5.41 can be rewritten, taking into account channel length modulation as

$$I_D = \frac{KP_n}{2} \cdot \frac{W}{L} (V_{GS} - V_{THN})^2 [1 + \lambda_c (V_{DS} - V_{DS,sat})] \tag{5.44}$$

Microchip Ex. 1055, p. 122
Microchip v. Aptiv
IPR2024-00494

Unless otherwise specified for digital applications, we will assume $\lambda_c = 0$. For analog applications, $\lambda_c$ is normally measured. Also note that in Eq. (5.42) we assumed that the mobility does not vary with $V_{DS}$. In the next chapter we will find that this is not true. Figure 5.11 shows typical curves for an n-channel MOSFET. Notice how the device *appears to go into saturation earlier* than predicted by $V_{DS} = V_{GS} - V_{THN}$. The actual charge distribution in the channel is not constant but rather a function of $V_{DS}$. $Q'_I(y)$ decreases as we move away from the source of the MOSFET, causing $Q'_I(L)$ to become zero earlier. This is shown in Fig. 5.10. Another reason, especially for short channel devices, why the MOSFET, becomes pinched off earlier is that the mobility is not a constant; that is, the velocity of the electrons saturates above a certain $V_{DS}$ (related to the electric field and channel length), causing $V_{DS,sat}$ and $I_{DS,sat}$ (the drain current at $V_{DS,sat}$) to decrease. Shown in Appendix A are the curves for both n- and p-channel MOSFETs of varying sizes, together with the threshold voltage and transconductance parameters for the MOSFETs in the CN20 process, which is what we will use for hand calculations throughout the book.

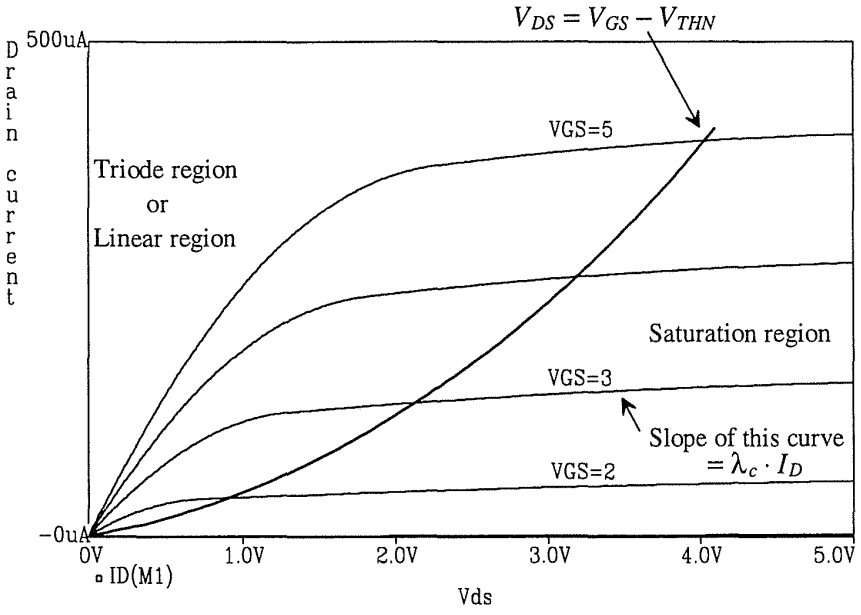



**Figure 5.11** IV characteristics of a MOSFET.

## 5.4 SPICE Modeling of the MOSFET

In this section we consider the level 1 SPICE model and how it relates to the equations derived in the last section. The level 1 model is a subset of the level 2 and 3 models.

Microchip Ex. 1055, p. 123
Microchip v. Aptiv
IPR2024-00494

### 5.4.1 Level 1 Model Parameters Related to $V_{THN}$

The following SPICE model parameters are related to the calculation of $V_{THN}$,

| Symbol | Name | Description | Default | Typ. | Units |
|---|---|---|---|---|---|
| $V_{THN0}$ | VTO | Zero-bias threshold voltage | 1.0 | 0.8 | Volts |
| $\gamma$ | GAMMA | Body-effect parameter | 0 | 0.4 | $V^{1/2}$ |
| $2\|\phi_F\|$ | PHI | Surface to bulk potential | 0.65 | 0.58 | V |
| $N_A$ | NSUB | Substrate doping | 0 | 1E15 | $cm^{-3}$ |
| $Q'_{ss}/q$ | NSS | Surface state density | 0 | 1E10 | $cm^{-2}$ |
| | TPG | Type of gate material | 1 | 1 | |

Using Eq. (5.21), we can calculate the threshold voltage, $V_{THN}$, given the above parameters. If $V_{THN0}$ or $\gamma$ is not given, then SPICE will calculate them using the above information and Eqs. (5.20), (5.22), and (5.23). TPG specifies the type of gate material, 1 opposite to substrate, -1 same as substrate, and 0 for aluminum gate.

**Example 5.6**

Using SPICE and the level 2 model given in Appendix A for the n-channel MOSFET, plot $I_D$ versus $V_{GS}$ for $V_{SB}$ changing from 0 to 5 $V$ and $V_{DS}$ = 5 $V$. Assume the device $W = L = 5$ μm.

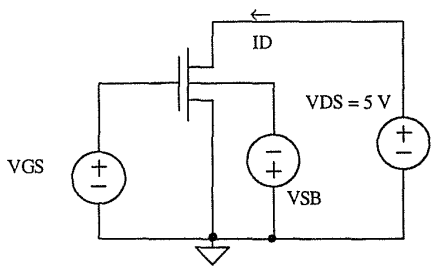
The circuit schematic, SPICE output and netlist, are shown in Figs. 5.12 and 5.13. Notice that, as we would predict from Eq. (5.21), the *threshold voltage increases* with increasing source-substrate potential, $V_{SB}$ . The results of this simulation *do not* match well with measured results. The inaccuracy of the level 1, 2, or 3 SPICE models in predicting the threshold voltage makes their use limited to basic functionality tests. The BSIM SPICE model of the next chapter should be used in most simulations (see Fig. 6.2). ■

### 5.4.2 Level 1 Model Parameters Related to Transconductance

The following SPICE parameters are related to the calculation of the transconductance parameter.

| Symbol | Name | Description | Default | Typ. | Units |
|---|---|---|---|---|---|
| $KP$ | KP | Transconductance parameter | 20E-6 | 50E-6 | $A/V^2$ |
| $t_{ox}$ | TOX | Gate-oxide thickness | 1E-7 | 40E-9 | m |
| $\lambda$ | Lambda | Channel-length modulation | 0 | 0.01 | $V^{-1}$ |
| $LD$ | LD | Lateral diffusion | 0 | 2.5E-7 | m |
| $\mu_{n,p}$ | UO | Surface mobility | 600 | 580 | $cm^2/Vs$ |

Microchip Ex. 1055, p. 124
Microchip v. Aptiv
IPR2024-00494




```
*** Top Level Netlist ***
M1_5u_5u 2 3 0 4 CMOSN  L=5u W=5u
VDS 2 0 DC 5
VGS 3 0 DC 0
VSB 0 4 DC 0

.MODEL CMOSN NMOS
(level 2 model in App. A)
.probe
.DC VGS 0 5 .05 VSB 0 5 1
.end
```

**Figure 5.12** Schematic and netlist used in Example 5.6.

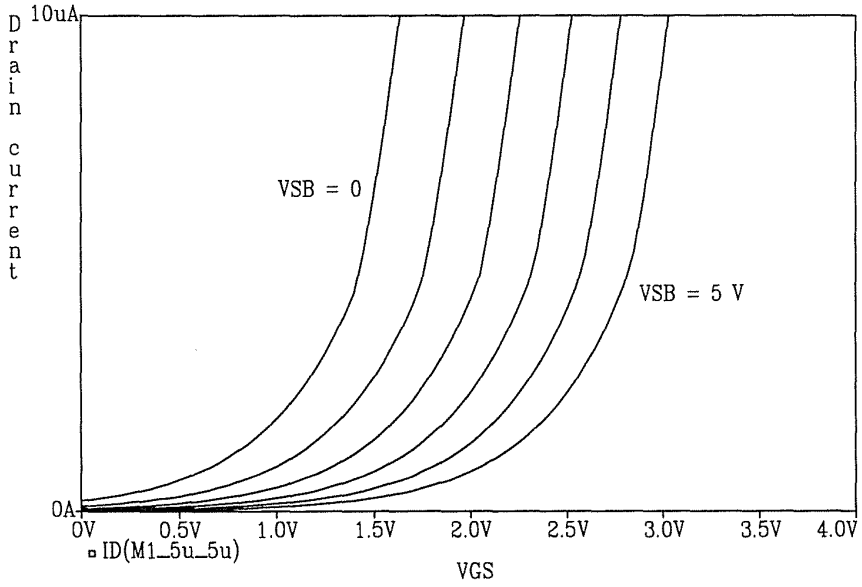



**Figure 5.13** Simulation results for Example 5.6.

The transconductance of a MOSFET is given in terms of SPICE parameters by

$$\beta = KP \cdot \frac{W_{drawn}}{L_{drawn} - 2 \cdot LD} \tag{5.45}$$

or, if *KP* is not specified in the model,

$$KP = UO \cdot \frac{\varepsilon_{ox}}{TOX} = UO \cdot C'_{ox} \tag{5.46}$$

The $\lambda$ parameter was specified above because it affects the large-signal characteristics of the MOSFETs as well as the output conductance.

Microchip Ex. 1055, p. 125
Microchip v. Aptiv
IPR2024-00494

### 5.4.3 SPICE Modeling of the Source and Drain Implants

The following SPICE model parameters are related to calculating the parasitics associated with the source/drain implant regions.

| Name | Description | Default | Typical | Units |
|---|---|---|---|---|
| RD | Drain contact resistance | 0 | 40 | Ω |
| RS | Source contact resistance | 0 | 40 | Ω |
| RSH | Source/drain sheet resistance | 0 | 50 | Ω/sq. |
| CGBO | Gate-bulk overlap capacitance | 0 | 4E-10 | F/m |
| CGDO | Gate-drain overlap capacitance | 0 | 4E-10 | F/m |
| CGSO | Gate-source overlap capacitance | 0 | 4E-10 | F/m |
| PB, PBSW | Bottom, sidewall built-in potential | 0.8 | 0.8 | V |
| MJ, MJSW | Bottom, sidewall grading coefficient | 0.5 | 0.5 | |
| CJ | Bottom zero-bias depletion capacitance | 0 | 3E-4 | $F/m^2$ |
| CJSW | Sidewall zero-bias depletion capacitance | 0 | 2.5E-10 | F/m |
| IS | Bulk-junction saturation current density | 1E-14 | 1E-14 | A |
| JS | Bulk-junction saturation current density | 0 | 1E-8 | $A/m^2$ |
| FC | Bulk-junction forward bias coefficient | 0.5 | 0.5 | |

**Example 5.7**
Specify the SPICE device statement for the layout shown in Fig. 5.14. Assume that the MOSFET level 2 model name is CMOSN and L = 5 μm and W = 3 μm.

The first parameter we must determine is the number of squares in the drain and source regions, NRD and NRS. The calculation of the number of squares is an approximation, so we neglect the n+ directly adjacent to the gate on the left and right. From the figure above the number of squares in the drain (L/W) is 12 μm/12 μm or NRD =1. For the source NRS = 0.5. If the SPICE model specifies a sheet resistance, RSH, then a resistance, NRD·RSH, due to the n+ (or p+) implant sheet resistance is added in series with the drain of the intrinsic MOSFET model. If the sheet resistance is not specified, then the values of RD or RS are added in series with the MOSFET's drain and source, respectively.

One important parameter that is not calculated here is the metal1 to n+ contact resistance. The value of NRD and NRS can be adjusted to account for this contact resistance, or an external resistor can be added. For a large *W/L* MOSFET NRD and NRS should always be set to zero.

Microchip Ex. 1055, p. 126
Microchip v. Aptiv
IPR2024-00494

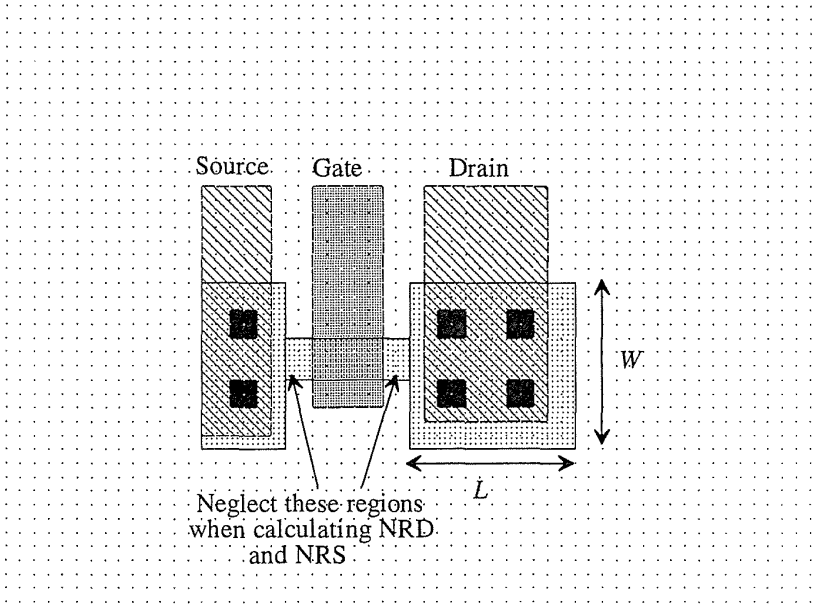



**Figure 5.14** Layout of Example 5.7.

In order for SPICE to calculate the depletion capacitances of the drain and source, we need to specify the areas and perimeters of these regions. The area of the drain is 12 μm by 12 μm or 144 $pm^2$, while the perimeter is 48 μm. The area of the source is 72 $pm^2$ and the perimeter is 36 μm. The SPICE statement for a MOSFET, M1 with its drain connected to node 1, gate to node 2, source to node 3, and substrate to node 4 is given by

```
M1 1 2 3 4 CMOSN L=5u W=3u AD=144p AS=72p PD=48u PS=36u NRD=1 NRS=0.5
```

■

### 5.4.4 Layout of the MOSFET

The layouts presented thus far for the MOSFET are for near minimum-size devices. In practice, devices with widths several hundred or even thousand μm are possible. Consider the connection of parallel MOSFETs shown in Fig. 5.15. These MOSFETs

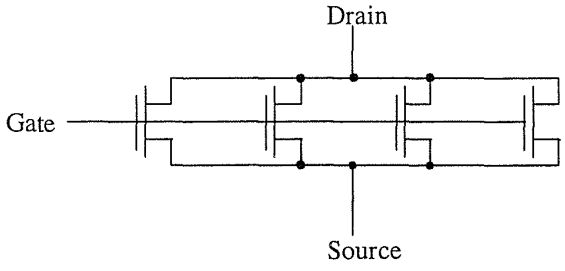



**Figure 5.15** Parallel connection of MOSFETs used for layout of a large device.

Microchip Ex. 1055, p. 127
Microchip v. Aptiv
IPR2024-00494

operate as a single MOSFET with a width equal to the sum of the individual MOSFET's widths, assuming equal lengths. The layout for the circuit of Fig. 5.15 is shown in Fig. 5.16. The drain and source areas are shared between adjacent MOSFETs. Two benefits are achieved with this layout: (1) smaller layout size and (2) reduction of source and drain depletion capacitances. The second benefit is important in analog design or in output driver design where latch-up, discussed in Ch. 11, is a concern. See Ch. 11 for further discussion of layout of large MOSFETs.

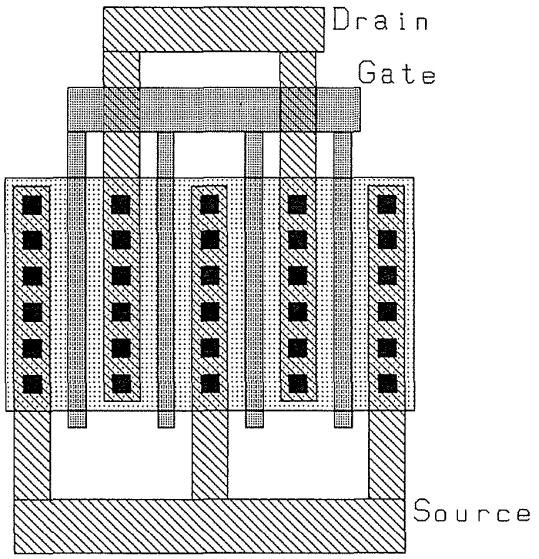



**Figure 5.16** Layout of a large (width) MOSFET.

## REFERENCES

[1] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits,* McGraw-Hill Publishing Company, 2nd ed., 1988. ISBN 0-07-029158-6.

[2] R. S. Muller and T. I. Kamins, *Device Electronics for Integrated Circuits,* 2nd ed., John Wiley and Sons, 1986. ISBN 0-471-88758-7.

[3] G. Massobrio and P. Antognetti, *Semiconductor Device Modeling with SPICE,* 2nd ed., McGraw-Hill, 1993. ISBN 0-07-002469-3.

Additional texts covering the operation and modeling of the MOSFET.

[4] Y. P. Tsividis, *Operation and Modeling of the MOS Transistor,* McGraw-Hill, 1987. ISBN 0-07-065381-X.

[5] R. F. Pierret, *Volume IV in the Modular Series on Solid State Devices-Field Effect Devices,* Addison-Wesley, 1990.

Microchip Ex. 1055, p. 128
Microchip v. Aptiv
IPR2024-00494

[6] D. K. Schroder, *Modular Series on Solid State Devices-Advanced MOS Devices,* Addison-Wesley, 1987.

[7] J. Y. Chen, *CMOS Devices and Technology for VLSI,* Prentice-Hall, 1990. ISBN 0-13-138082-6.

[8] S. M. Sze, *Physics of Semiconductor Devices,* 2nd ed., John-Wiley and Sons, 1981. ISBN 0-471-05661-8.

[9] D. A. Neamen, *Semiconductor Physics and Devices-Basic Principles,* Richard D. Irwin, 1992. ISBN 0-256-08405-X.

Texts covering MOSFET fabrication.

[10] S. Wolf, *Silicon Processing for the VLSI Era-Volume 3: The Submicron MOSFET,* Lattice Press, 1995.

[11] C. Y. Chang and S. M. Sze, *ULSI technology,* McGraw-Hill, 1996. ISBN 0-07-063062-3.

[12] R. C. Jaeger, *Modular Series on Solid State Devices-Introduction to Microelectronic Fabrication,* Addison-Wesley, 1989.

## PROBLEMS

Unless otherwise stated, use the CN20 process.

**5.1** Estimate $v_{out}$ (AC) in the following circuit, Fig. P5.1. Assume that the MOSFET was fabricated using the CN20 process.

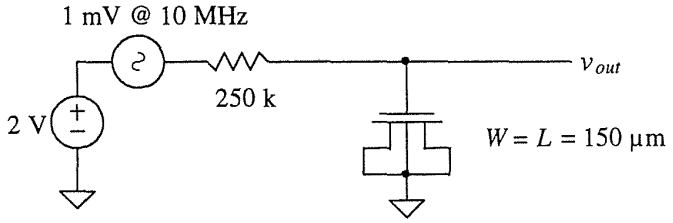



**Figure P5.1**

**5.2** An n-channel MOSFET is known to have $2|\phi_F| = 0.57$ V, $\gamma = 0.45\ V^{1/2}$, $\mu_n = 550$ cm$^2$/V·sec, and $V_{THN0} = 0.8$ V. Assume $\lambda = 0$, $n_i = 1.45E10$ atoms/cm$^{-3}$ and kT/q = 26 mV, compute the value of *KP*. Suppose *W/L* = 10/2. Compute $I_D$ when $V_{GS}$ = 2 V, $V_{SB}$ = 1 V and $V_{DS}$ = 1.1V.

**5.3** If a MOSFET is used as a capacitor in the strong inversion region where the gate is one electrode and the source/drain is the other electrode, does the gate overlap of the source/drain change the capacitance? Why? What is the capacitance?

Microchip Ex. 1055, p. 129
Microchip v. Aptiv
IPR2024-00494

**5.4** Repeat Problem 3 when the MOSFET is operating in the accumulation region. Keep in mind that the question is not asking for the capacitance from gate to substrate.

**5.5** If the oxide thickness of a MOSFET is 400 Å, what is $C'_{ox}$?

**5.6** Repeat Ex. 5.2 when $V_{SB} = 2$ V.

**5.7** Repeat Ex. 5.3 for a p-channel device with a well doping concentration of $10^{16}$ atoms/cm$^3$.

**5.8** What is the electrostatic potential of the oxide-semiconductor interface when $V_{GS} = V_{THN0}$?

**5.9** Repeat Ex. 5.5 to get a threshold voltage of 0.8 V.

**5.10** What happens to the threshold voltage in Problem 9 if sodium contamination of $100 \times 10^9$ sodium ions/cm$^2$ is present at the oxide-semiconductor interface?

**5.11** How much charge (enhanced electrons) is available under the gate for conducting a drain current at the drain-channel interface when $V_{DS} = V_{GS} - V_{THN}$? Why? Assume the MOSFET is operating in strong inversion, $V_{GS} > V_{THN}$.

**5.12** Derive Eq. (5.35) for the p-channel MOSFET following the same procedure presented in the chapter for the n-channel MOSFET.

**5.13** Show that the parallel connection of MOSFETs shown in Fig. P5.13 behave as a single MOSFET with a width equal to the sum of each individual MOSFET's width.

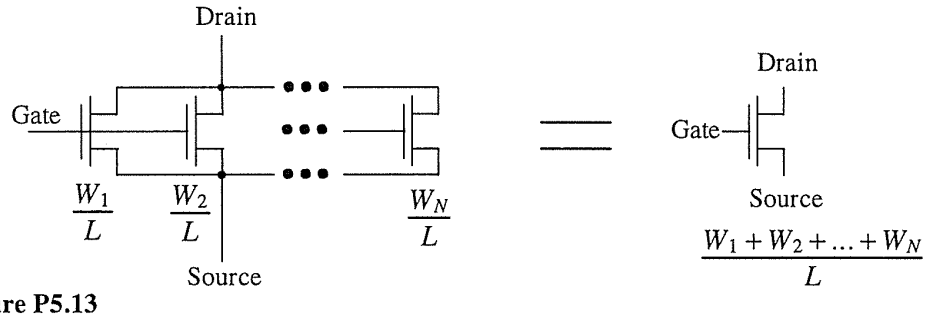



**Figure P5.13**

**5.14** Show that the bottom MOSFET, Fig. P5.14, in a series connection of two MOSFETs cannot operate in the saturation region. Neglect the body effect. [*Hint*: Show that M1 is always in either cutoff ($V_{GS1} < V_{THN}$) or triode ($V_{DS1} < V_{GS1} - V_{THN}$).]

**5.15** Show that the series connection of MOSFETs shown in Fig. P5.14 behaves as a single MOSFET with twice the length of the individual MOSFETs. Again neglect the body effect.

Microchip Ex. 1055, p. 130
Microchip v. Aptiv
IPR2024-00494

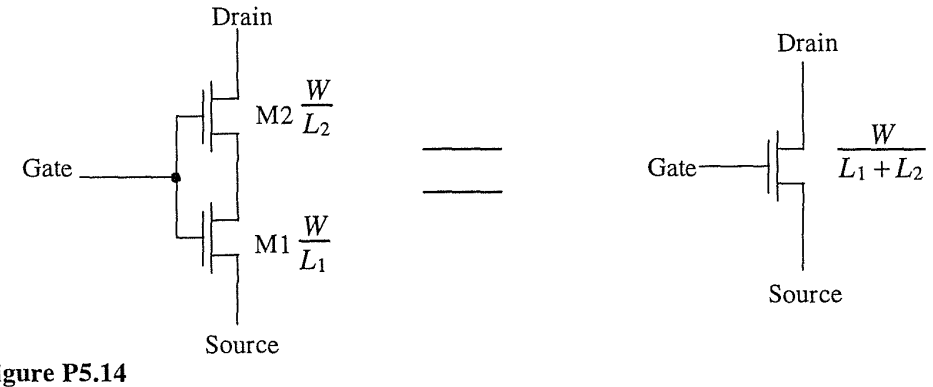



**Figure P5.14**

**5.16** Repeat Ex. 5.6 for a CN20 p-channel MOSFET.

**5.17** Lay out and DRC an n-channel MOSFET with a size of 200 μm /2 μm. Use the standard cell frame. Note that the standard cell frames can be placed end to end to achieve a larger frame; that is, the height is constant, while the width of the frame depends on the number used.

**5.18** Specify the SPICE statement for the MOSFET of Problem 17.

**5.19** The layout of an n-channel MOSFET is shown in Fig. P5.19. What is this device's width and length?

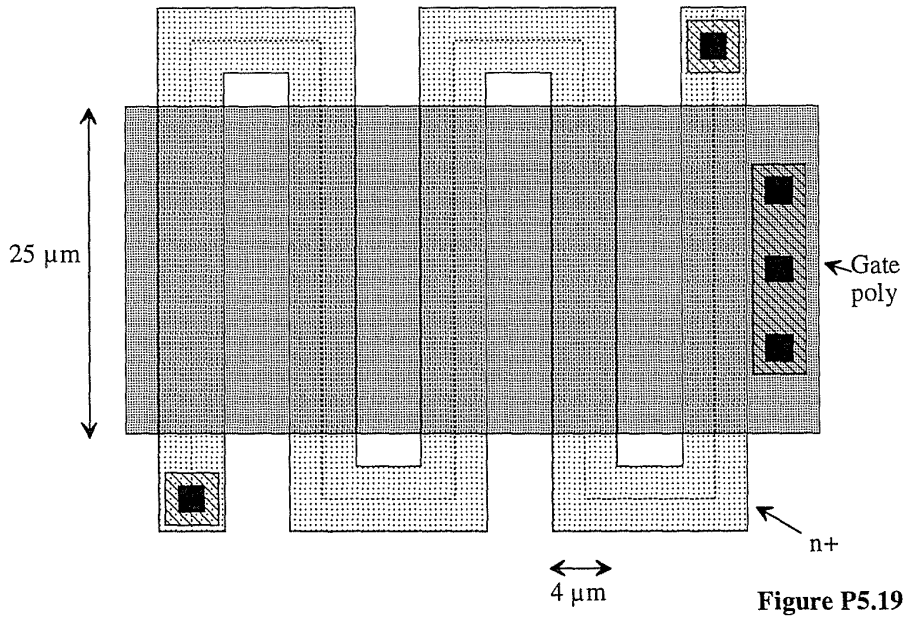



**Figure P5.19**

Microchip Ex. 1055, p. 131
Microchip v. Aptiv
IPR2024-00494

Chapter

6

# The BSIM SPICE Model

The Berkeley Short Channel Igfet (insulated gate field-effect transistor) Model, BSIM, is a SPICE model for both n- and p-channel MOS transistors. Automated parameter extraction and model generation software are used to generate the SPICE model. Most fabrication houses use this model to describe the performance of their devices. The BSIM1 SPICE model [1] is adequate for modeling MOSFETs with channel lengths down to 1 μm [2]. Shorter channel lengths require the use of the BSIM2 [2] or BSIM3 [3] model. This chapter will introduce the BSIM1 and BSIM3 models.

In the past, the level 1, 2, or 3 MOS SPICE models were discussed in texts, and thus courses covering CMOS circuit design. Since most fabricators do not supply a level 1, 2, or 3 SPICE model (MOSIS does supply a level 2 or 3 in addition to the BSIM model), students making the transition from academe to industry were met with learning the BSIM model. Furthermore, students were faced with trying to determine threshold voltage, transconductance parameter, and so on, using the unfamiliar BSIM model. In order to perform hand calculations, we need to be able to extract some of these simple parameters from the BSIM model statement. This is the goal of the chapter.

Figure 6.1 shows a comparison, using Berkeley SPICE3, between the level 2 and the BSIM1 (level 4) models for an NMOS transistor with $W$ = 3 μm and $L$ = 2 μm. Below is a SPICE netlist used to generate these plots.

```
*** Top Level Netlist ***
M1_3u_2u      3 4 0 0 CMOSN  L=2u W=3u AD=18p AS=18p PD=18u PS=18u
M2_3u_2u      5 4 0 0 CMOSNB  L=2u W=3u AD=18p AS=18p PD=18u PS=18u
Vdd     2 0     DC 0
Vgs     4 0     DC 0
VIMTR1        2 3 0V
VIMTR2        2 5 0V

.MODEL CMOSNB NMOS LEVEL=4
```

Microchip Ex. 1055, p. 132
Microchip v. Aptiv
IPR2024-00494

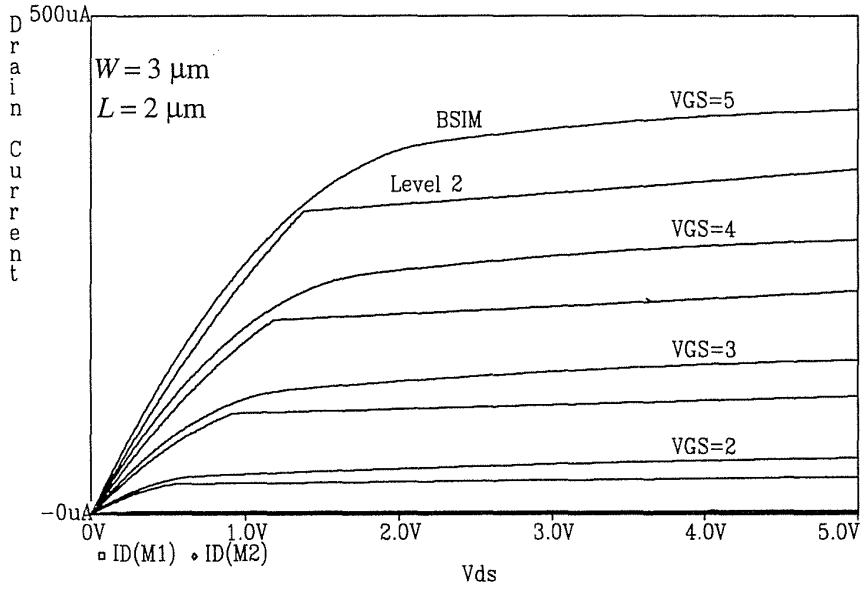



**Figure 6.1** A comparison between the Level 2 model and the BSIM1 model in the CN20 process.

```
+vfb=-9.73820E-01, lvfb=3.67458E-01,wvfb=-4.72340E-02
+phi=7.46556E-01,lphi=-1.92454E-24, wphi=8.06093E-24
+k1=1.49134E+00,lk1=-4.98139E-01, wk1=2.78225E-01
+k2=3.15199E-01,lk2=-6.95350E-02,wk2=-1.40057E-01
+eta=-1.19300E-02, leta=5.44713E-02,weta=-2.67784E-02
+muz=5.98328E+02,dl=6.38067E-001,dw=1.35520E-001
+u0=5.27788E-02, lu0=4.85686E-02,wu0=-8.55329E-02
+u1=1.09730E-01, lu1=7.28376E-01,wu1=-4.22283E-01
+x2mz=7.18857E+00,lx2mz=-2.47335E+00, wx2mz=7.12327E+01
+x2e=-3.00000E-03,lx2e=-7.20276E-03,wx2e=-5.57093E-03
+x3e=3.71969E-04,lx3e=-3.16123E-03,wx3e=-3.80806E-03
+x2u0=1.30153E-03, lx2u0=3.81838E-04, wx2u0=2.53131E-02
+x2u1=-2.04836E-02, lx2u1=3.48053E-02, wx2u1=4.44747E-02
+mus=7.79064E+02, lmus=3.62270E+02,wmus=-2.71207E+02
+x2ms=-2.65485E+00, lx2ms=3.68637E+01, wx2ms=1.12899E+02
+x3ms=1.18139E+01, lx3ms=7.24951E+01,wx3ms=-5.25361E+01
+x3u1=2.12924E-02, lx3u1=5.85329E-02,wx3u1=-5.29634E-02
+tox=4.35000E-002, temp=2.70000E+01, vdd=5.00000E+00
+cgdo=3.79886E-010,cgso=3.79886E-010,cgbo=3.78415E-010
+xpart=1.00000E+000
+n0=1.00000E+000 ln0=0.00000E+000 wn0=0.00000E+000
+nb=0.00000E+000 lnb=0.00000E+000 wnb=0.00000E+000
+nd=0.00000E+000 lnd=0.00000E+000 wnd=0.00000E+000
+rsh=27.9   cj=1.037500e-04   cjsw=2.169400e-10 js=1.000000e-08   pb=0.8
+pbsw=0.8   mj=0.66036   mjsw=0.178543   wdf=0   dell=0

.MODEL CMOSN NMOS LEVEL=2 PHI=0.600000 TOX=4.3500E-08 XJ=0.200U TPG=1
+ VTO=0.8756 DELTA=8.5650E+00 LD=2.3950E-07 KP=4.5494E-05
+ UO=573.1 UEXP=1.5920E-01 UCRIT=5.9160E+04 RSH=1.0310E+01
+ GAMMA=0.4179 NSUB=3.3160E+15 NFS=8.1800E+12 VMAX=6.0280E+04
```

Microchip Ex. 1055, p. 133
Microchip v. Aptiv
IPR2024-00494

```
+ LAMBDA=2.9330E-02 CGDO=2.8518E-10 CGSO=2.8518E-10
+ CGBO=4.0921E-10 CJ=1.0375E-04 MJ=0.6604 CJSW=2.1694E-10
+ MJSW=0.178543 PB=0.800000
.probe
.DC Vds 0 5 .1 Vgs 1 5 1
.end
```

## 6.1 BSIM1 Model Parameters

All electrical parameters, $P'$, used in the BSIM model are determined using

$$P' = P + \frac{LP}{L - DL} + \frac{WP}{W - DW} \tag{6.1}$$

$L$ and $W$ are the drawn channel length and width, while $DL$ and $DW$ are the deltalength and deltawidth resulting from lateral diffusion and encroachment discussed earlier. The differences $L - DL$ and $W - DW$ are effective channel length, and widths $L_{eff}$ and $W_{eff}$. $P$, $LP$, and $WP$ are the electrical parameters associated with the electrical parameter $P'$.

Spice BSIM1 (level 4) parameters in SPICE3.

| NAME, units | PARAMETER | W and L dependence |
|---|---|---|
| VFB, V | flat-band voltage | LVFB, WVFB |
| PHI, V | surface inversion potential | LPHI, WPHI |
| K1, $V^{1/2}$ | body effect coefficient | LK1, WK1 |
| K2, none | drain/source depletion charge-sharing coeff. | LK2, WK2 |
| ETA, none | zero-bias drain-induced barrier-lowering coeff. | LETA, WETA |
| MUZ, $cm^2/Vs$ | zero-bulk-bias mobility | |
| DL, µm | shortening of channel | |
| DW, µm | narrowing of channel | |
| U0, $V^{-1}$ | zero-bias transverse-field mobility degrad. coeff. | LU0, WU0 |
| U1, µm/V | zero-bias velocity saturation coefficient | LU1, WU1 |
| X2MZ, $cm^2/Vs$ | sens. of mobility to substrate bias at $V_{DS}$ = VDD | LX2MZ, WX2MZ |
| X2E, $V^{-1}$ | sens. of DIBL to substrate bias | LX2E, WX2E |
| X3E, $V^{-1}$ | sens. of DIBL a drain bias at Vds=Vdd | LX3E, WX3E |
| X2U0, $V^{-2}$ | sens. of trans. field mobility degrad. eff. subs. bias | LX2U0, WX2U0 |
| X2U1, $\mu m/V^2$ | sens. of velocity saturation effect to substrate bias | LX2U1, WX2U1 |
| MUS, $cm^2/Vs$ | mobility at zero substrate bias and at Vds=Vdd | LMUS, WMUS |

Microchip Ex. 1055, p. 134
Microchip v. Aptiv
IPR2024-00494

| | | |
|---|---|---|
| X2MS, $cm^2/Vs$ | sens. of mobility to substrate bias at Vds=Vdd | LX2MS, WX2MS |
| X3MS, $cm^2/Vs$ | sens. of mobility to drain bias at Vds=Vdd | LX3MS, WX3MS |
| X3U1, $\mu m/V^2$ | sens. of vel. sat. eff. on drain bias at Vds=Vdd | LX3U1, WX3U1 |
| TOX, μm | gate oxide thickness | |
| TEMP, C° | temperature at which parameters were measured | |
| VDD, V | measurement bias range | |
| CGDO, F/m | gate-drain overlap capacitance per meter channel width | |
| CGSO, F/m | gate-source overlap capacitance per meter channel width | |
| CGBO, F/m | gate-bulk overlap capacitance per meter channel length | |
| XPART | gate-oxide capacitance-charge model flag | |
| N0 | zero-bias sub threshold slope coefficient | LN0, WN0 |
| NB | sens. of sub threshold slope to substrate bias | LNB, WNB |
| ND | sens. of sub threshold slope to drain bias | LND, WND |
| RSH, Ω/sq. | drain and source diffusion sheet resistance | |
| JS, $A/m^2$ | source drain junction current density | |
| PB, V | built in potential of source drain junction | |
| MJ, none | grading coefficient of source drain junction | |
| PBSW, V | built in potential of source drain junction sidewall | |
| MJSW, none | grading coefficient of source drain junction sidewall | |
| CJ, $F/m^2$ | source drain junction capacitance per unit area | |
| CJSW, F/m | source drain junction sidewall capacitance per unit length | |
| WDF, m | source drain junction default width | |
| DELL, m | source drain junction length reduction | |

XPART = 0 selects a 40/60 drain/source charge partition in saturation, while XPART=1 selects a 0/100 drain/source charge partition. ND, NG, and NS are the drain, gate, and source nodes, respectively. MNAME is the model name, AREA is the area factor, and OFF indicates an (optional) initial condition on the device for dc analysis. If the area factor is omitted, a value of 1.0 is assumed. The (optional) initial condition specification using I=VDS, VGS is intended for use with the UIC option on the .TRAN control line, when a transient analysis is desired starting from other than the quiescent operating point.

Microchip Ex. 1055, p. 135
Microchip v. Aptiv
IPR2024-00494

*BSIM1 Model Parameters in SPICE2G6*

The BSIM model parameters used in SPICE2G6 are slightly different from the BSIM model parameters used in SPICE3. Many commercial suppliers of SPICE, including Metasofts, HSPICE, have based their models on SPICE2G6. To change the BSIM models in the file C:\LASI5\CN20\spice.inf from SPICE3 to SPICE2G6, change the following:

| | | |
|---|---|---|
| LMUS | to | LMS |
| WMUS | to | WMS |
| CJSW | to | CJW |
| JS | to | IJS |
| PB | to | PJ |
| PBSW | to | PJW |
| MJSW | to | MJW |
| dell | to | DL |

Keeping in mind the order of the variables is important. The BSIM1 model in SPICE3 is known as the level 4 model, whereas in HSPICE it is known as the level 13 model.

## 6.2 BSIM1 DC Equations

The following subsections give the equations and examples for determining the threshold voltage, drain current, and subthreshold drain current from the BSIM1 model parameters.

### 6.2.1 The Threshold Voltage

The threshold voltage is calculated in the BSIM1 model using

$$V_{THN} = \text{VFB}' + \text{PHI}' + \text{K1}' \cdot \sqrt{\text{PHI}' + V_{SB}} - \text{K2}' \cdot (\text{PHI}' + V_{SB}) - \text{ETADB}' \cdot V_{DS} \quad (6.2)$$

The voltages $V_{SB}$ and $V_{DS}$ are the substrate-to-source and drain-to-source voltages. The model parameters are calculated using Eq. (6.1). ETADB is given by

$$\text{ETADB}' = \text{ETA}' - \text{X2E}' \cdot V_{SB} + \text{X3E}' \cdot (V_{DS} - \text{VDD}) \quad (6.3)$$

It is not practical to calculate the threshold voltage by hand using Eq. (6.2). For hand calculations, assuming the device we are modeling is large so that the second and third terms in Eq. (6.1) are negligible,a simple method of determining the threshold voltage is

$$V_{THN} = \text{VFB} + \text{PHI} + \text{K1} \cdot \sqrt{\text{PHI} + V_{SB}} - \text{K2} \cdot (\text{PHI} + V_{SB}) \quad (6.4)$$

We arrived at this simplification by neglecting the effect of the device sizes and the lowered drain-induced barrier resulting from nonzero $V_{DS}$.

**Example 6.1**

Compare the exact equation (6.2) for $V_{THN}$ with the approximate equation (6.4) when the substrate potential is –5 V, the source is grounded, and the drain is at 2V. Use the BSIM1 model parameters for the CN20 process with $W/L = 3/2$.

Microchip Ex. 1055, p. 136
Microchip v. Aptiv
IPR2024-00494

First, let's calculate the simple approximation for the threshold voltage using Eq. (6.4). This is given by

$$V_{THN} = -0.97 + 0.75 + 1.49\cdot\sqrt{0.75 + 5} - 0.315\cdot(0.75 + 5) = 1.54$$

Or a generic equation in terms of $V_{SB}$ for this process is

$$V_{THN} = -0.22 + 1.49\cdot\sqrt{0.75 + V_{SB}} - 0.315\cdot(0.75 + V_{SB})$$

To calculate the threshold voltage used in the BSIM model, we calculate the primed variables in Eq. (6.2).

$$VFB' = VFB + \frac{LVFB}{L - DL} + \frac{WVFB}{W - DW} = -0.97 + \frac{0.37}{2 - 0.64} + \frac{-0.047}{3 - 0.14} = -0.713 \text{ V}$$

$$PHI' = PHI + \frac{LPHI}{L - DL} + \frac{WPHI}{W - DW} = 0.75 + \frac{0}{2 - 0.64} + \frac{0}{3 - 0.14} = 0.75 \text{ V}$$

$$K1' = K1 + \frac{LK1}{L - DL} + \frac{WK1}{W - DW} = 1.49 + \frac{-0.5}{2 - 0.64} + \frac{0.28}{3 - 0.14} = 1.212 \text{ V}^{\frac{1}{2}}$$

$$K2' = K2 + \frac{LK2}{L - DL} + \frac{WK2}{W - DW} = 0.32 + \frac{-0.07}{2 - 0.64} + \frac{-0.14}{3 - 0.14} = 0.224$$

$$ETA' = ETA + \frac{LETA}{L - DL} + \frac{WETA}{W - DW} = -0.012 + \frac{0.054}{2 - 0.64} + \frac{-0.027}{3 - 0.14} = 0.02$$

$$X2E' = X2E + \frac{LX2E}{L - DL} + \frac{WX2E}{W - DW} = -0.003 + \frac{-0.0072}{2 - 0.64} + \frac{-0.0056}{3 - 0.14} = -0.01$$

$$X3E' = X3E + \frac{LX3E}{L - DL} + \frac{WX3E}{W - DW} = -0.0004 + \frac{-0.003}{2 - 0.64} + \frac{-0.004}{3 - 0.14} = -0.003$$

$$ETADB' = 0.02 + (-0.01)(-5) + (-0.003)(2 - 5) = 0.079$$

Now we substitute these values into Eq. (6.2) to yield,

$$V_{THN} = -0.713 + 0.75 + 1.212\cdot\sqrt{0.75 + 5} - 0.224\cdot(0.75 + 5) - 0.079\cdot 2 = 1.5 \text{ V}$$

We see a difference of 40 mV for this minimum-size device between the exact calculation (6.2) and the hand calculations using Eq. (6.4). The above results should also emphasize that considerably more work is required for performing the full calculations. *These results are not representative of submicron device sizes where the BSIM1 parameters may lose their physical significance.* They may behave like curve-fitting parameters. ■

We will use Eq (6.4) for threshold calculations in this book (see Appendix A). The next example illustrates how a potential difference between the source and substrate (or well) affects the threshold voltage.

**Example 6.2**

Using the CN20 BSIM1 model parameters, plot the drain current of a minimum-size MOSFET against gate-source voltage when $V_{SB}$ varies from 0 to 5 V and the $V_{DS}$ of the MOSFET is 5 V.

Microchip Ex. 1055, p. 137
Microchip v. Aptiv
IPR2024-00494

The plot and SPICE netlist are shown below. Notice that the threshold voltage increases as the substrate potential becomes more negative (the body effect). Also notice that the threshold voltage tends to change less as *VSB* is decreased. This fact is used to get better threshold matching in analog CMOS design. The accurate representation of the body effect shown in Fig. 6.2 should be compared to the poor representation given in Fig. 5.13. ■

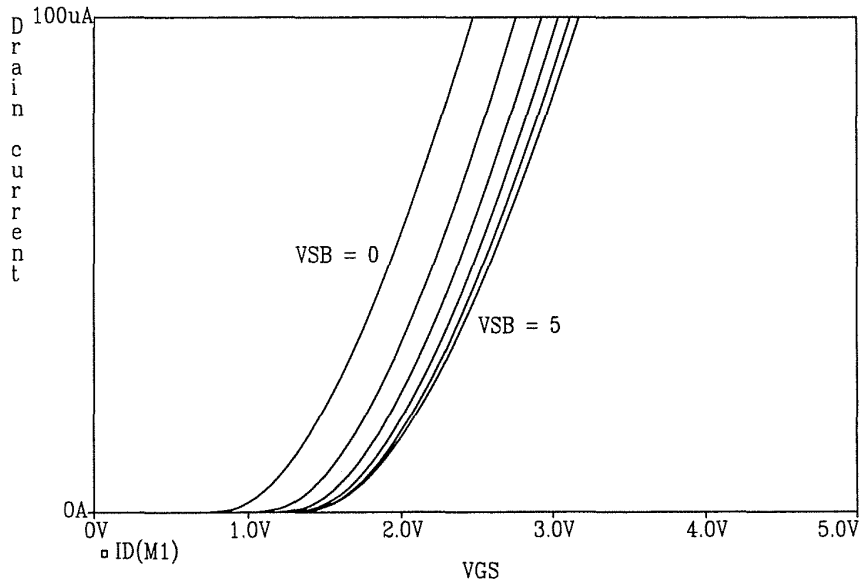



**Figure 6.2** Threshold voltage change with substrate bias.

```
*** Top Level Netlist ***
M1_3u_2u        3 4 0 5 CMOSNB  L=2u W=3u
VBB      5 0     DC 0
Vdd      2 0     DC 5
Vgs      4 0     DC 0
VIMTR1           2 3 0V
***** Spice models and macro models *****
.MODEL CMOSNB NMOS LEVEL=4
+vfb=-9.73820E-01, lvfb=3.67458E-01,wvfb=-4.72340E-02
...same model parameters as before
+pbsw=0.8    mj=0.66036    mjsw=0.178543    wdf=0    dell=0
.probe
.DC Vgs 0 2 .05 Vbb 0 -5 -1
.end
```

### 6.2.2 The Drain Current

When $V_{GS} < V_{THN}$, the MOSFET is in the cutoff region and zero drain current (neglecting the subthreshold current) flows, that is, $I_D = 0$. When $V_{DS} < V_{GS} - V_{THN} > 0$, the MOSFET is in the triode or linear region. The drain current in this region is given by

Microchip Ex. 1055, p. 138
Microchip v. Aptiv
IPR2024-00494

$$I_{DS} = \frac{MU0'}{[1 + U0Z' \cdot (V_{GS} - V_{THN})]} \cdot \frac{C'_{ox} \cdot \frac{W-DW}{L-DL}}{\left(1 + \frac{U1Z'}{L-DL} \cdot V_{DS}\right)} \cdot \left[(V_{GS} - V_{THN})V_{DS} - \frac{a}{2}V_{DS}^2\right] \quad (6.5)$$

where $C'_{ox}$, in units of F/μm², is given by

$$C'_{ox} = \frac{\varepsilon_{ox}}{TOX} \text{ and } \varepsilon_{ox} = 35.1 \times 10^{-18} \text{ F/μm} \quad (6.6)$$

$a$ is given by

$$a = 1 + \frac{g \cdot K1'}{2\sqrt{PHI' + V_{SB}}} \quad (6.7)$$

and $g$ is given by

$$g = 1 - \frac{1}{1.744 + 0.8364(PHI' + V_{SB})} \quad (6.8)$$

The parameter $MU0'$ is obtained by a quadratic interpolation through three data points: $MU0(@\ V_{DS} = 0)$, $MU0(@\ V_{DS} = VDD)$, and the sensitivity of $MU0$ around the point $V_{DS} = VDD$, or

$$MU0(@V_{DS} = 0) = MUZ' - X2MZ' \cdot V_{SB} \quad (6.9)$$

and

$$MU0(@V_{DS} = VDD) = MUS' - X2MS' \cdot V_{SB} \quad (6.10)$$

The parameters $U0Z'$ and $U1Z'$ account for mobility degradation effects and are given by

$$U0Z' = U0' - X2U0' \cdot V_{SB} \quad (6.11)$$

and

$$U1Z' = U1' - X2U1' \cdot V_{SB} + X3U1' \cdot (V_{DS} - VDD) \quad (6.12)$$

Equation (6.5) is not suitable for hand calculations. We can use the following equation for describing the IV characteristics of a MOSFET in the linear region:

$$I_{DS} = \frac{MUZ \cdot C'_{ox} \cdot W}{L}\left[(V_{GS} - V_{THN}) \cdot V_{DS} - \frac{V_{DS}^2}{2}\right] \quad (6.13)$$

Here we use $W$ and $L$ as the drawn width and length to simplify hand calculations. It should also be pointed out that $MUZ \cdot C'_{ox}$ is approximately the parameter $KP$ used in the level 1-3 models discussed in Ch. 5.

**Example 6.3**

Compare $KP$ obtained using the BSIM1 model parameters (Eq. [6.13]) for n- and p- MOSFETs, with the value of $KP$ given in the level 2 model.

Microchip Ex. 1055, p. 139
Microchip v. Aptiv
IPR2024-00494

From the SPICE listings given in the Appendix, we can determine the transconductance parameters. For the n-channel BSIM1, *KP* is given by

$$KP = MUZ \cdot C'_{ox} = (600\ cm^2/V{\cdot}s) \cdot \frac{35.1\times10^{-18}F/\mu m}{0.0435\mu m} \cdot \frac{10^8\mu m^2}{cm^2} = 48.4\ \mu A/V^2$$

This is compared with 45 $\mu A/V^2$. For the p-channel, the BSIM1 model gives a *KP* of 17 $\mu A/V^2$, while the level 2 model supplied gives 15 $\mu A/V^2$. ■

The drain current of the MOSFET in the saturation region, $V_{DS} > V_{GS} - V_{THN}$, is calculated by

$$I_{DS} = \frac{MU0'}{[1 + U0Z' \cdot (V_{GS} - V_{THN})]} \cdot \frac{C_{ox} \cdot \frac{W-DW}{L-DL}}{2aK} \cdot (V_{GS} - V_{THN})^2 \tag{6.14}$$

where

$$K = \frac{1 + v_c + \sqrt{1+2v_c}}{2} \tag{6.15}$$

and

$$v_c = \frac{U1Z'}{L-DL} \cdot \frac{(V_{GS} - V_{THN})}{a} \tag{6.16}$$

Again, Eq. (6.14) is not very useful for hand calculations. The simple solution is

$$I_{DS} = \frac{MUZ \cdot C_{ox} \cdot W}{2 \cdot L} \cdot (V_{GS} - V_{THN})^2 = \frac{KP \cdot W}{2 \cdot L} \cdot (V_{GS} - V_{THN})^2 \tag{6.17}$$

This equation will work well for DC hand calculations when $V_{DS}$ is close to $V_{GS} - V_{THN}$. However, this equation shows no $V_{DS}$ dependence. Figure 6.1 showed an increase in drain current with increasing $V_{DS}$ at a constant $V_{GS}$. We showed earlier that the drain current was dependent on the channel length modulation effect. The mobility in Eq. (6.14) is also dependent on $V_{DS}$, something we neglected earlier. We can write the mobility, *MU*0, neglecting length and width dependence by using a linear interpolation between the mobility at $V_{DS} = 0$ (*MUZ*) and the mobility at $V_{DS} = VDD$ (*MUS*) as

$$MU0 = MUZ + \frac{MUS - MUZ}{VDD} \cdot v_{DS} \tag{6.18}$$

The output resistance of the MOSFET is given by

$$r_o^{-1} = \frac{d}{dv_{DS}}\left(\frac{MU0 \cdot C_{ox} \cdot W}{2 \cdot L}(v_{GS} - V_{THN})^2\right) = \frac{dMU0}{dv_{DS}} \cdot \frac{C_{ox} \cdot W}{2 \cdot L}(v_{GS} - V_{THN})^2 \tag{6.19}$$

where the dependence of $L$ on $V_{DS}$ has been neglected. (Channel length modulation is discussed in Ch. 5.) The output resistance is then given by

$$r_o^{-1} = \left(\frac{MUS - MUZ}{VDD}\right)\frac{C_{ox}W}{2L}(v_{GS} - V_{THN})^2 = \left(\frac{MUS - MUZ}{VDD \cdot MUZ}\right)\frac{MUZ \cdot C_{ox}W}{2L}(v_{GS} - V_{THN})^2 \tag{6.20}$$

Microchip Ex. 1055, p. 140
Microchip v. Aptiv
IPR2024-00494

From Eq. (6.17) this is written as

$$r_o^{-1} = \left(\frac{MUS - MUZ}{VDD \cdot MUZ}\right) \cdot I_{DS} = \lambda_m \cdot I_{DS} \tag{6.21}$$

The parameter $\lambda_m$ is called the mobility modulation parameter. The curve-fitted mobility at $V_{DS} = VDD$ can be less than the mobility at $V_{DS} = 0$. This mobility change has the effect of increasing or decreasing the output resistance of the MOSFET depending on the operating point. When $MUS < MUZ$, Eq. (6.21) gives a negative value for $\lambda_m$, which may be close to zero. In this case, we will assume $\lambda_m = 0$, and the channel length modulation will dominate the output resistance. Equation (6.17) may now be written as

$$I_{DS} = \frac{MUZ \cdot C_{ox} \cdot W}{2 \cdot L} \cdot (V_{GS} - V_{THN})^2[1 + (\lambda_c + \lambda_m)V_{DS}] \tag{6.22}$$

where $\lambda_c$ is the channel length modulation parameter discussed earlier. This equation is very useful in analog applications where the output resistance is important. For digital applications we assume that both modulation parameters are zero.

For short channel lengths, the channel length modulation will dominate the output resistance, whereas for long channel lengths the mobility modulation parameter will dominate. It is also important to note that increasing the channel length will increase the output resistance of the MOSFET to a point. Appendix A shows, for several sizes of MOSFETs, the variation of $r_o$ with $V_{DS}$ and $V_{GS}$. Notice that $r_o$ for an n-channel MOSFET can change by a factor of 4 in some cases. Hand calculations often give ballpark answers when compared with SPICE. This is understandable given the complexity of the modern MOS device.

**Example 6.4**

Estimate the $\lambda_m$ obtained using the BSIM1 model parameters for both the n- and p-channel MOS transistors.

For the n-channel MOSFET BSIM gives $\lambda_m = \frac{779 - 598}{5 \cdot 598} = 0.061\ \text{V}^{-1}$ while for the p-channel $\lambda_m = \frac{206 - 211}{5 \cdot 211} = -0.005\ \text{V}^{-1}$ which we assume is zero.

Again, it should be emphasized that these equations are useful for hand calculations. They are heavily dependent on the measurement of the BSIM parameters. After hand calculations are performed, we can simulate the circuit using SPICE to get a more accurate representation of performance. ■

### 6.2.3 The Subthreshold Current

The current that flows when $V_{GS} < V_{THN}$ is called the subthreshold current. When the MOSFET is operating in this region, it is said to be in the weak inversion region. This current is due mainly to diffusion current between the drain and the source, similar to the bipolar junction transistor (BJT). The total drain current $I_{DS,tot}$ is the sum of the

Microchip Ex. 1055, p. 141
Microchip v. Aptiv
IPR2024-00494

weak inversion component and the strong inversion component. From the previous subsection we know that the strong inversion component is zero when $V_{GS} < V_{THN}$. The weak inversion component is modeled by

$$I_{DS,weak} = \frac{I_{exp} \cdot I_{Limit}}{I_{Limit} + I_{exp}} \tag{6.23}$$

where

$$I_{exp} = MU0' \cdot C_{ox} \cdot \frac{W - DW}{L - DL}\left(\frac{kT}{q}\right)^2 e^{1.8} e^{q(V_{GS} - V_{THN})/N'kT}(1 - e^{-qV_{DS}/kT}) \tag{6.24}$$

and

$$I_{Limit} = \frac{MU0' \cdot C_{ox}}{2} \cdot \frac{W - DW}{L - DL}\left(\frac{3kT}{q}\right)^2 \tag{6.25}$$

The subthreshold slope parameter $N'$ is given by

$$N' = N0' - NB' \cdot V_{SB} + ND' \cdot V_{DS} \tag{6.26}$$

An expression useful for hand calculations is

$$I_{exp} = MUZ \cdot C_{ox} \cdot \frac{W}{L}\left(\frac{kT}{q}\right)^2 e^{1.8} e^{q(V_{GS} - V_{THN})/N'kT}(1 - e^{-qV_{DS}/kT}) \tag{6.27}$$

When $V_{DS} > 2kT/q$ ($kT/q$ = 26 mV at room temp), Eq. (6.24) can be written as

$$I_{exp} = I_{D0} \cdot \frac{W}{L} e^{q(V_{GS} - V_{THN})/N0 \cdot kT} \tag{6.28}$$

where

$$I_{D0} = MUZ \cdot C_{ox} \cdot \left(\frac{kT}{q}\right)^2 e^{1.8} = KP \cdot \left(\frac{kT}{q}\right)^2 e^{1.8} \tag{6.29}$$

Notice the similarity to the BJT.

---

**Example 6.5**

For a CN20 NMOS transistor with $W = L = 5$ μm, use SPICE to plot $I_{DS}$ versus $V_{DS}$ for $V_{GS}$, changing from 0.65 to 0.75 V. Comment on the usefulness of Eq. (6.28).

SPICE results are shown in Fig. 6.3. Notice that the curves flatten out when $V_{DS}$ is approximately 50 mV. Although Eq. (6.28) eliminates the $V_{DS}$ dependence on the drain current, in practice the drain current increases with increasing $V_{DS}$. ■

The output resistance of a MOSFET operating in the subthreshold or strong inversion can be characterized using the early voltage parameter, $V_A$. The output resistance is given by

$$r_o = \frac{V_A + V_{DS}}{I_D} \approx \frac{V_A}{I_D} \tag{6.30}$$

Microchip Ex. 1055, p. 142
Microchip v. Aptiv
IPR2024-00494

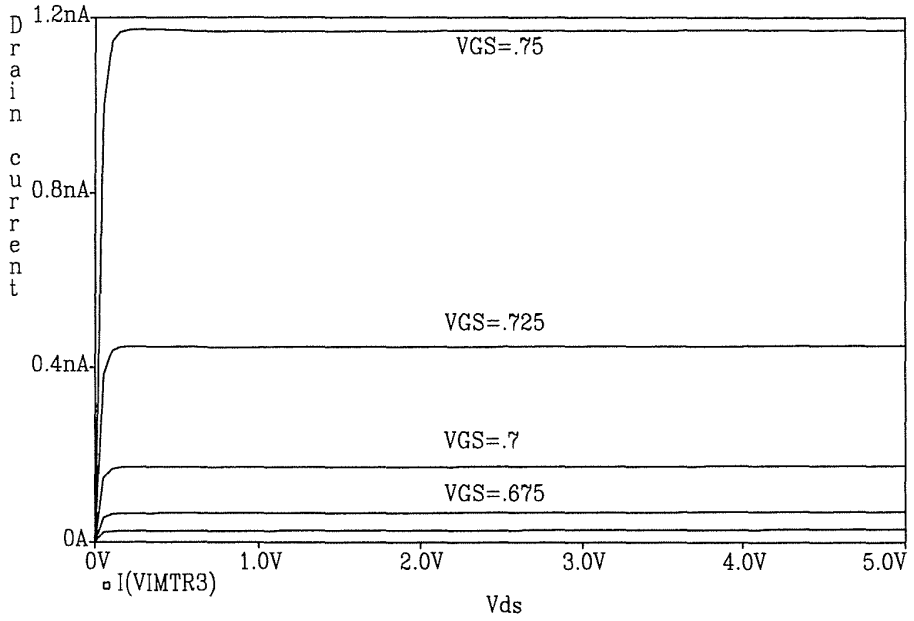



**Figure 6.3** Subthreshold characteristics of an n-channel MOSFET with $L = W = 5$ μm.

Figure 6.4 presents a graphical interpretation of this equation.

Subthreshold operation can be very useful for low-power operations. The main problem that plagues circuits designed to operate in the subthreshold region is matching. Since the drain current is exponentially related to the gate-source voltage, any mismatch in these voltages can cause significant differences in the drain current. Another problem relating to the matching of devices is the relative difference between the subthreshold slope parameter, $N'$, of different devices.

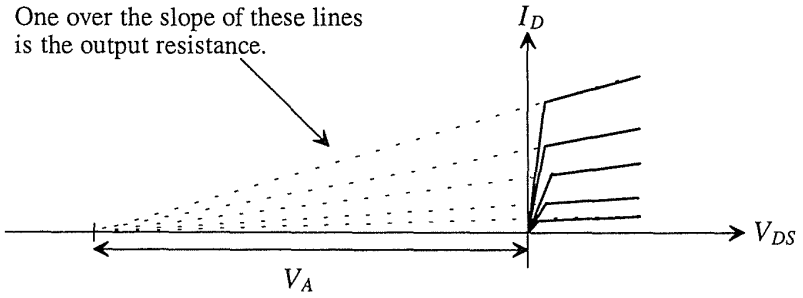



**Figure 6.4** Early voltage and its relation to the output resistance of a MOSFET.

The subthreshold region is often characterized using the log $I_D$ plotted against $V_{GS}$ (see Fig. 6.5). We can write the drain current of the MOSFET in the subthreshold region, assuming $I_{exp} << I_{Limit}$, by

Microchip Ex. 1055, p. 143
Microchip v. Aptiv
IPR2024-00494

$$I_D = I_{D0} \cdot \frac{W}{L} \cdot e^{q(V_{GS}-V_{THN})/N' \cdot kT} \tag{6.31}$$

Taking the log of both sides, we get

$$\log I_D = \log \frac{W}{L} + \log I_{D0} + \frac{q}{kT \cdot N'}(-V_{THN}) \cdot \log e + \overbrace{\left[\frac{q}{kT \cdot N'} \cdot \log e\right]}^{\text{subthreshold slope}} \cdot V_{GS} \tag{6.32}$$

The reciprocal of the subthreshold slope is given by

$$\text{sub. slope}^{-1} = \frac{kT}{q} \cdot \frac{1}{\log e} \cdot N' = \frac{kT}{q} \cdot \frac{1}{\log e} \cdot (N0' + NB' \cdot V_{BS} + ND' \cdot V_{DS}) \tag{6.33}$$

If $kT/q$ = 0.026 V = $V_T$, and $N'$ =1, the reciprocal of the subthreshold slope is 60 mV/decade. From Appendix A, Table A.5, the subthreshold slope is typically 100 mV/decade. The slope parameter $N'$ is used to adjust the model subthreshold slope so that it matches measured data, that is, measured data in the form of Fig. 6.5. Reviewing the BSIM models given in the Appendix reveals that the subthreshold region was not characterized when the models were extracted. The model uses the default 60 mV/decade, which is the result when $N0$ = 1, with the remaining eight subthreshold parameters equal to zero.

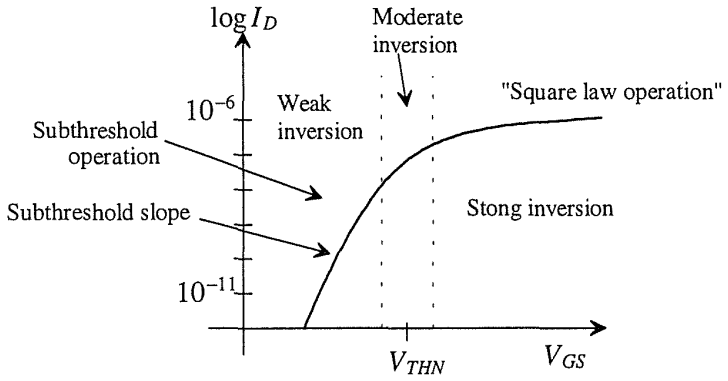



**Figure 6.5** Drain current plotted from weak to strong inversion.

## 6.3 Short Channel MOSFETs

The CN20 process used throughout the text is useful in illustrating the fundamentals of CMOS design. However, modern CMOS transistors have channel lengths that are well below the 2 μm minimum length of the CN20 process. The gradual channel approximation used in the last chapter to develop the current-voltage characteristics of the MOSFET falls apart for modern short channel devices. The electric field under the gate oxide can no longer be treated in a single dimension. In addition, the velocity of the carriers drifting between the source and drain of the MOSFET saturates, an effect

Microchip Ex. 1055, p. 144
Microchip v. Aptiv
IPR2024-00494

called carrier velocity saturation. The result is a reduction in mobility of the electron, $\mu_n$, and thus an increase in the channel's effective sheet resistance [see Eq. (5.29)]. Since the mobility of the electron decreases with increasing temperature,[1] this effect is sometimes referred to as the hot-carrier effect. A cross section of a modern MOS device using a lightly doped drain (LDD) structure, used to reduce the effects of hot carriers, is shown in Fig. 6.6. This structure is formed by implanting a shallow n-, forming the spacer adjacent to the poly, and implanting the n+.

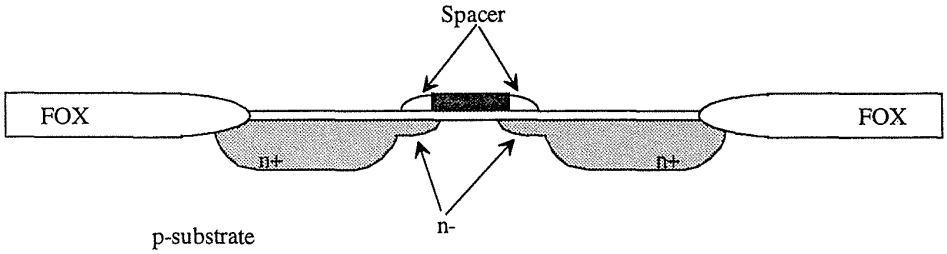



**Figure 6.6** LDD MOSFET.

### 6.3.1 MOSFET Scaling

Reducing the channel length of a MOSFET can be described in terms of scaling theory [5]. A parameter $S$ ($S < 1$) is used to scale the dimensions of a MOSFET. New technologies typically use an $S$ in the neighborhood of 0.7. If a process uses $VDD$ of 5 V, a scaled process would use a $VDD'$ of 3.5 V. In other words

$$VDD' = VDD \cdot S \tag{6.34}$$

The channel length of the scaled process is reduced to

$$L' = L \cdot S \tag{6.35}$$

while the width is reduced to

$$W' = W \cdot S \tag{6.36}$$

Table 6.1 [5] describes how $S$ affects the MOSFET parameters. The main benefits of scaling are (1) smaller device sizes and thus reduced chip size (increased yield and more parts per wafer), (2) lower gate delays, allowing higher frequency operation, and (3) reduction in power dissipation. Associated with these benefits are some unwanted side effects referred to as short channel effects. These unwanted effects are discussed in the next section.

Trends in MOSFET scaling are shown in Fig. 6.7[5]. Notice that the slopes of the curves in this figure are equal to the scaling parameter $S$.

[1] See Ch. 9 for a discussion of the temperature dependence of the mobility.

Microchip Ex. 1055, p. 145
Microchip v. Aptiv
IPR2024-00494

| Parameter | Scaling |
|---|---|
| Supply voltage (VDD) | S |
| Channel length | S |
| Channel width (W) | S |
| Gate-oxide thickness (tox) | S |
| Substrate doping (NA) | $S^{-1}$ |
| Drive current (ID) | S |
| Gate capacitance (Cox) | S |
| Gate delay | S |
| Active power | $S^3$ |

**Table 6.1** Scaling relationships.

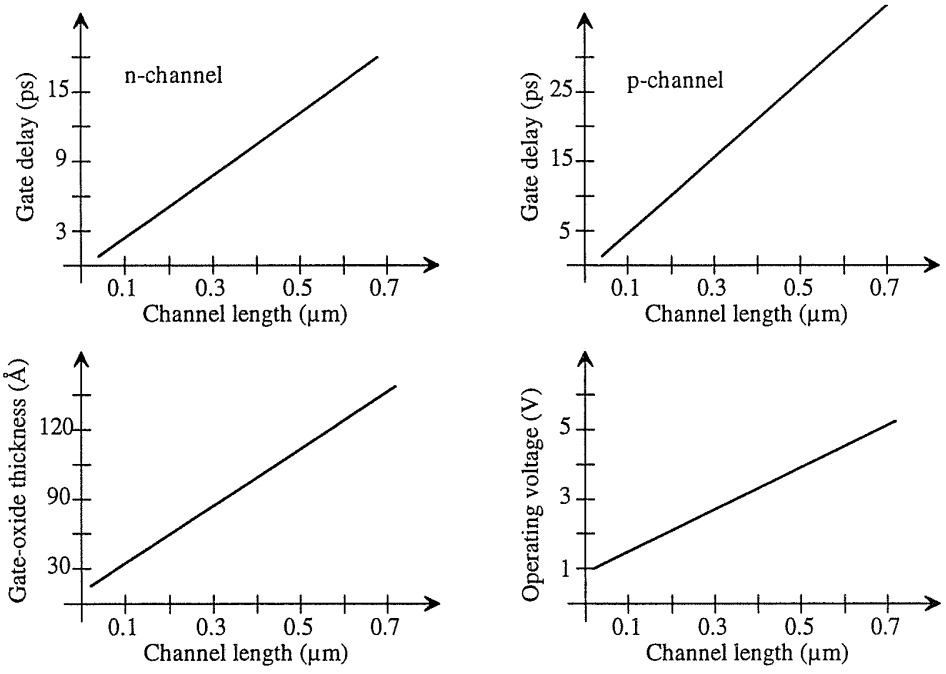



**Figure 6.7** Trends in MOS device scaling.

Microchip Ex. 1055, p. 146
Microchip v. Aptiv
IPR2024-00494

### 6.3.2 Short Channel Effects

The average drift velocity, $v$, of an electron plotted against electric field, $E$[2], is shown in Fig. 6.8. When the electric field reaches a critical value, labeled $E_{crit}$, the velocity saturates at a value $v_{sat}$, that is, the velocity ceases to increase with increasing electric field. The ratio of electron drift velocity to applied electric field is the electron mobility, or

$$\mu_n = \frac{v}{E} \tag{6.37}$$

Above the critical electric field the mobility starts to decrease, whereas below $E_{crit}$ the mobility is essentially constant. Rewriting Eq. (5.31) results in

$$I_D = \mu_n \cdot \frac{dV(y)}{dy} \cdot W \cdot C'_{ox}[V_{GS} - V_{THN} - V(y)] \tag{6.38}$$

We are interested in determining how the drain current of a short channel MOSFET changes with $V_{GS}$ when operation lies in the saturation region. (The charge under the gate oxide at the drain channel interface is zero, and the channel is pinched off.) The MOSFET enters the saturation region when $V(L) = V_{DS,sat}$. At high electric fields, the mobility can be approximated by

$$\mu_n = \frac{v_{sat}}{E} = \frac{v_{sat}}{dV(y)/dy} \tag{6.39}$$

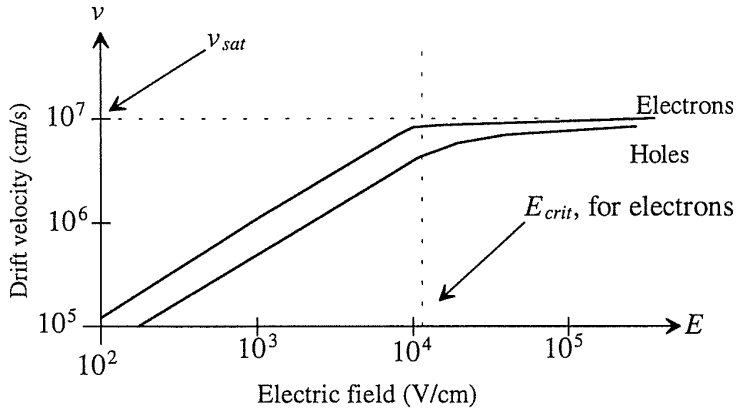



**Figure 6.8** Drift velocity plotted against electric field.

[2] A simple estimate (ballpark value) of $E$ in a MOSFET is simply $V_{DS}/L$. This value can be used to determine if short channel effects are present, that is, to determine if the carrier velocity saturates in Fig. 6.8.

Microchip Ex. 1055, p. 147
Microchip v. Aptiv
IPR2024-00494

so that Eq. (6.38) can be written as [6]

$$I_D = W \cdot v_{sat} \cdot C'_{ox}(V_{GS} - V_{THN} - V_{DS,sat}) \tag{6.40}$$

This result is important! The drain current of a short channel MOSFET operating in the saturation region increases linearly with $V_{GS}$. The long channel theory, Eq. (5.39), showed the drain current increasing with the square of the gate-source voltage. This result also presents a practical relative figure of merit for the modern CMOS process, the drive current per width of a MOSFET. The drive current, $I_{drive}$ (μA/μm), is given by

$$I_{drive} = v_{sat} \cdot C'_{ox}(V_{GS} - V_{THN} - V_{DS,sat}) \tag{6.41}$$

and therefore

$$I_D = I_{drive} \cdot W \tag{6.42}$$

The drive current can be estimated using these equations; however, it is normally measured with $V_{GS} = VDD$.

*Hot Carriers*

The low field mobility decreases with increasing temperature (again see Ch. 9). This effect can be used to describe velocity saturation, that is, assume an increase in temperature at the electric fields above $E_{crit}$. Carriers traveling at $v_{sat}$ are therefore called hot carriers. Note that the drift velocity and temperature are not proportional. *Hot carriers* is simply a term used to describe mobility reduction and thus velocity saturation.

Hot carriers can tunnel through the gate oxide and cause gate current or become trapped in the gate oxide, having the effect of changing the threshold voltage. Hot carriers can also cause impact ionization (avalanche breakdown).

*Oxide Breakdown*

For reliable device operation, the maximum electric field across a device gate oxide should be limited to 7 MV/cm [7]. This translates into 0.7 V / 10 Å of gate oxide. A device with $t_{ox}$ of 70 Å should limit the applied gate voltages to 4.9 V for reliable long-term operation.

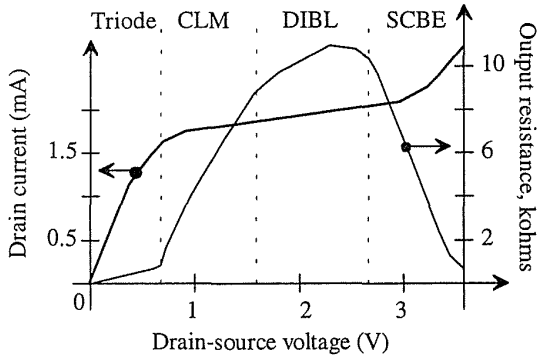
*Drain-Induced Barrier Lowering*

Drain-induced barrier lowering (DIBL, pronounced dibble) causes a threshold voltage reduction with the application of a drain-source voltage. The positive potential at the drain terminal helps to attract electrons under the gate oxide and thus increase the surface potential $\phi_s$. Since $V_{THN}$ decreases with increasing $V_{DS}$, the result is an increase in drain current and thus a decrease in the MOSFET's output resistance (see Fig. 6.9).

*Substrate Current-Induced Body Effect*

Substrate current-induced body effect, (SCBE), is the result of hot carriers causing impact ionization and generating a substrate current. This occurs for electric fields

Microchip Ex. 1055, p. 148
Microchip v. Aptiv
IPR2024-00494




CLM - Channel length modulation dominates output resistance.

DIBL - Drain-induced barrier lowering dominates output resistance.

SCBE - Substrate current -induced body effect dominates output resistance.

**Figure 6.9** Characteristics of a MOSFET with W/L = 10 μm / 0.43 μm and TOX = 75 Å [8].

greater than $10^5$ V/cm. The substrate current will flow through the resistance of the substrate, increasing the substrate potential and further decreasing the threshold voltage. The result, again, is an increase in drain current and a decrease in the MOSFETs' output resistance.

## 6.4 The BSIM3 SPICE Model

The BSIM3v3 (BSIM3 version 3) SPICE model is the third generation of BSIM MOSFET model from Berkeley [4].[3] The BSIM3 model was developed for submicron devices. It has approximately 120 possible parameters, each having physical significance such as modeling of noise and temperature effects (which were not modeled in the BSIM1 model). The drain current and its first-order derivative are continuous over all regions of operation, which helps with simulation convergence problems.

*Note that here we are using approximations to the actual equations used in the BSIM3v3 model in order to keep the introduction short. For an exact listing, see[4].*

*The Threshold Voltage Model*

The threshold voltage is determined using

$$V_{THN} = V_{THN0} + K_1\left(\sqrt{\phi_s + V_{SB}} - \sqrt{\phi_s}\right) + K_2 V_{SB} + K_1\left(\sqrt{1 + \frac{N_{LX}}{L_{eff}}} - 1\right)\sqrt{\phi_s} - \Delta V_{THN} \tag{6.43}$$

where $V_{THN0}$ is the long channel threshold voltage, $\phi_s$ (= PHI) is the surface potential, $\Delta V_{THN}$ is the threshold voltage reduction due to short channel effects, and $K_1$, $K_2$ take into account vertical nonuniform doping effects. The threshold voltage reduction has an exponential dependence on channel length and is given by

$$\Delta V_{THN} = \theta(L) \cdot [2(V_{bi} - \phi_s) + V_{DS}] \tag{6.44}$$

[3] This section is based directly on the information given in [4].

Microchip Ex. 1055, p. 149
Microchip v. Aptiv
IPR2024-00494

and

$$\theta(L) = D_{VTHN0}[e^{-(L\cdot DVT1)/2l_t} + 2e^{-(L\cdot DVT1)/l_t}] \tag{6.45}$$

where $V_{bi}$ is the built-in potential given by

$$V_{bi} = \frac{kT}{q}\cdot \ln\left(\frac{N_{ch}N_d}{n_i^2}\right) \tag{6.46}$$

and $l_t$ is the characteristic length given by

$$l_t = \sqrt{\frac{\varepsilon_{si}t_{ox}X_d}{\varepsilon_{ox}\eta}} \tag{6.47}$$

The depletion width between the channel and the substrate is $X_d$, while the average depletion width along the channel is $X_d/\eta$. The parameters $D_{VTHN0}$, $DVT1$, and $\eta$ can be determined from experimental data.

*The Drain Current Model*

The operating regions in the BSIM3 model are divided into strong inversion ($V_{GS} > V_{THN}$), weak inversion ($V_{GS} < V_{THN}$), and a transition region ($V_{GS} \approx V_{THN}$). The strong inversion region is divided into the triode region ($V_{DS} < V_{DS,sat}$) and the saturation region ($V_{DS} > V_{DS,sat}$). The drain current in the saturation region is given by

$$I_D = W\cdot v_{sat}\cdot C'_{ox}(V_{GS} - V_{THN} - V_{DS,sat})\left[1 + \frac{V_{DS} - V_{THN} - V_{DS,sat}}{V_A}\right] \tag{6.48}$$

where $V_A$ is the early voltage generated by the MOSFET output generated by the output resistance model. The drain current in the triode region is estimated using

$$I_D = W\cdot \mu_{eff}\cdot C'_{ox}\cdot \frac{W}{L}\cdot \frac{1}{1+\frac{V_{DS}/L}{E_{crit}}}\cdot \left[\frac{V_{GS} - V_{THN} - V_{DS}}{2}\right]\cdot V_{DS} \tag{6.49}$$

where the effective mobility is estimated using

$$\mu_{eff} = 2\cdot \frac{v_{sat}}{E_{crit}} \tag{6.50}$$

The drain current in the weak inversion or subthreshold region is estimated using

$$I_D = \mu_{eff}\cdot \frac{W}{L}\cdot \left[\frac{kT}{q}\right]^2\cdot C_d\cdot \left[e^{(V_{GS}-V_{THN}-V_{off})q/nkT}\right][1 - e^{-qV_{DS}/kT} \tag{6.51}$$

where $V_{off}$ is an offset voltage and $n$ is the subthreshold slope factor.

*Model Examples*

The BSIM3v3 model is implemented in SPICE3f5 as a level 8 MOSFET model. Metasoft's HSPICE implements their version of the BSIM3v3 model as a level 49 MOSFET model. Below are example model listings [9] for both an n-channel and a p-channel MOSFET model based on BSIM3v3.

Microchip Ex. 1055, p. 150
Microchip v. Aptiv
IPR2024-00494

```
*model = bsim3v3
*Berkeley Spice Compatibility
* Lmin= .35 Lmax= 20 Wmin= .6 Wmax= 20
.model N1 NMOS
+Level=      8
+Tnom=27.0
+Nch= 2.498E+17  Tox=9E-09 Xj=1.00000E-07
+Lint=9.36e-8 Wint=1.47e-7
+Vth0= .6322    K1= .756  K2= -3.83e-2  K3= -2.612
+Dvt0= 2.812  Dvt1= 0.462  Dvt2=-9.17e-2
+Nlx= 3.52291E-08  W0= 1.163e-6
+K3b= 2.233
+Vsat= 86301.58  Ua= 6.47e-9  Ub= 4.23e-18  Uc=-4.706281E-11
+Rdsw= 650  U0= 388.3203 wr=1
+A0= .3496967 Ags=.1    B0=0.546    B1= 1
+ Dwg = -6.0E-09 Dwb = -3.56E-09 Prwb = -.213
+Keta=-3.605872E-02  A1= 2.778747E-02  A2= .9
+Voff=-6.735529E-02  NFactor= 1.139926  Cit= 1.622527E-04
+Cdsc=-2.147181E-05
+Cdscb= 0  Dvt0w =  0 Dvt1w =  0 Dvt2w =  0
+ Cdscd =  0 Prwg =  0
+Eta0= 1.0281729E-02  Etab=-5.042203E-03
+Dsub= .31871233
+Pclm= 1.114846  Pdiblc1= 2.45357E-03  Pdiblc2= 6.406289E-03
+Drout= .31871233  Pscbe1= 5000000  Pscbe2= 5E-09 Pdiblcb = -.234
+Pvag= 0 delta=0.01
+ Wl =  0 Ww = -1.420242E-09 Wwl =  0
+ Wln =  0 Wwn =  .2613948 Ll =  1.300902E-10
+ Lw =  0 Lwl =  0 Lln =  .316394
+ Lwn =  0
+kt1=-.3  kt2=-.051
+At= 22400
+Ute=-1.48
+Ua1= 3.31E-10  Ub1= 2.61E-19 Uc1= -3.42e-10
+Kt1l=0 Prt=764.3

.model P1 PMOS
+Level=      8
+Tnom=27.0
+Nch= 3.533024E+17  Tox=9E-09 Xj=1.00000E-07
+Lint=6.23e-8 Wint=1.22e-7
+Vth0=-.6732829 K1= .8362093  K2=-8.606622E-02  K3= 1.82
+Dvt0= 1.903801  Dvt1= .5333922  Dvt2=-.1862677
+Nlx= 1.28e-8  W0= 2.1e-6
+K3b= -0.24 Prwg=-0.001 Prwb=-0.323
+Vsat= 103503.2  Ua= 1.39995E-09  Ub= 1.e-19  Uc=-2.73e-11
+ Rdsw= 460  U0= 138.7609
+A0= .4716551 Ags=0.12
+Keta=-1.871516E-03  A1= .3417965  A2= 0.83
+Voff=-.074182  NFactor= 1.54389  Cit=-1.015667E-03
+Cdsc= 8.937517E-04
+Cdscb= 1.45e-4  Cdscd=1.04e-4
+ Dvt0w=0.232 Dvt1w=4.5e6 Dvt2w=-0.0023
+Eta0= 6.024776E-02  Etab=-4.64593E-03
```

Microchip Ex. 1055, p. 151
Microchip v. Aptiv
IPR2024-00494

```
+Dsub= .23222404
+Pclm= .989  Pdiblc1= 2.07418E-02  Pdiblc2= 1.33813E-3
+Drout= .3222404  Pscbe1= 118000  Pscbe2= 1E-09
+Pvag= 0
+kt1= -0.25  kt2= -0.032 prt=64.5
+At= 33000
+Ute= -1.5
+Ua1= 4.312e-9 Ub1= 6.65e-19  Uc1= 0 Kt1l=0
```

## 6.5 Convergence

A major frustration with using the BSIM SPICE model is the difficulty with convergence. *Assuming* the circuit contains no connection errors, there are basically three parameters that can be adjusted to help convergence. (Other methods of helping convergence for a particular type of analysis are discussed elsewhere in the text.) These three parameters are ABSTOL, VNTOL, and RELTOL [10].

ABSTOL is the absolute current tolerance. Its default value is 1 pA. This means that when a simulated circuit gets within 1 pA of its "actual" value, SPICE assumes that the current has converged and moves onto the next time step or AC/DC value. VNTOL is the node voltage tolerance, default value of 1 μV. RELTOL is the relative tolerance parameter, default value of 0.001 (0.1 percent). RELTOL is used to avoid problems with simulating large and small electrical values in the same circuit. For example, suppose the default value of RELTOL and VNTOL were used in a simulation where the actual node voltage is 1 V. The RELTOL parameter would signify an end to the simulation when the node voltage was within 1 mV of 1 V (1V·RELTOL), while the VNTOL parameter signifies an end when the node voltage is within 1 μV of 1 V. SPICE uses the larger of the two, in this case the RELTOL parameter results, to signify that the node has converged.

Increasing the value of these three parameters helps speed up the simulation and assists with convergence problems at the price of reduced accuracy. To help with convergence, the following statement can be added to a SPICE netlist:

```
.OPTIONS ABSTOL=1uA VNTOL=1mV RELTOL=0.01
```

To force convergence, these values can be increased to

```
.OPTIONS ABSTOL=1mA VNTOL=100mV RELTOL=0.1
```

The following is a list of common mistakes that can be made when learning to simulate circuits using SPICE.

1. The first line in a SPICE netlist must be a comment line. SPICE ignores the first line in a netlist file.
2. One megaohm is specified using 1MEG, not 1M, 1m, or 1 MEG.
3. One farad is specified by 1, not 1f or 1F. 1F means one femto-farad or $10^{-15}$ farads.

Microchip Ex. 1055, p. 152
Microchip v. Aptiv
IPR2024-00494

4. Areas of MOSFET drains and sources are, in most cases, specified in terms of picometers squared. An area measuring 6 μm on one side and 8 μm on the other side is specified by 48pm or 48E-12.

5. Voltage source names should always be specified with a first letter of V. Current source names should always start with an I.

6. Transient simulations display time data; that is, the x-axis in probe is time. A jagged plot such as a sinewave that looks like a triangle wave or is simply not smooth is the result of not specifying a maximum print step size (in the .tran statement) or specifying a maximum print step that is too large. For example, in order to display a 1 kHz sinewave in SPICE, a maximum print step of 10u (ten microeconds) can be specified.

7. When displaying the results of an AC simulation where the x-axis is frequency, the probe cursor should be displaying either the magnitude or phase of the voltage or current. For example, a cursor that displays "voltage drop at a node" will sum the real and imaginary parts of the voltage at that node and display the somewhat meaningless result. The probe functions vary widely between simulators. (Some simulators are smart enough to start the user in the magnitude mode after performing an AC simulation.)

8. MOSFET widths and lengths are specified using the letter "u" for microns. A common mistake is to forget to include this letter. For example, a minimum-size MOSFET in the CN20 process is specified by L=2u W=3u not L=2 W=3. The latter means a MOSFET with a 2 meter length and a 3 meter width!

9. In general, the body connection of a p-channel MOSFET is connected to *VDD*, and the body connection of an n-channel MOSFET is connected to *VSS*. For example, in the CN20 process which is an n-well process, all n-channel bulk connections must be tied to *VSS*. This matter is easily checked in the SPICE netlist.

10. Convergence in a DC sweep can often be helped by avoiding the power supply boundaries. For example, sweeping a circuit from 0 to 5 V may not converge, but sweeping from 0.1 to 4.9 will.

**REFERENCES**

[1] J. R. Pierret, *A MOS Parameter Extraction Program for the BSIM Model*, Electronics Research Laboratory, University of California, Berkeley, Calif. 94720. Memorandum No. UCB/ERL M84/99, November 21, 1984.

[2] G. Massobrio and P. Antognetti, *Semiconductor Device Modeling with SPICE*, 2nd ed., McGraw-Hill, 1993. ISBN 0-07-002469-3.

[3] J. H. Huang, Z. H. Liu, M. C. Jeng, P. K. Ko, and C. Hu, *A Robust Physical Predictive Model for Deep-Submicron MOS Circuit Simulator,* Electronics

**Microchip Ex. 1055, p. 153**
**Microchip v. Aptiv**
**IPR2024-00494**

Research Laboratory, University of California, Berkeley, Calif. 94720. Memorandum No. UCB/ERL M93/57, July 21, 1993.

[4] BSIM3v3 Manual (final version) from the Department of Electrical Engineering and Computer Science, University of California, Berkeley. Also can be downloaded at http://rely.EECS.Berkeley.EDU:8080/bsim3www/.

[5] M. Bohr, "MOS Transistors: Scaling and Performance Trends," *Semiconductor International*, pp. 75-79, June 1995.

[6] K. Y. Toh, P. K. Ko, and R. G. Meyer, "An Engineering Model for Short-Channel MOS Devices," *IEEE Journal of Solid State Circuits*, Vol. 23, No. 4, August 1988.

[7] C. Hu, *ULSI Device Scaling and Reliability,* Research and Development seminar at Micron Semiconductor, Boise, Idaho, 1995.

[8] J. H. Huang, Z. H. Liu, M. C. Jeng, P. K. Ko, and C. Hu, *A Physical Model for MOSFET Output Resistance,* Electronics Research Laboratory, University of California, Berkeley, Calif. 94720. Memorandum No. UCB/ERL M93/56, July 21, 1993.

[9] http://rely.EECS.Berkeley.EDU:8080/bsim3www/

[10] R. Kielkowski, *Inside SPICE: Overcoming the Obstacles of Circuit Simulation,* McGraw-Hill, 1994. ISBN 0-07-911525-X.

## PROBLEMS

**6.1** Using the CN20 BSIM model parameters for the p-channel MOSFET, calculate, by hand, the zero-bias threshold voltage.

**6.2** Calculate $KP$ and $C'_{ox}$ for both the n- and p-channel MOSFETs using the CN20 BSIM model parameters.

**6.3** Using SPICE, plot $I_D$ against $V_{DS}$ ($V_{SD}$) for both the n- and p-channel MOSFETs ($L$ = 2 μm and $W$ = 3 μm) using the CN20 BSIM model parameters for $V_{GS}$ ($V_{SG}$) = 1, 2, 3, 4, and 5 V. The SPICE models are located in the file spice.inf in the C:\LASI5\CN20 directory.

**6.4** Using SPICE and the BSIM model-generate Fig. 6.4 for an n-channel MOSFET with $L = W$ = 10 μm. What is the subthreshold slope?

**6.5** Calculate $I_{drive}$ for a short channel MOSFET with $t_{ox}$ = 75 Å, $V_{THN}$ = 0.5 V, $V_{GS}$ = $VDD$ = 2.5 V and $V_{DS,sat}$ = 1.5 V. Comment on why $V_{DS,sat}$ is not equal to $V_{GS} - V_{THN}$ for a short channel MOSFET (see Ch. 5).

**6.6** Estimate the mobility of the electron and hole at $10^5$ V/cm from Fig. 6.8.

**6.7** Show that using the level 1 SPICE MOSFET model results in a small-signal output resistance of

Microchip Ex. 1055, p. 154
Microchip v. Aptiv
IPR2024-00494

$$r_o = \frac{1}{\lambda I_D} = \frac{L}{\lambda \cdot K}$$

where $K$ is essentially constant for small changes in $V_{GS}$. This equation shows that the output resistance of a MOSFET is linearly dependent on the channel length for a constant drain current. Increasing the channel length of the MOSFET increases the output resistance. This result, from the discussion in this chapter, is *incorrect.* (The level 1 models neglect $\lambda_m$.) Consider the results of Problem. 8.

**6.8** Using the results of an .OP (operating point analysis), plot the output resistance of a MOSFET versus channel length for the circuit of Fig. P6.8 using the BSIM model for channel lengths from 2 to 10 μm. The SPICE output file should specify the MOSFET small-signal parameters after the simulation has run. The MOSFET output resistance is the reciprocal of the parameter "gds" in this file.

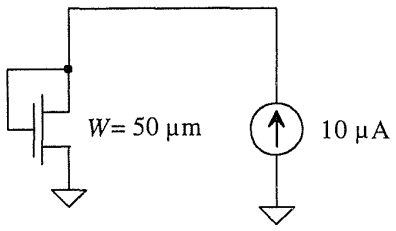



**Figure P6.8**

**6.9** Calculate $KP$ and $C'_{ox}$ for both the n- and p-channel MOSFETs using the CMOS14TB BSIM model parameters.

**6.10** Using SPICE, plot $I_D$ against $V_{DS}$ ($V_{SD}$) for both the n- and p-channel MOSFETs ($L$ = 0.6 μm and $W$ = 0.9 μm) using the CMOS14TB BSIM model parameters for $V_{GS}$ ($V_{SG}$) = 1, 2, and 3 $V$. The SPICE models are located in the file spice.inf in the C:\LASI5\CN20 directory.

**6.11** Calculate $I_{drive}$ for both the n- and p-channel MOSFETs in the CMOS14TB process.

**6.12** Label Field Oxide, the spacer for LDD, gate, drain/source and lateral diffusion in the scanning electron microscope (SEM) picture of Fig. P6.12.

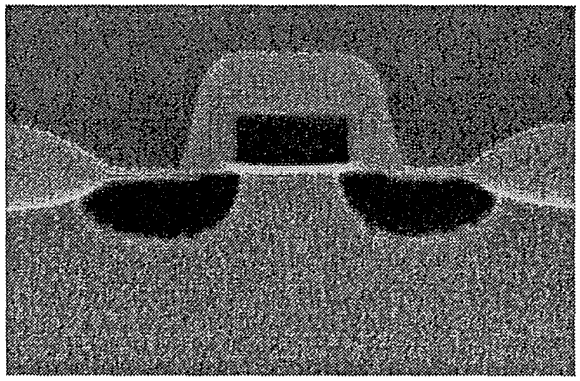


**Figure P6.12**

Microchip Ex. 1055, p. 155
Microchip v. Aptiv
IPR2024-00494

Chapter

7

# CMOS Passive Elements

In this chapter we discuss the layout of the poly1-poly2 capacitor, as well as the temperature and voltage dependence of the resistor and capacitor, and finally we introduce noise analysis of resistor-capacitor circuits.

## 7.1 The Second Poly Layer (poly2)

Often, in a CMOS process, a second layer of polysilicon is added for MOSFET and capacitor formation. This poly2 layer follows similar design rules as the poly1 layer. The layer for contact to poly2 is the "contact" layer ("cont" in the LASI CN20 setups). We can use poly2 for MOSFET formation instead of poly1. The electrical characteristics of the MOSFET made with poly2 are given in the CN20 parametric data available from MOSIS.

### 7.1.1 Design Rules for Capacitor Formation

The design rules specified here are derived from the Orbit design rules. The design rules for capacitor formation are as follows.

| | |
|---|---|
| 4.1 Width | 3 μm |
| 4.2 Space | 3 μm |
| 4.3 Poly1 overlap of poly2 | 2 μm |
| 4.4 Space to active or well edge | 2 μm |
| 4.5 Space to contact | 3 μm |

These rules are illustrated in Fig. 7.1 along with an example cross-sectional view of the capacitor.

Microchip Ex. 1055, p. 156
Microchip v. Aptiv
IPR2024-00494

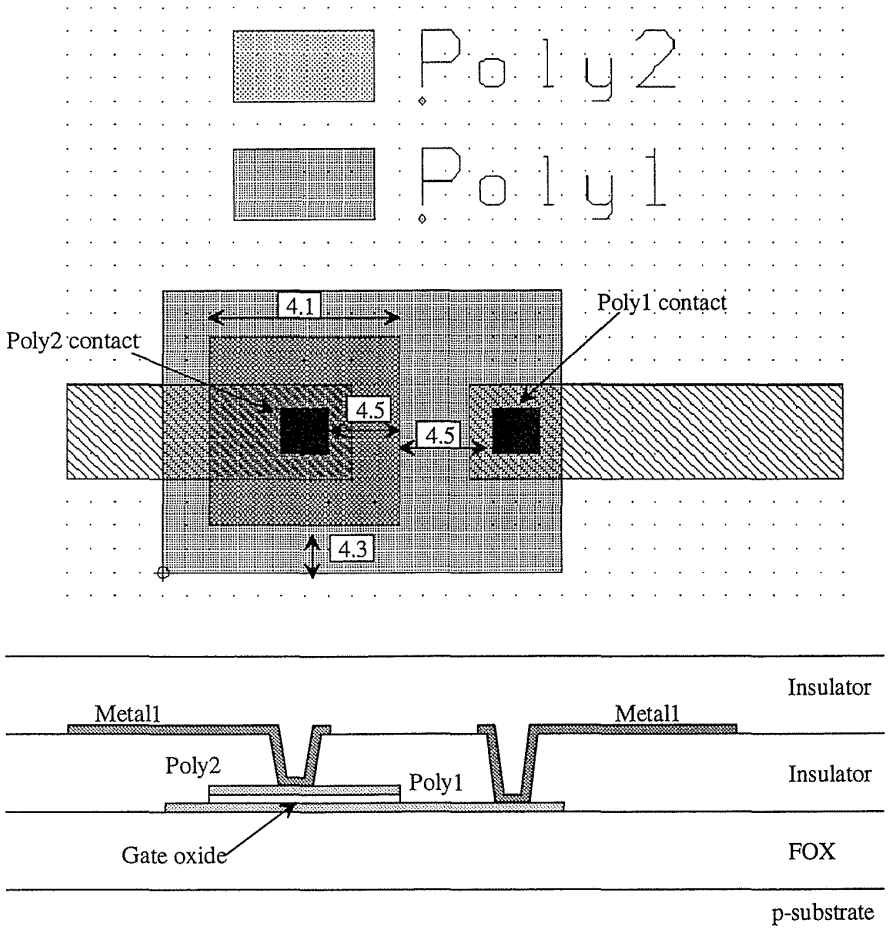



**Figure 7.1** Design Rules for Poly1-Poly2 capacitor and associated cross-sectional view.

**Example 7.1**
Estimate the value of the capacitor shown in Fig. 7.1.

From the Orbit electrical parameters, the capacitance between poly1 and poly2 can vary from a minimum of 443 aF/μm$^2$ to 557 aF/μm$^2$. The amount of overlap of poly1 and poly2 shown in the figure is 64 μm$^2$. So the capacitance can range from 28.4 fF to 35.6 fF. ■

### 7.1.2 Parasitics of the Poly Cap

In addition to the capacitance between poly1 and poly2, we also have a capacitance between poly1 and substrate, poly1 (or poly2) and metal1, and possibly metal2 to poly1, poly2, or metal1. These capacitances are shown in Fig. 7.2. When we are laying out capacitors, we should not waste much time trying to analyze each component, but should just try to get an idea for the approximate values.

Microchip Ex. 1055, p. 157
Microchip v. Aptiv
IPR2024-00494

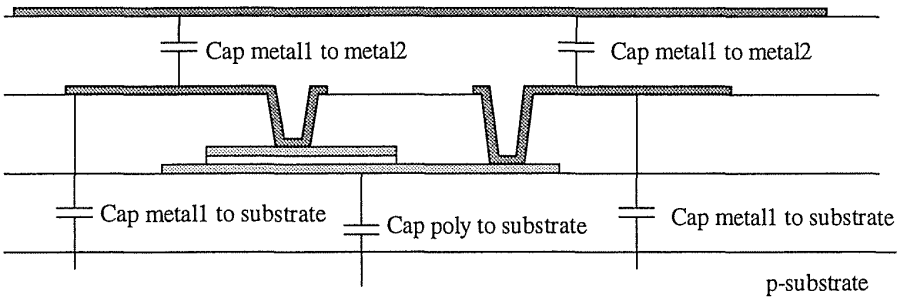



**Figure 7.2** Parasitic capacitance associated with poly1-poly2 capacitor.

Another parasitic associated with the capacitor is the contact resistance, metal1 to poly, and the distributed resistance across the plate of the capacitor. To minimize the resistance, contacts should be made everywhere possible on poly. Since we only have access to poly1 on the end, we might want to contact the poly1 down the center or in a finger style similar to laying out a large MOSFET. An example layout is shown in Fig. 7.3. We know that with every contact to poly we have a contact resistance. We think of this as a resistor between metal1 and the plate of the capacitor. Adding another contact halves this resistance because the contact resistances are in parallel. Also, the sheet resistance of poly is on the order of 20 ohms/square. The plate of the capacitor is then

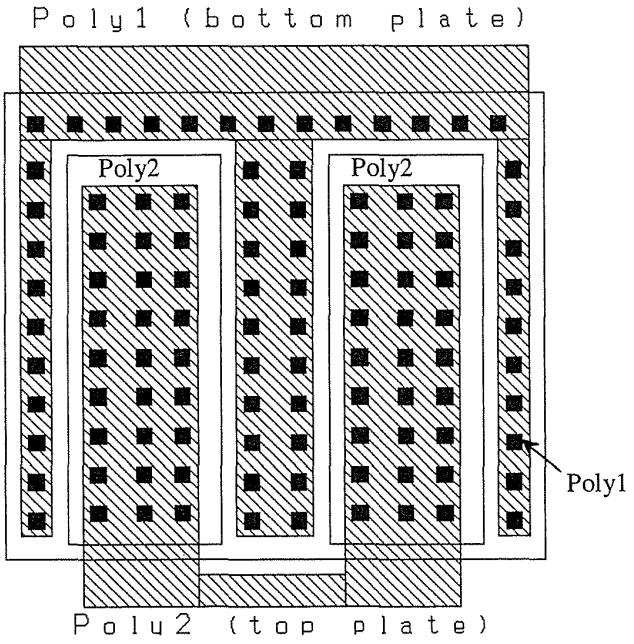



**Figure 7.3** Layout of a 1 pF capacitor using multiple contacts and fingers to reduce the series resistance.

Microchip Ex. 1055, p. 158
Microchip v. Aptiv
IPR2024-00494

actually a distributed *RC*. The *R* is from the resistance of the poly, while the capacitance is the actual capacitance we are interested in.

### 7.1.3 Other Types of Capacitors

As we discussed earlier, we can make a capacitor using metal1 and metal2, a MOSFET with the drain and source shorted with $V_{GS} > V_{THN}$ (operating in the strong inversion region), and we can use the n+ or p+ diffusion regions to substrate or well.

## 7.2 Temperature and Voltage Dependence of Capacitors and Resistors

The resistor and capacitor in a CMOS process have a value that changes with temperature and voltage. The change is usually listed ppm/°C (parts per million per degree C). The ppm/°C is equivalent to a multiplier of $10^{-6}$/°C.

### 7.2.1 Resistors

The first-order temperature coefficient of a resistor, *TCR*, is given by

$$TCR = \frac{1}{R} \cdot \frac{dR}{dT} \tag{7.1}$$

Typical values of *TCR* for n-well, n+/p+ (active) and poly resistors are 10,000, 2,000, and 1,000ppm/°C respectively. SPICE uses a quadratic, in addition to this first order term, to model the temperature dependence of a resistor

$$R(T) = R_{T0} \cdot [1 + TCR \cdot (T - T_0) + TCR2 \cdot (T - T_0)^2] \tag{7.2}$$

where $R_{T0}$ is the value of the resistor at $T_0$, typically measured at 27 °C. For hand calculations, we will assume *TCR*2 is 0.

---

**Example 7.2**

Estimate the minimum and maximum resistance of an n-well resistor with a length of 100 μm and a width of 10 μm over a temperature range of 0 to 100 °C. Assume that the *TCR* of the resistor is 10,000 ppm/°C.

The n-well resistor sheet resistance can vary from 2,000 to 3,000 Ω/square. The value of the resistor in this example can change from 20 k to 30 kΩ at 27 °C. The minimum resistance is then determined, using Eq. (7.2) by

$$R_{\min} = 20,000 \cdot [1 + 0.01 \cdot (0 - 27)] = 14.6\ \text{k}\Omega$$

and the maximum resistance is determined by

$$R_{\max} = 30,000 \cdot [1 + 0.01 \cdot (100 - 27)] = 51.90\ \text{k}\Omega \quad \blacksquare$$

The two resistors discussed in the previous example would correspond to resistors fabricated in two different wafers. Typical matching of n-well resistors on the same wafer is around 1 percent. For a 25 k n-well resistor we might expect another resistor on the same die with the same size to have a value between 24,750 and 25,250

Microchip Ex. 1055, p. 159
Microchip v. Aptiv
IPR2024-00494

ohms. Again, this is an approximation. Temperature, voltage across, and place on the die all have an effect. If the resistor is placed close to a power amplifier which is dissipating large amounts of power, the resistor will effectively be at a higher temperature than the same-size resistor on the other side of the die.

Another important contributor to a changing resistance is the voltage coefficient given by

$$VCR = \frac{1}{R} \cdot \frac{dR}{dV} \tag{7.3}$$

where $V$ is the average voltage applied to the resistor, the sum of the voltages on each end of the resistor divided by two. For an n-well resistor, this is the voltage difference between each side of the resistor and the potential of the n-well. The resistance as a function of voltage is then given by

$$R(\mathrm{V}) = R_{V0} \cdot (1 + VCR \cdot \mathrm{V}) \tag{7.4}$$

The value $R_{V0}$ is the value of the resistor at the temperature $T_0$, again typically 27 °C. A typical value of $VCR$ is 200 ppm/$V$ for an n-well resistor. The main contributor to the voltage coefficient is the depletion layer width between the n-well and the p-substrate. The depletion layer extends into the n-well, resulting in an effective increase in the sheet resistance. The thickness of the n-well available to conduct current decreases with increasing potential between the n-well and the substrate.

**Example 7.3**

Estimate the average resistance of an n-well resistor with a typical value of 50 k at 27 °C, for an average voltage across the resistor of 0, 5, and 10 V.

$$R(0) = 50,000 \cdot (1 + 0.0002 \cdot 0) = 50\ k\Omega$$

$$R(5) = 50,000 \cdot (1 + 0.0002 \cdot 5) = 50.05\ k\Omega$$

$$R(10) = 50,000 \cdot (1 + 0.0002 \cdot 10) = 50.10\ k\Omega$$

This is a small change compared to the change due to temperature. However, as the next example will show, the change due to the applied voltage can have a greater effect on the circuit performance than the temperature. ■

**Example 7.4**

Compare the change in $V_{out}$ in the following circuit due to $VCR$ with the change due to $TCR$. Assume that the resistors are fabricated using the n-well.

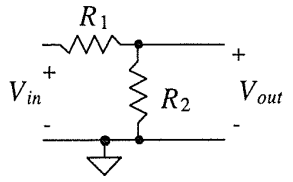



**Figure Ex7.4**

Microchip Ex. 1055, p. 160
Microchip v. Aptiv
IPR2024-00494

We start by writing the voltage divider equation and then substituting the temperature or voltage dependence. For the temperature dependence

$$V_{out} = V_{in} \cdot \frac{R2(\mathrm{T})}{R1(\mathrm{T}) + R2(\mathrm{T})} = V_{in} \cdot \frac{R2 \cdot [1 + TCR(\mathrm{T} - \mathrm{T}_0)]}{(R1 + R2) \cdot [1 + TCR(\mathrm{T} - \mathrm{T}_0)]} = V_{in} \cdot \left[\frac{R2}{R1 + R2}\right]$$

and for the voltage dependence

$$V_{out} = V_{in} \cdot \frac{R2(\mathrm{V})}{R1(\mathrm{V}) + R2(\mathrm{V})} = V_{in} \cdot \frac{R2 \cdot (1 + VCR \cdot \mathrm{V}_2)}{R1 \cdot (1 + VCR \cdot \mathrm{V}_1) + R2 \cdot (1 + VCR \cdot \mathrm{V}_2)}$$

where $\mathrm{V}_1 = \frac{Vin + Vout}{2}$ and $\mathrm{V}_2 = \frac{Vout}{2}$ assuming the substrate is at ground potential. These results show that the voltage divider has no temperature dependence, to a first order, while it does have a voltage dependence. ■

The preceding example should also show the advantage of ratioing components. For precision design over large temperature ranges, this fact is very important. Usually, a unit resistor is laid out with some nominal resistance. Figure 7.4a shows a unit cell resistor with a nominal resistor value of 20 kΩ. Figure 7.4b shows how the nominal 20 kΩ unit cell resistor is used to implement a nominally 100 kΩ resistor. The errors due to corners and differing perimeters are eliminated with this scheme when the ratio of the resistors is used in a circuit. Amplifier circuits, such as those using op-amps with feedback, are examples of circuits where ratioing is important.

Any precision circuit is susceptible to substrate noise. Substrate noise is the result of adjacent circuits injecting current into one another. The simplest method of

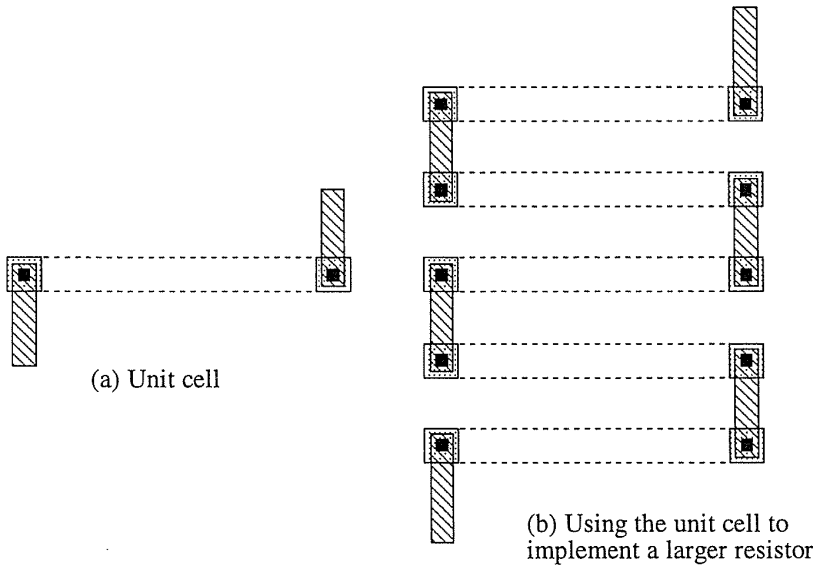



**Figure 7.4** Using a unit cell in resistor layout.

Microchip Ex. 1055, p. 161
Microchip v. Aptiv
IPR2024-00494

reducing substrate noise between adjacent circuits is to place a p+ implant (for a p-substrate wafer) between the two circuits that is tied to *VSS*. The diffusion removes the injected carriers and holds the substrate, ideally, at a fixed potential (*VSS*). Figure 7.5 shows the basic idea for a resistor. The p+ implant guards the circuit against minority carrier injection. Since the implants are laid out in a ring, they are often called *guard rings*.

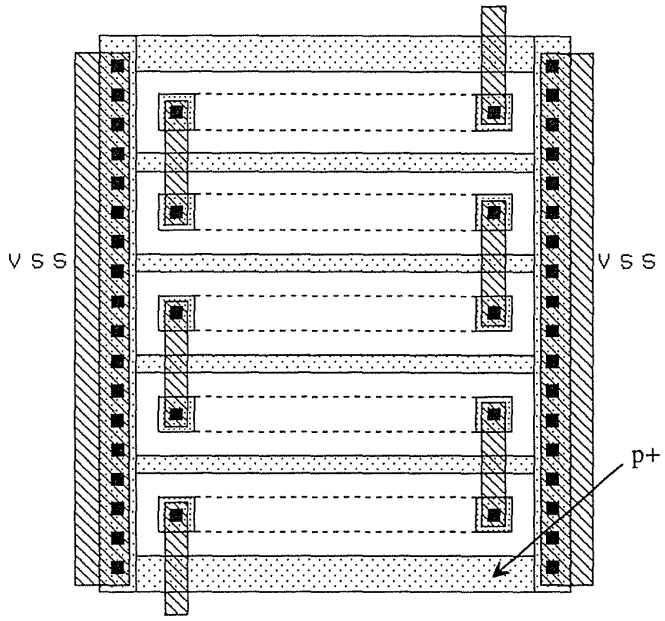



**Figure 7.5** Layout of a resistor using guard rings.

Matching between two different resistors can be benefited by using the layout shown in Fig. 7.6. Process gradients, in this case changes in the n-well doping at different places on the die, are spread between the two resistors. Notice that the orientation of the resistors is consistent between unit cells; that is, all cells are laid out either horizontally or vertically. In a practical matched resistor layout, the unit cells in Fig. 7.6 would be surrounded by guard rings. Also, each resistor has essentially the same parasitics.

*Common-Centroid Layout*

Common-centroid (common center) layout can be used to help improve the matching between two resistors [2] (at the cost of uneven parasitics between the two elements). Consider the arrangement of unit resistors shown in Fig. 7.7a. This arrangement is sometimes called an interdigitated resistor and is the same form as that shown in Fig. 7.6. Furthermore, consider the effects of a linear varying sheet resistance on the overall value of each resistor. If we assign a normalized value to each unit resistor, as shown,

Microchip Ex. 1055, p. 162
Microchip v. Aptiv
IPR2024-00494

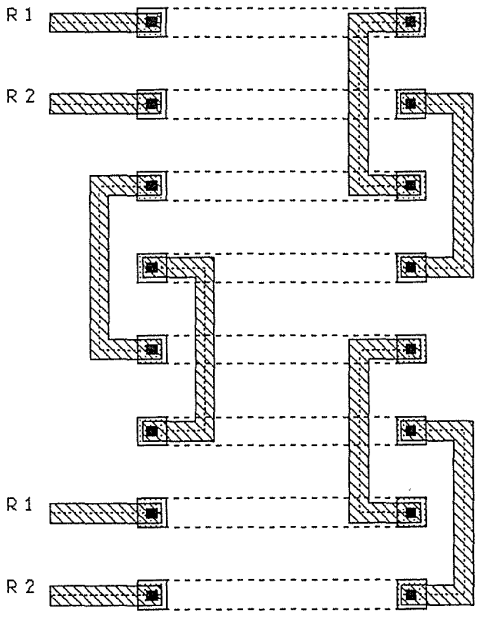



**Figure 7.6** Layout of two matched resistors. Note that the parasitic capacitance from metal1 to substrate is essentially the same for each resistor.

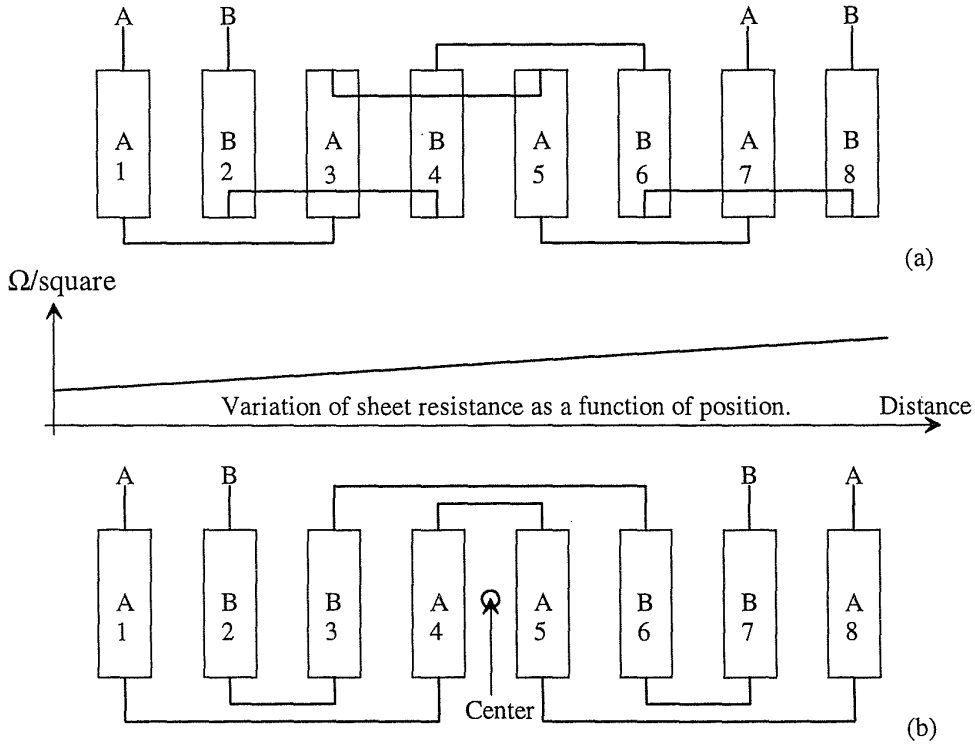



**Figure 7.7** (a) Interdigitated layout and (b) common-centroid layout.

Microchip Ex. 1055, p. 163
Microchip v. Aptiv
IPR2024-00494

then resistor A has a value of 16 and resistor B has a value of 20. Ideally, the resistor values are equal.

Next consider the common-centroid layout shown in Fig. 7.7b, noting that resistors A and B share a common center. The value of either resistor in this figure is 18. In other words, the use of a common center (ABBAABBA) will give better matching than the interdigitized layout (ABABABAB). Figure 7.8 shows the common-centroid layout (two different possibilities) of four matched resistors. Note that common-centroid layout can also improve matching in MOSFETs or capacitors.

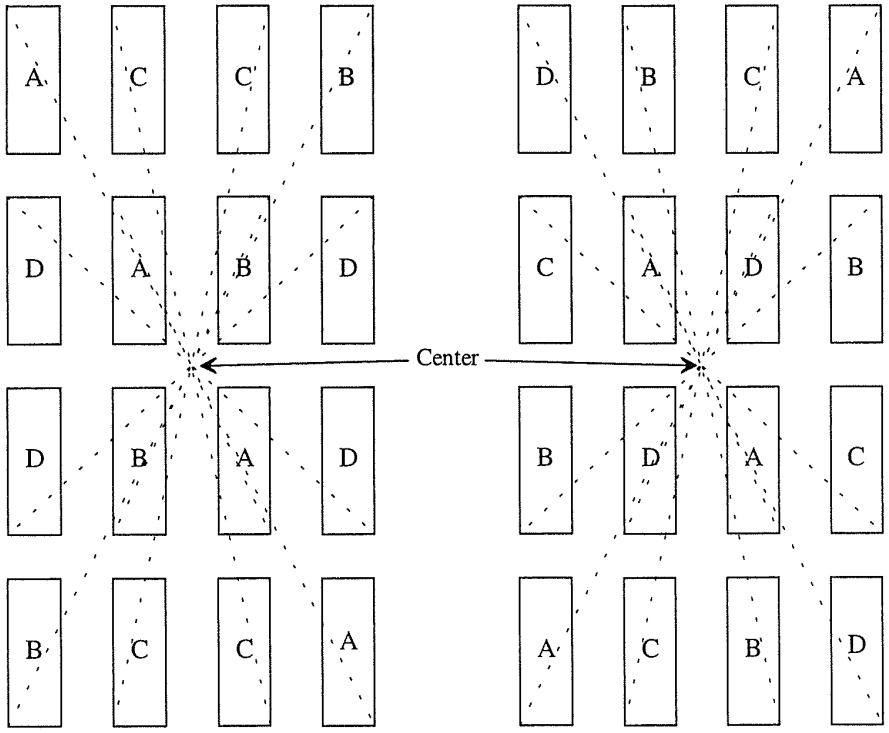



**Figure 7.8** Common-centroid layout of four matched resistors (or elements).

*Dummy Elements*

Another technique useful in improving the matching between two or more elements is the use of dummy elements. Consider the cross-sectional views of n-well shown in Fig. 7.9a. The final amount of diffusion under the resist, on the edges, is different between the outer and the inner unit cells. This is the result of differing dopant concentrations at differing points on the surface (during the diffusion process). This difference will result in a mismatch between unit resistors.

Microchip Ex. 1055, p. 164
Microchip v. Aptiv
IPR2024-00494

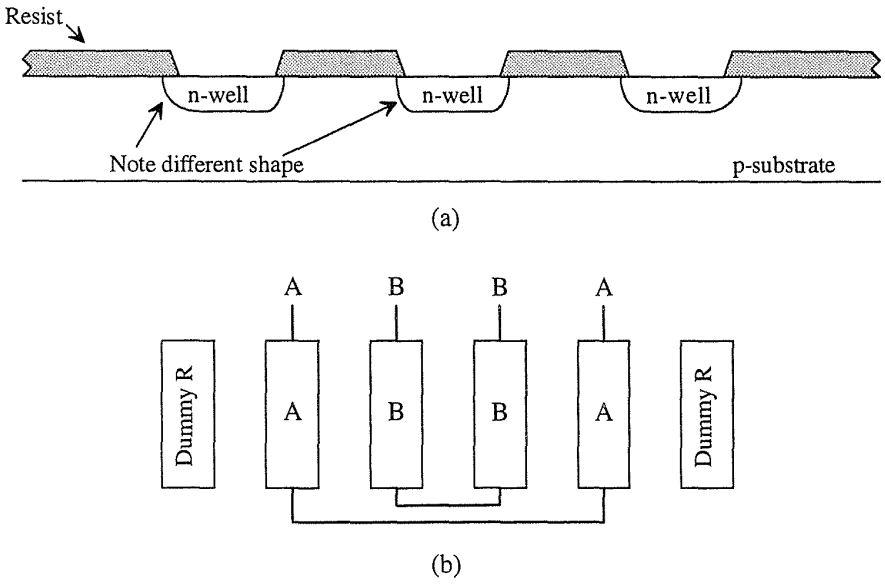



**Figure 7.9** (a) Edge effects and (b) a common-centroid layout with dummy R.

To compensate for this effect, dummy elements can be added (see Fig. 7.9b) to an interdigitated or common-centroid layout. The dummy element does nothing electrically. It is simply used to ensure that the unit resistors used in matched resistors see the same adjacent structures.

### 7.2.2 Capacitors

The first-order temperature coefficient of a capacitor, *TCC*, is given by

$$TCC = \frac{1}{C} \cdot \frac{dC}{dT} \tag{7.5}$$

A typical value of *TCC* for a poly1-poly2 capacitor is 20 ppm/°C. Matching of poly1-poly2 capacitors on a die is typically better than 0.1 percent. The capacitance as a function of temperature is given by

$$C(\mathrm{T}) = C_{T0} \cdot [1 + TCC \cdot (\mathrm{T} - T_0)] \tag{7.6}$$

where $C_{T0}$ is the capacitance at $T_0$. The voltage coefficient of a capacitor is given by

$$VCC = \frac{1}{C} \cdot \frac{dC}{dV} \tag{7.7}$$

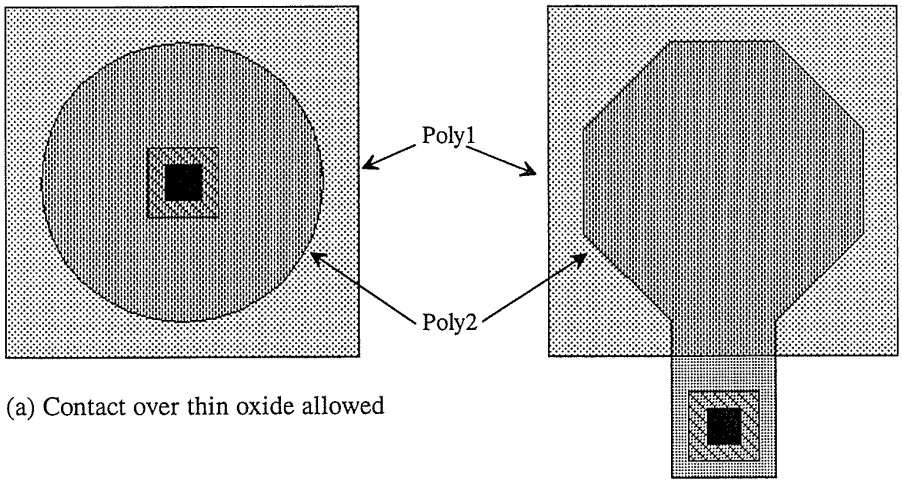
The voltage coefficient of the poly capacitor is in the neighborhood of 10 ppm/V. The capacitance as a function of voltage is given by

$$C(\mathrm{V}) = C_{V0} \cdot (1 + VCC \cdot \mathrm{V}) \tag{7.8}$$

where $C_{V0}$ is the capacitance between the two poly layers with zero applied voltage, and $V$ is the voltage between the two plates.

Microchip Ex. 1055, p. 165
Microchip v. Aptiv
IPR2024-00494

The layout of capacitors also uses the unit cell concept. Figure 7.10 shows two nominally 100 fF unit cells (a practical minimum for the CN20 process). The unit cell of Fig. 7.10a allows contacts to poly to be placed directly on top of the thin (or gate) oxide (the oxide between the two poly plates of the capacitor). The bottom plate of the capacitor is made using poly1, while the top plate, the plate whose area determines the capacitance, is made using poly2. A circular disk is used for poly2 (see Fig. 7.10a), so that the amount of poly under etch is constant around the perimeter of the disk. The unit cell of Fig. 7.10b is used when the contact to poly2 must lie over the field region. Sharp corners are avoided in this layout as well. In both unit cells of Fig. 7.10, the contact to poly1 (the bottom plate) is not shown.




(a) Contact over thin oxide allowed

(b) Contact over thin oxide not allowed

**Figure 7.10** Layout of a nominally 100 fF capacitor unit cell.

## 7.3 Noise in Resistors

Noise is a term used to describe an unwanted signal, voltage, or current present in a circuit. Noise in a resistor is primarily the result of random motion of electrons due to thermal effects. This type of noise is then termed *thermal noise*, or Johnson noise, after J. B. Johnson who performed experimental studies on thermal noise.

Thermal noise in a resistor can be characterized by adding a parallel current generator to the resistor. The root mean squared value of the current generator is given by

$$\sqrt{\overline{i^2}} = \sqrt{\frac{4kT}{R}} \cdot \sqrt{B} \tag{7.9}$$

Microchip Ex. 1055, p. 166
Microchip v. Aptiv
IPR2024-00494

where

$k$ = Boltzmann's constant = 1.38 x $10^{-23}$ Watt ·sec/°K

$T$ = temperature in °K

$R$ = resistance in Ω

$B$ = bandwidth in Hz

Often we do not include the bandwidth over which we calculate the noise until the end of the noise calculation. In this case the units of the RMS noise current are $A/\sqrt{Hz}$. An alternative representation is given in terms of the RMS voltage, or

$$\sqrt{\overline{v^2}} = \sqrt{4kTR} = \sqrt{\overline{i^2}} \cdot R \tag{7.10}$$

Addition of the RMS noise source is shown schematically in Fig. 7.11. Performing a noise analysis on a circuit begins with the addition of the RMS noise sources to the circuit. Then using superposition, we determine the RMS output noise due to each source. We then square each contribution to get the mean squared noise voltage from each noise source. Next, we sum the mean squared components to obtain the total mean squared noise. The total RMS output noise, $\sqrt{\overline{v_{on}^2}}$, is the square root of the integral of the total mean squared output noise, over the bandwidth of interest. The following examples should make the procedures for doing noise analysis clear.

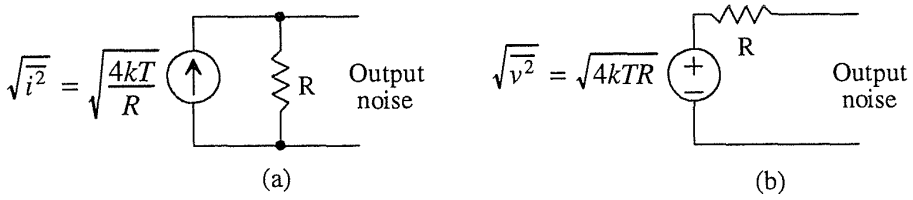



**Figure 7.11** Noise sources (a) current and (b) voltage.

**Example 7.5**

For the circuit shown in Fig. Ex7.5, determine the RMS output and input noise over a bandwidth from DC to 1 kHz. Verify your answer with SPICE.

The RMS noise currents are given by

$$\sqrt{\overline{i_{10k}^2}} = \sqrt{\frac{4 \cdot (1.38 \times 10^{-23}) \cdot (300)}{10,000}} = 1.29 \times 10^{-12} \frac{A}{\sqrt{Hz}}$$

and

$$\sqrt{\overline{i_{1k}^2}} = 4.1 \times 10^{-12} \frac{A}{\sqrt{Hz}}$$

Microchip Ex. 1055, p. 167
Microchip v. Aptiv
IPR2024-00494

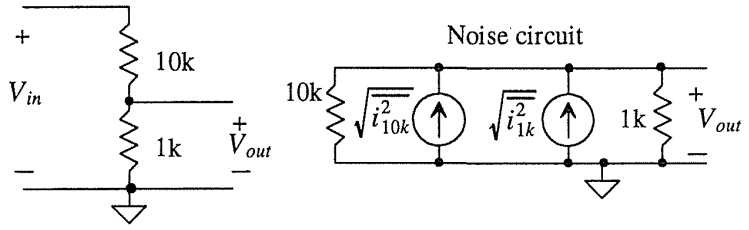



**Figure Ex7.5**

The root mean squared output noise voltage, $\sqrt{\overline{v_{10k}^2}}$ due to $\sqrt{\overline{i_{10k}^2}}$ is $1.29\times10^{-12}\frac{\text{A}}{\text{Hz}}\cdot\frac{1\text{k}\cdot10\text{k}}{1\text{k}+10\text{k}}$ or 1.2 nV/$\sqrt{\text{Hz}}$ and the mean squared output noise voltage, $\sqrt{\overline{v_{1k}^2}}$ from the 1k resistor is 3.7 nV/$\sqrt{\text{Hz}}$. The total mean squared output noise voltage over a bandwidth of 1 KHz is given by

$$\overline{v_{on}^2}=\int_{f_L}^{f_H}\overline{v_T^2}\cdot df=\int_0^{1\text{kHz}}\overline{(v_{10k}^2+v_{1k}^2)}\cdot df=15.1\times10^{-15}\ \text{V}^2$$

Notice that no distinction was made between the noise voltage with and without the bandwidth introduced. The RMS output noise of this circuit is then

$$\sqrt{\overline{v_{on}^2}}=123\ \text{nV}$$

This noise source voltage source can be added in series with the output of the noiseless circuit. The noise voltage gives the engineer an idea of the minimum voltage that can be measured in a circuit.

Another specification is the effective input noise at the two terminals that are connected to the input source. The effective input noise is a noise voltage in series with the input, which will generate the 123 nV of noise at the output. For this example,

$$\sqrt{\overline{v_{in}^2}}=123nV\cdot\frac{10\text{k}+1\text{k}}{1\text{k}}=1.35\ \mu\text{V}$$

is the RMS input noise. SPICE3 simulation gives an output mean squared noise of 15.1 × $10^{-15}$ $V^2$ (= 123 nV RMS) and an input noise referenced to 1 V of 1.8 × $10^{-12}$ (= 1.35 μV RMS). The SPICE3 netlist is shown below.

```
*** top level netlist ***
.noise   v(2,0)  vin dec  100 1 1k
r1       1 2 10k
r2       2 0 1k
vin      1 0     dc 0 ac 1
.print noise all
.probe
.end
```

Microchip Ex. 1055, p. 168
Microchip v. Aptiv
IPR2024-00494

```
---------------------------------------------------------------------------
Index   inoise_total      onoise_total
---------------------------------------------------------------------------
0       1.821510e-012 1.505380e-014
```

Note that a bandwidth of 1 to 1,000 Hz was used in this simulation rather than DC to 1 kHz. Also note that the reference supply was a voltage, so the units of the SPICE output are in $V^2$. The input AC source has a magnitude of 1 V, so the input noise SPICE gives is divided by 1V squared. If we had used a 1 mV AC supply, then the inoise_total above would be $1.82 \times 10^{-6}$ $V^2$.

The PSPICE netlist is as follows.

```
*** top level netlist ***
.noise v(2,0) vin
.ac dec 100 1 1k
r1 1 2 10k
r2 2 0 1k
vin 1 0 dc 0 ac 1
.probe
.end
```

The magnitude of the input source, in this case "vin" in the netlist, affects the output noise in SPICE3, while in PSPICE the magnitude has no effect on the output noise. The input and output noise voltage density, units $V/\sqrt{\text{Hz}}$, is specified in PSPICE by v(inoise) and v(onoise), respectively. The specification for input and output noise current density follows a similar form. Determining the total output noise in PROBE consists of determining the mean squared output voltage, specified by v(onoise)*v(onoise), summing (integrating) each mean squared output voltage over the simulation frequencies, s(v(onoise)*v(onoise)) and taking the square root of the result, sqrt(s(v(onoise)*v(onoise))). The results of the PSPICE simulation are shown in Fig. 7.12. ■

In summary the procedure for doing a noise analysis is as follows.

1. Add RMS noise sources to circuit.
2. Determine RMS circuit output noise from each component using superposition.
3. Square each component of the output noise to get mean squared component.
4. Take the sum of each contribution.

*For the output noise,*

5. Integrate the sum over the bandwidth.
6. The square root of this result is the RMS output noise.

*and or the input noise,*

Microchip Ex. 1055, p. 169
Microchip v. Aptiv
IPR2024-00494

5. Divide the sum, step 4 above, by the squared magnitude of the network transfer function, that is, $|H(j\omega)|^2 = \left|\frac{V_o(j\omega)}{V_i(j\omega)}\right|^2$, assuming we are interested in noise voltages.

6. Integrate the result over the bandwidth of interest.

7. The square root of this result is the RMS input noise.

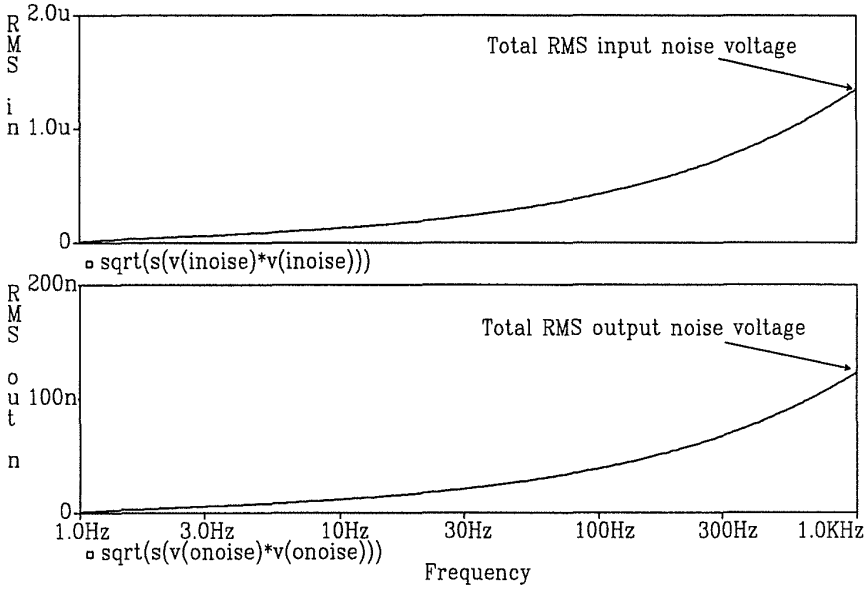



**Figure 7.12** PSPICE simulation results for Example 7.5.

**Example 7.6**

Determine the RMS input and output noise voltages for the following RC circuit over bandwidths from DC to 10 MHz and 100 MHz. Verify your hand calculations with SPICE.

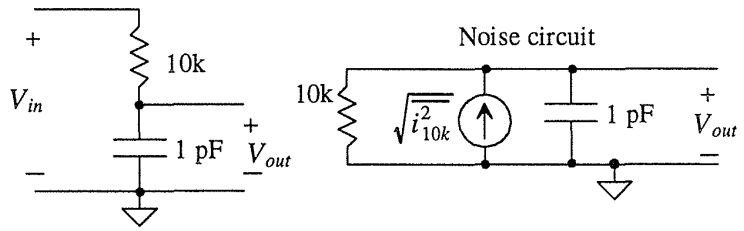



**Figure Ex7.6**

We begin with step 1, adding the noise sources to the circuit. We know from the previous example that the RMS thermal noise current from the 10 k resistor is

Microchip Ex. 1055, p. 170
Microchip v. Aptiv
IPR2024-00494

$$\sqrt{\overline{i_{10k}^2}} = 1.29 \times 10^{-12} \frac{A}{\sqrt{\text{Hz}}}$$

This generates an RMS noise voltage at the output given by

$$\sqrt{\overline{v_{10k}^2}} = \sqrt{\overline{i_{10k}^2}} \cdot \frac{R}{\sqrt{1 + (2\pi RCf)^2}}$$

which is the end of step 2 given above. Combining steps 3 and 4 gives

$$\overline{v_T^2} = \overline{v_{10k}^2} = \overline{i_{10k}^2} \cdot \frac{R^2}{1 + (2\pi RCf)^2}$$

To determine the output noise using step 5, we get,

$$\overline{v_{on}^2} = \int_{f_L}^{f_H} \overline{i_{10k}^2} \cdot \frac{R^2}{1 + (2\pi RCf)^2} \cdot df$$

This gives the mean squared output noise in units of $V^2$, which is what SPICE gives you as well. At this point, a useful relationship is

$$\int \frac{du}{a^2 + u^2} = \frac{1}{a} \tan^{-1} \frac{u}{a} + C \tag{7.11}$$

Applying this to $\overline{v_{on}^2}$ we get

$$\overline{v_{on}^2} = \frac{\overline{i_{10k}^2} \cdot R^2}{2\pi RC} \cdot [\tan^{-1}(2\pi RC \cdot f_H) - \tan^{-1}(2\pi RC \cdot f_L)]$$

or putting in the numbers for this example with $f_L = 0$ gives

$$\overline{v_{on}^2} = \frac{1.66 \times 10^{-24} \cdot 10k}{2\pi \times 10^{-12}} \cdot \tan^{-1} 2\pi 10k \cdot 10^{-12} \cdot f_H$$

For a bandwidth of 10 MHz, the mean squared output noise voltage is $\overline{v_{on}^2}$ = 1.48 $nV^2$, while for the bandwidth of 100 MHz $\overline{v_{on}^2}$ = 3.73 $nV^2$. Both of these results were verified with SPICE. The RMS output noise voltages for the 10 MHz and 100 MHz bandwidths are then 38.5 μV and 61.07 μV, respectively.

To determine the input noise, we must first determine the transfer function, $H(j\omega)$ from step 5 of the input noise calculation, given by

$$H(j\omega) = \frac{1}{1 + j\omega RC} = \frac{V_o(j\omega)}{V_i(j\omega)}$$

Thus

$$\frac{\overline{v_T^2}}{|H(j\omega)|^2} = R^2 \cdot \overline{i_{10k}^2}$$

or a constant independent of frequency. Combining steps 6 and 7 for the input noise calculation gives

Microchip Ex. 1055, p. 171
Microchip v. Aptiv
IPR2024-00494

$$\sqrt{\overline{v_{in}^2}} = \sqrt{\int_{f_L}^{f_H} \frac{\overline{v_T^2}}{|H(j\omega)|^2} \cdot df} = \sqrt{\int_0^{f_H} R^2 \overline{i_{10k}^2} \cdot df} = R \cdot \sqrt{i_{10k}^2} \cdot \sqrt{f_H}$$

The RMS input noise voltage over the bandwidths from DC to 10 MHz and 100 MHz are 40.8 μV and 129 μV, respectively. It is interesting to observe that at large $f_H$, compared to $\frac{1}{2\pi RC}$, the output noise does not increase, while the input noise continues at a rate determined by the resistor. This is because the capacitor shorts the high-frequency components of the output noise to ground. The following example illustrates the determination of the maximum output noise possible in an *RC* circuit with infinite bandwidth. ■

**Example 7.7**

Estimate the output noise of the following *RC* low-pass circuit over the frequency range of DC to infinity.

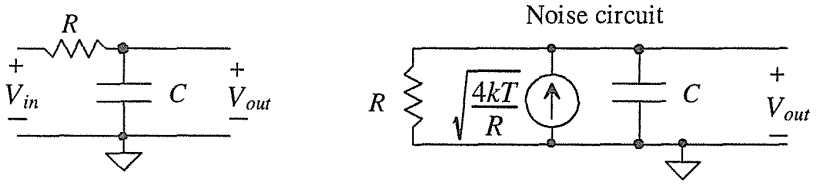



**Figure Ex7.7**

The total mean squared output noise is given by

$$\overline{v_T^2} = \frac{4kT}{R} \cdot \frac{R^2}{1+(\omega RC)^2}$$

and the RMS output noise is given by

$$\sqrt{\overline{v_{on}^2}} = \sqrt{\int_0^{\infty} \overline{v_T^2} \cdot df} = \sqrt{\lim_{f_H \to \infty} \frac{4kT}{R} \cdot \frac{R^2}{2\pi RC} \cdot \tan^{-1}(2\pi RC \cdot f_H)}$$

$$= \sqrt{\frac{kT}{C}} \text{ V} \qquad (7.12)$$

This maximum limitation on the output noise is, for obvious reasons, referred to as *kT/C* noise (pronounced "Kay Tee over Cee noise"). The maximum RMS output noise of the *RC* circuit in Ex. 7.6 is given by

$$\sqrt{\overline{v_{on,\max}^2}} = \sqrt{\frac{kT}{C}} = \sqrt{\frac{(1.38 \times 10^{-23}) \cdot 300}{10^{-12}}} = 64 \text{ μV}$$

Notice that this maximum RMS output noise is close to what was obtained over the 100 MHz bandwidth, a bandwidth much greater than the 3 dB bandwidth of the circuit. ■

Microchip Ex. 1055, p. 172
Microchip v. Aptiv
IPR2024-00494

## REFERENCES

[1] D. J. Allstot and W. C. Black, "Technology Design Considerations for Monolithic MOS Switched-Capacitor Filtering Systems," *Proceedings of the IEEE*, Vol. 71, No. 8, August 1983, pp. 967-986.

[2] R. A. Pease, J. D. Bruce, H. W. Li, and R. J. Baker, "Comments on Analog Layout Using ALAS!" *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 9, September 1996, pp. 1364-1365.

## PROBLEMS

**7.1** Lay out and DRC an *RC* circuit with a nominal time constant of 100 ns. Due to process variations, what are the maximum and minimum RC values possible?

**7.2** For an n-well resistor fabricated in the CN20 process, plot resistance versus temperature for a 20 kΩ resistor at room temperature (27 °C) over –50 to 100 °C.

**7.3** If a 1 pF capacitor at room temperature (27 °C) has a temperature coefficient of 100 ppm/°C, estimate the minimum and maximum capacitances from –40 to 100 °C. Plot the capacitor value versus temperature.

**7.4** Lay out and DRC two 50 kΩ matched resistors. Include guard rings in your layout. Using the padframe cell, from Ch. 4, show the connection of the two resistors to four pads and the connection of *VSS* to a *VSS* pad.

**7.5** Determine the RMS input and output noise voltages over a bandwidth from DC to 100 kHz for the following circuit.

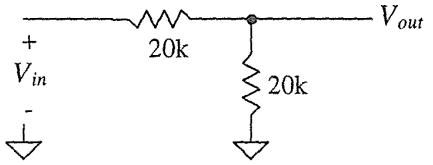



**Figure P7.5**

**7.6** Repeat Problem 5 for the following circuit.

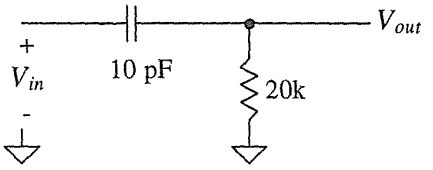



**Figure P7.6**

Microchip Ex. 1055, p. 173
Microchip v. Aptiv
IPR2024-00494

Chapter

8

# Design Verification with LASICKT

An important step in the design process is verifying that the layout of an integrated circuit matches the schematics of the integrated circuit. From either the layout or the schematic we should be able to generate a SPICE circuit file and simulate the operation of the IC. A comparison between the netlists generated from the layout and schematic as well as the resulting SPICE outputs can be used to verify the schematic and layout match. This chapter discusses design verification using the LASICKT program.

An example chip design, useful in illustrating schematic generation and documentation of layout, is given in the directory C:\LASI5\2UCHIP. This chip is by no means an elegant layout. It was designed using the MOSIS Scalable CMOS Design Rules (Appendix B) with a $\lambda$ of 1 $\mu$m to give examples of how to generate layouts and schematics for use with LASICKT.

The MOSIS design rules use layers that are slightly different from the layers used with the CN20 setups of Appendix A. For example, to lay out a box of n+ in the CN20 process, we simply select the n+ layer and draw a box, Fig. 8.1a. However, to lay

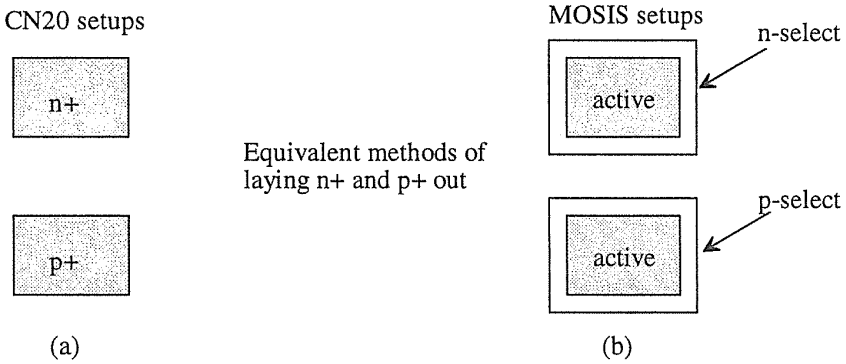



**Figure 8.1** Comparison between the layouts of active area using the CN20 and MOSIS setups.

Microchip Ex. 1055, p. 174
Microchip v. Aptiv
IPR2024-00494

out a box of n+ using the MOSIS design rules, we begin by laying out a box on the "active" (also sometimes called composite) layer. To make this active area n+ we draw a box around this area on the n-select (nsel) layer (Fig. 8.1b). The n-select box must be at least 2 λ bigger than the active box; that is, any edge of active must be at least 2 λ away from any edge of n-select. Similarly, for p+, we draw a box on the active layer, using the MOSIS setups, and surround this box with p-select.

## 8.1 Fundamentals of LASICKT

The basic drawing of an n-channel MOSFET with connector nodes and labeling is shown in Fig. 8.2. This cell's name is $NMOS1 ($ at the beginning of a cell name indicates a schematic) and has a rank of 1. (Again the following drawings are available in the C:\LASI5\2UCHIP directory.) The actual symbol of the MOSFET was drawn on the schematic layer (layer 3) using a zero-width path or polygon.

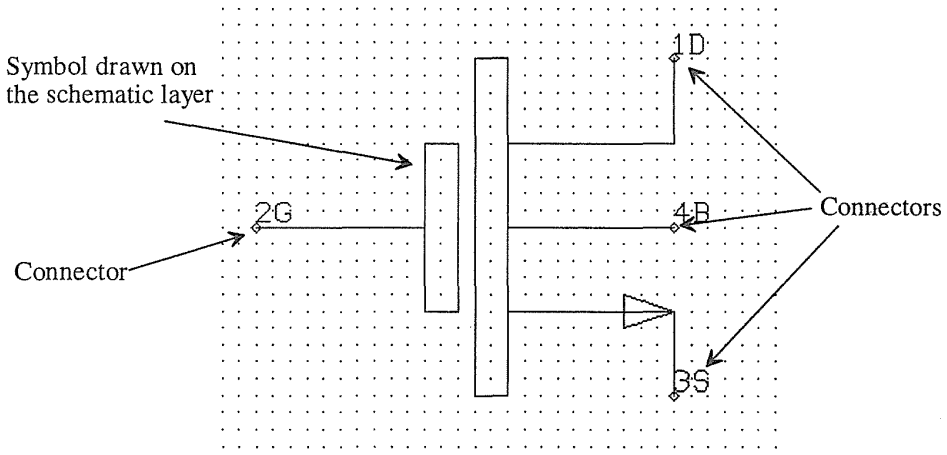



**Figure 8.2** Schematic symbol of an n-channel MOSFET.

*Specifying Connections to the Symbol*

After drawing the symbol, the next step is to label the connections or points on the symbol that will be connected to the wires in the schematic drawing. Text on the connector layer (layer 5) is used to label the connections to the symbol. The vertex of the text specifies the exact location of the connector. Remember that displaying the vertex can be enabled or disabled by pressing "**t**" on the keyboard followed by executing a **Draw** command.

*Layout of the MOSFET*

Figure 8.3 shows the layout of two MOSFETs with cell names N3x2 and N15x5 (both with a rank of 1). Again, connector text (on layer 5) is used to specify the connections to the drain, gate, source, and substrate. Notice that the order of the connectors is important; that is, it must correspond to the order of nodes used in SPICE.

Microchip Ex. 1055, p. 175
Microchip v. Aptiv
IPR2024-00494

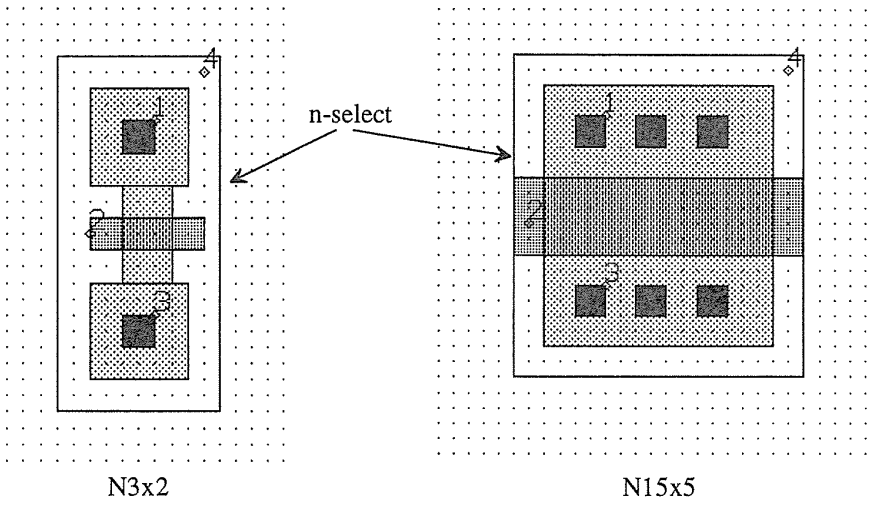



**Figure 8.3** Layout with node connections of an n-channel MOSFET.

### 8.1.1 The Inverter

To help in understanding the process used to extract a circuit file from a schematic or layout, let's consider the design and layout of a basic inverter. Figure 8.4 shows the placement of $NMOS1 and $PMOS1 cells into cell $INVERT (rank 2). Notice that an outline of the cell, enabled by pressing "**i**" on the keyboard, is used to show the actual boundaries of the cells. The next step in drawing the schematic of the inverter is to label the device and parameter information. We will label the NMOS M1 and the PMOS M2 using the device text layer (layer 6), as shown in Fig. 8.5. The layouts in Fig. 8.3 correspond to NMOS devices with width to length ratios of 3/2 and 15/5. Since we have a single symbol for an NMOS transistor, the size of the MOSFET and the

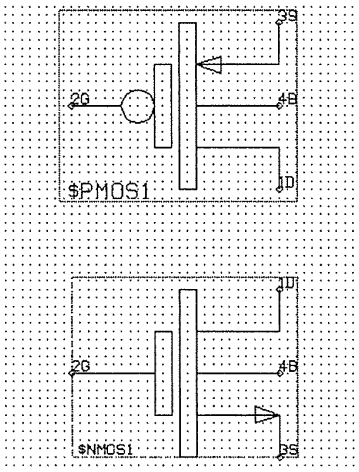



**Figure 8.4** Starting the schematic of an inverter.

Microchip Ex. 1055, p. 176
Microchip v. Aptiv
IPR2024-00494

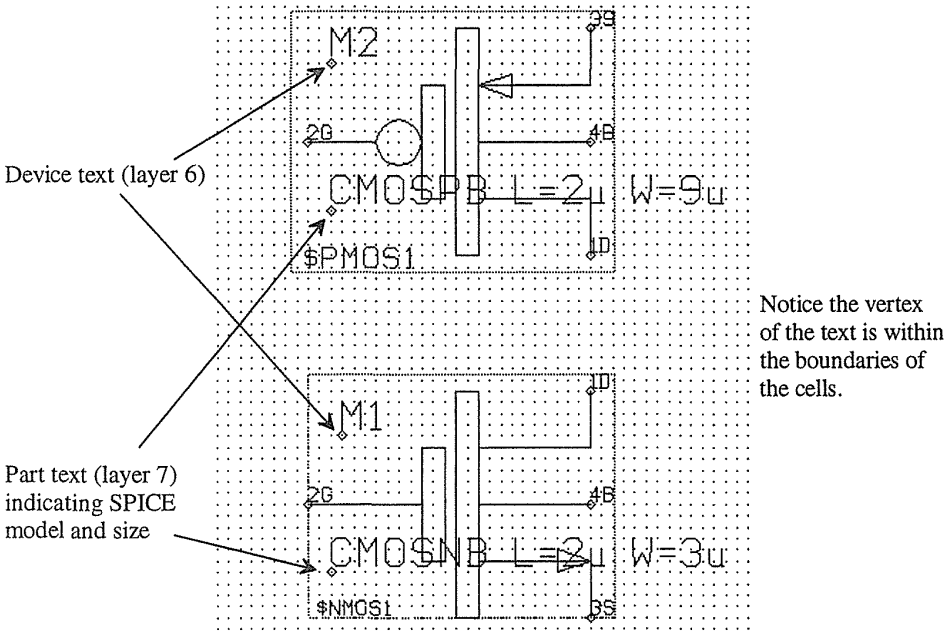



**Figure 8.5** Adding device and parameter text to a schematic.

model used are specified on the parameter text layers (layer 7). Notice that the vertex of both the parameter text and device text must be placed within the boundaries of the cell.

*Labeling the Layout*

The layout corresponding to the schematic of Fig. 8.5 is shown in Fig. 8.6. A cell named INVERT with a rank of 2 was created. The cells P9X2, SFRAME, and N3X2 were placed into the INVERT cell. Device and parameter text were used to specify the device name (M1 and M2) and the parameter information (SPICE model, widths, and lengths). It is important that the device name on the schematic correspond directly to a particular device in the layout.

*Making Connections*

Wires, for use in schematics only, are drawn on the schematic layer (layer 3). Zero-width paths are used to make contact to the device connectors. A connection is made when a wire (zero-width path drawn on layer 3) crosses the vertex of text written on the connector text layer (layer 5). Figure 8.7a shows a wire making contact with the source and substrate of an n-channel MOSFET. Note that the vertex of a wire doesn't have to coincide with the vertex of the connector text to make a connection. One of the vertices of the wire coincides with the substrate connection, while the other vertex is outside of the cell boundary. Figure 8.7b shows the connection between wires. In this situation, the vertex of the wires must coincide for a connection to be made. This enables wires to cross without making a connection. Note that the **fGet** command

Microchip Ex. 1055, p. 177
Microchip v. Aptiv
IPR2024-00494

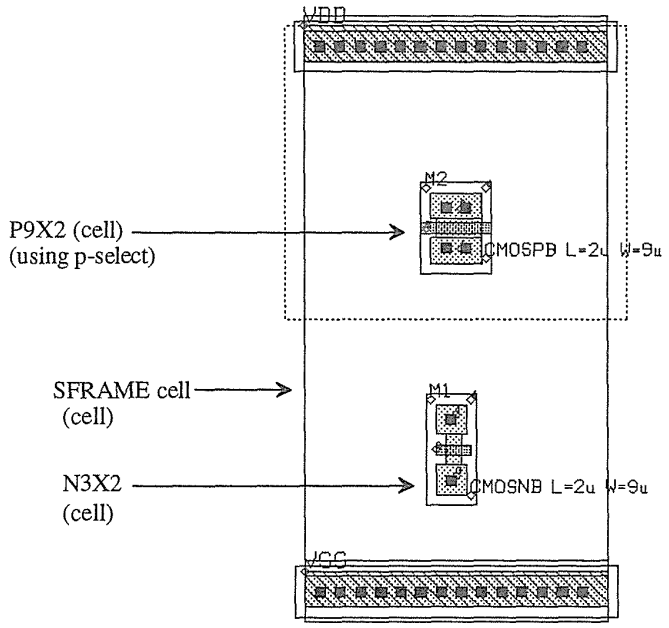



**Figure 8.6** Placing the MOSFET cells in the standard cell frame and labeling.

(around a vertex) was used to show the vertices of the wires in this figure. Figure 8.8 shows the schematic of the inverter with wire connections in place.

The layout of the inverter with interconnections is shown in Fig. 8.9. Poly1, metal1, or metal2 can be used to connect the terminals of the two MOSFETs to *VDD*, ground, the input, and the output. The layer numbers used to specify the interconnect layers, using the MOSIS setups, are 46 (poly1), 49(metal1), and 51(metal2). We might now ask the question, "How does LASICKT know which connections are the inputs,

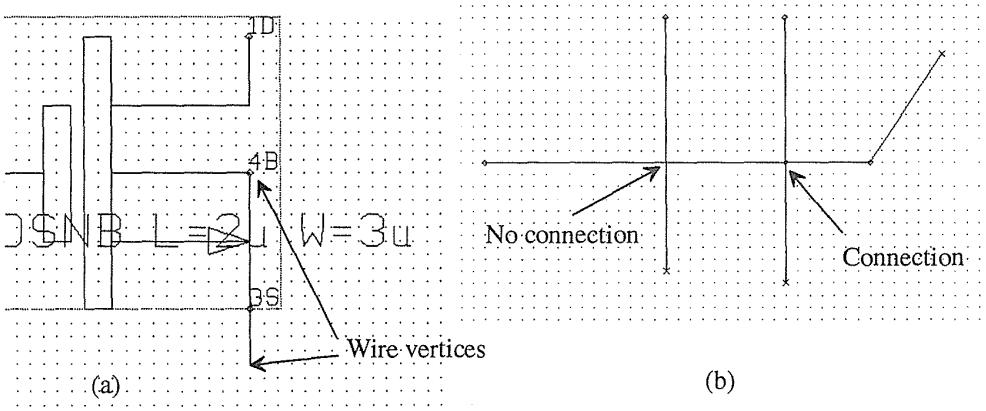



**Figure 8.7** How wires make connections.

Microchip Ex. 1055, p. 178
Microchip v. Aptiv
IPR2024-00494

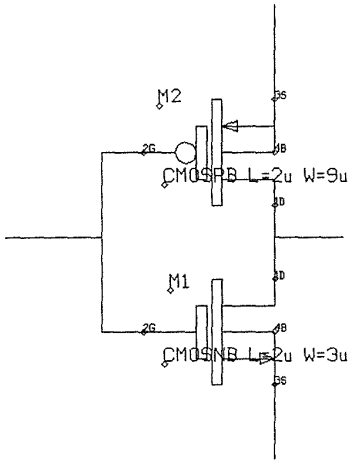



**Figure 8.8** Inverter with wire connections.

outputs, and power supplies?" The answer is that we label nodes in the layout and schematic that tell LASICKT which nodes are the inputs and which are the outputs.

*Labeling the Nodes*

The next step in creating a schematic or layout is to label the nodes that correspond to inputs/outputs and power supply connections. Figure 8.10 shows that the power supply node is labeled, on the node text layer (layer 4), *VDD*, the input is labeled A, the output

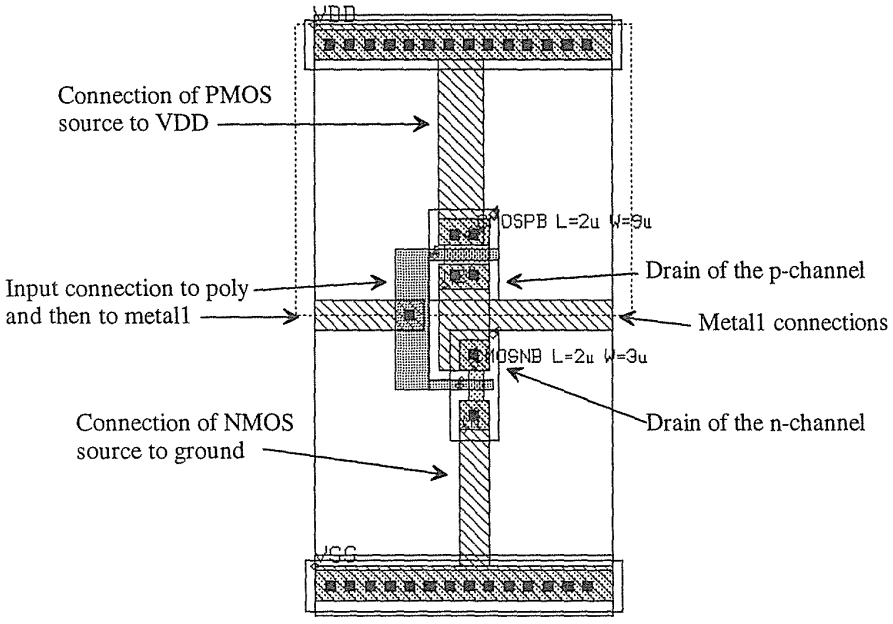



**Figure 8.9** Adding connections (poly and metal1) to the layout of the inverter.

Microchip Ex. 1055, p. 179
Microchip v. Aptiv
IPR2024-00494

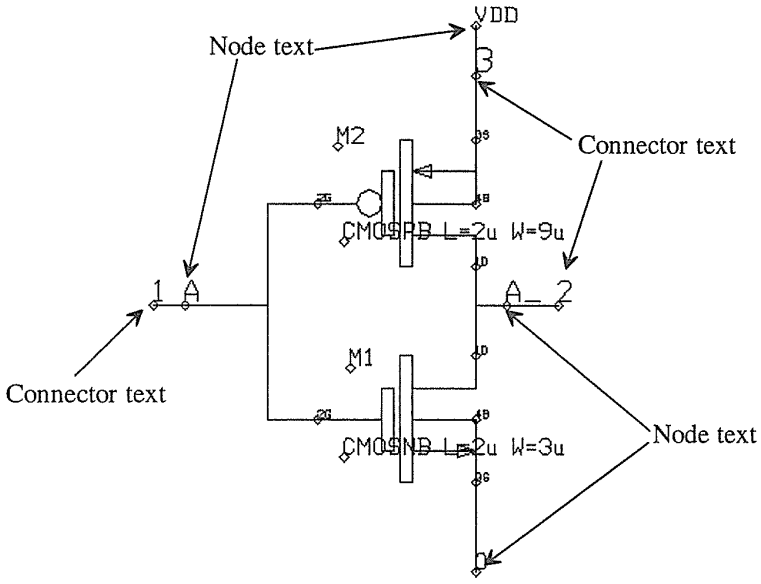



**Figure 8.10** Adding node and connector text to the schematic of the inverter.

is labeled A_ and the ground connection is labeled 0. (Ground is always node 0 in a SPICE circuit file.) It is important to label the input/output nodes, not letting LASICKT select a name, so that we know how to connect the input voltage or current sources when writing the header file.

In this schematic, we have also labeled the input, output, and *VDD* nodes with connectors (on the connector text layer). Ground was not labeled with a connector since node 0 is universally ground potential. Adding connectors to the inverter allows us to use the inverter as a subcircuit in a higher ranking cell.

The layout of the inverter with labeled nodes and connectors is shown in Fig. 8.11. Notice the direct correlation between the layout and schematic. Both the schematic and layout of the inverter are used in higher ranking cells in the example chip given in the directory C:\LASI5\2UCHIP.

From this layout we can make the following observations.

1. The vertex of the node text must be located on a box or path used for interconnection within a cell. It will have no effect if located on a lower ranking cell. For example, the vertex of the *VDD* node text in Fig. 8.11 must be on the metal1 box connecting the source of the p-channel with the SFRAME (standard frame) cell. Placing it on the metal1 of the underlying SFRAME cell will not label the node.

Microchip Ex. 1055, p. 180
Microchip v. Aptiv
IPR2024-00494

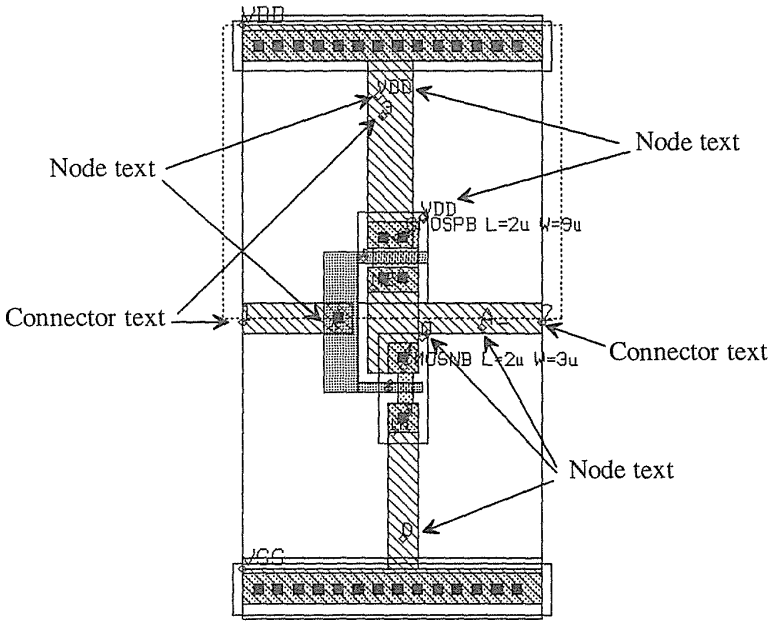



**Figure 8.11** Adding node text and connector text to the layout of an inverter.

2. Body connections, substrate or well, must be labeled with a node name separately from the other nodes in the circuit.

3. The vertex of node A in Fig. 8.11 is placed at the intersection of poly1 and metal1. Poly1, metal1, and metal2 are the interconnect layers in a layout. These layers can cross without making contact. Were the layers are connected, using a contact or via, the node text is used to indicate the connection.

4. The vertex of connector text may be placed anywhere in a cell where connection is to be made to the next higher cell. This normally will be where interconnect layers join to the next higher ranked cell.

5. Since connectors are used to specify how cells are connected upward in the cell hierarchy, connectors on a top cell are unnecessary. In fact, they will be flagged as "open connector" errors when LASICKT compiles. If the cell INVERT was the top cell, there would be no connector text placed on it.

6. There is no node or connector text on the SFRAME cell because it is not ar active device. Unless it has an actual SPICE model, it will simply be ignored by LASICKT.

Adding the additional text to label parameters, device type, nodes, and connectors can clutter a layout or schematic to the point that it becomes unreadable Using the **View** command button in LASI can limit the number of layers displayed in drawing.

Microchip Ex. 1055, p. 181
Microchip v. Aptiv
IPR2024-00494

### 8.1.2 Running LASICKT

To launch the LASICKT program, return to the system menu and click on the LASICKT command button. The screen shown in Fig. 8.12 will appear. Fill in the setup information described below and then press the Go button. If the schematic and layout are drawn using the methods just discussed, LASICKT will compile SPICE circuit and netlist files. The **Comp** command button can be used to compare the schematic and layout netlists (.NET files generated with LASICKT) to test agreement between schematic and layout. The SPICE outputs resulting from the simulation of each circuit file can be compared as well.

*LASICKT Inputs*

*Name of Cell* - LASICKT will compile a circuit file based on the contents of the cell specified in this field. The default is the cell that was open the last time LASI was in cell mode.

*Header File* - Name of file that LASICKT will insert and the beginning of the circuit file. This file generally contains information such as power supply voltages and input sources.

*Footer File* - Name of file that LASICKT will append to the end of the circuit file. Generally, this file contains the SPICE models and can be common to all cells in the drawing directory.

*Interconnect Layers* - Layers used to connect the cells together. For the MOSIS setups used in C:\LASI5\2UCHIP layer 3 (schematic), layer 46 (poly1), layer 49

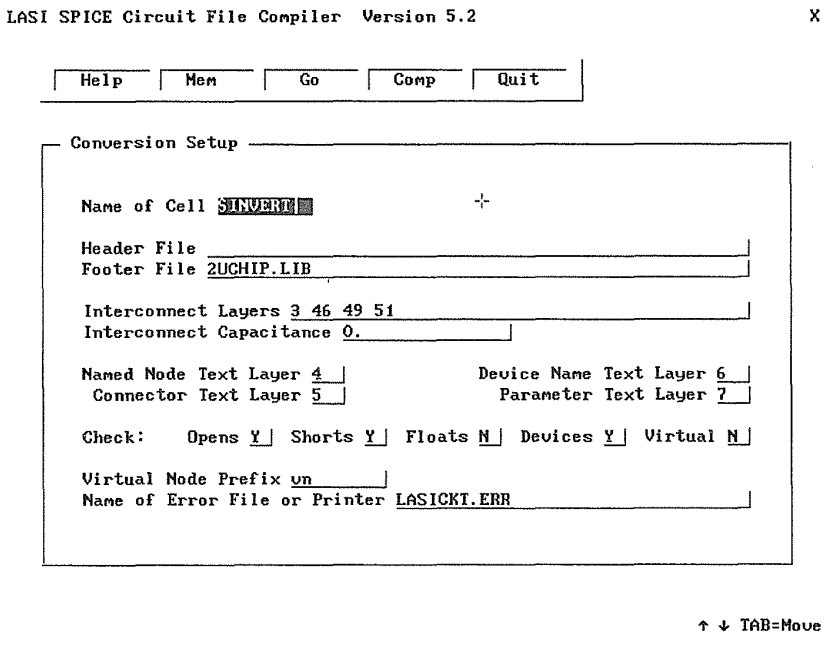



**Figure 8.12** LASICKT system screen.

Microchip Ex. 1055, p. 182
Microchip v. Aptiv
IPR2024-00494

(metal1), and layer 51 (metal2) are used. Other layers such as n+ or p+ can be used for connections with the appropriate labeling in the layout.

*Interconnect Capacitance* - A value that can be used to estimate the capacitance of interconnects based on the area of the interconnection.

Additional information covering the operation or meaning of the inputs used in LASICKT can be found in the on-line help available by pressing F1 while at the LASICKT command menu or by pressing the **Help** command button.

*The Header File*

The following header files are provided for simulating the operation of the circuits provided in C:\LASI5\2UCHIP.

| | |
|---|---|
| INVERT | Simple inverter. |
| COMP1 | Comparator with amp and a single buffer output. |
| COMP2 | Comparator with latch and double output buffers. |
| DFF | D-type flipflop. |
| NAND | Simple 2-input NAND gate. |
| OR | Simple 2-input OR gate (NOR+INVERT). |
| SRFF | SR type flipflop. |
| TRANGATE | Transmission gate. |
| WSOTA | Wide-swing operational transconductance amplifier. |

The contents of INVERT.HDR are shown below:

```
V1  VDD 0  DC 5V AC 0 0
V2  A 0  DC 0 AC 0 0 PULSE (0 5V 10n 1ns 1ns 50ns 100ns)
.options reltol=0.1 abstol=1u vntol=50mv
.probe
.tran 1ns 150ns
```

The first line in this file specifies the DC voltage source used by the circuit. Note that *VDD* is a node name that corresponds to the labeled nodes in the inverter layout or schematic. The second line is the input pulse to the inverter, discussed in more detail below. The .options statement is used to help with convergence.

*.tran 1ns 150ns*

This statement is used to specify a transient analysis (the x-axis in probe is time) from 0 to 150ns with a maximum print step of 1ns. Increasing the print step can cause the output to become jagged. Small-print step size can result in a very large output data file (the file used by probe). The full specification for a transient analysis is given by

.tran (print-step) (stop-time) (delay-time) (maximum step size) (UIC)

Microchip Ex. 1055, p. 183
Microchip v. Aptiv
IPR2024-00494

The delay-time is the time when the data starts being saved to a file. If we wanted to simulate the operation of a circuit from 0 to 100ns but we were only interested in the results from 50ns to 100ns we could use: .tran 1ns 100ns 50ns and only the data from 50 to 100 ns would be saved in the probe data file. The maximum step size will limit the next incremental increase in time as SPICE is simulating the operation of a circuit (and can help with smoothing of SPICE output). Specifying UIC at the end of a .tran statement causes SPICE to use the initial conditions specified in the circuit, for example, the initial voltage across a capacitor.

*Pulse Statements*

General form:
PULSE(VI V2 TD TR TF PW PER)

Examples:
VIN 3 0 PULSE(-1 1 2n 2n 2n 50n l00n)

| PARAMETER | DEFAULT | UNITS |
|---|---|---|
| V1 (initial value) | | Volts or amps |
| V2 (pulsed value) | | Volts or amps |
| TD (delay-time) | 0.0 | Seconds |
| TR (rise-time) | TSTEP | Seconds |
| TF (fall-time) | TSTEP | Seconds |
| PW (pulse width) | TSTOP | Seconds |
| PER (period) | TSTOP | Seconds |

*Piecewise Linear Source*

General Form:
PWL (T1 V1 <T2 V2 T3 V3 T4 V4 ...>)

Examples:
VCLOCK 7 5 PWL (0 -7 10NS -7 11NS -3 17NS -3 18NS -7 50NS -7)

*Extracting a Circuit File from the Inverter Schematic*

Pressing Go on the LASICKT system screen shown in Fig. 8.12 will generate a circuit file, shown below, in the file $INVERT.CIR. A circuit file generated from the cell INVERT and placed into the text file INVERT.CIR results in an identical circuit file to $INVERT.CIR.

```
*** SPICE Circuit File of $INVERT***
* START OF INVERT.HDR
V1  VDD 0  DC 5V AC 0 0
V2  A 0  DC 0 AC 0 0 PULSE (0 5V 10n 1ns 1ns 50ns 100ns)
.options reltol=0.1 abstol=1u vntol=50mv
.probe
.tran 1ns 150ns
.plot tran all
.print tran all
* END OF INVERT.HDR
```

Microchip Ex. 1055, p. 184
Microchip v. Aptiv
IPR2024-00494

```
* MAIN CIRCUIT
M1 A_ A 0 0 CMOSNB L=2u W=3u
M2 A_ A VDD VDD CMOSPB L=2u W=9u

SPICE models not shown

.end
```

*SPICE Simulation Results*

The simulation results shown in Fig. 8.13 were generated using the circuit file given above and SPICE3. Since the SPICE circuit file generated using the schematic of the inverter and the layout are the same, the SPICE outputs are the same.

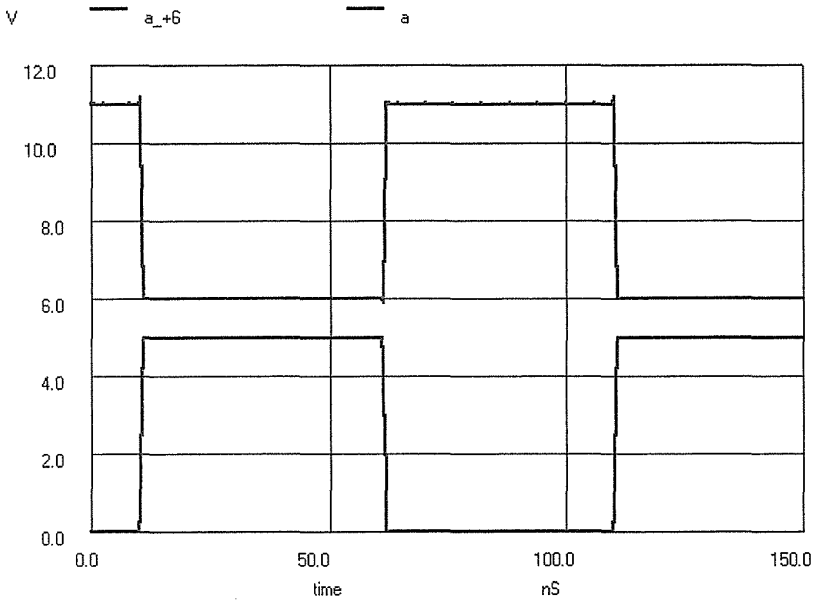



**Figure 8.13** Simulation output for the inverter of Figs. 8.10 or 8.11.

### 8.1.3 Higher-ranking Cells; The OR Gate

The schematic and layout of the inverter can be used in higher ranking cells. To illustrate this, consider the implementaion of an OR gate made using a NOR gate and an inverter. The schematic and layout of the NOR gate with rank 2 (named $NOR and NOR in C:\LASI5\2UCHIP) is shown in Fig. 8.14. These cells were created following the same procedures used to create the inverter cells discussed in the last two sections. SPICE circuit files can be compiled from these cells, keeping in mind that the header file must be changed to NOR.HDR, and the operation of the cells can be simulated.

Creating an OR gate schematic begins with creating a cell, in LASI, named OR with a rank of three. The $INVERT and $NOR cells are placed into this schematic

Microchip Ex. 1055, p. 185
Microchip v. Aptiv
IPR2024-00494

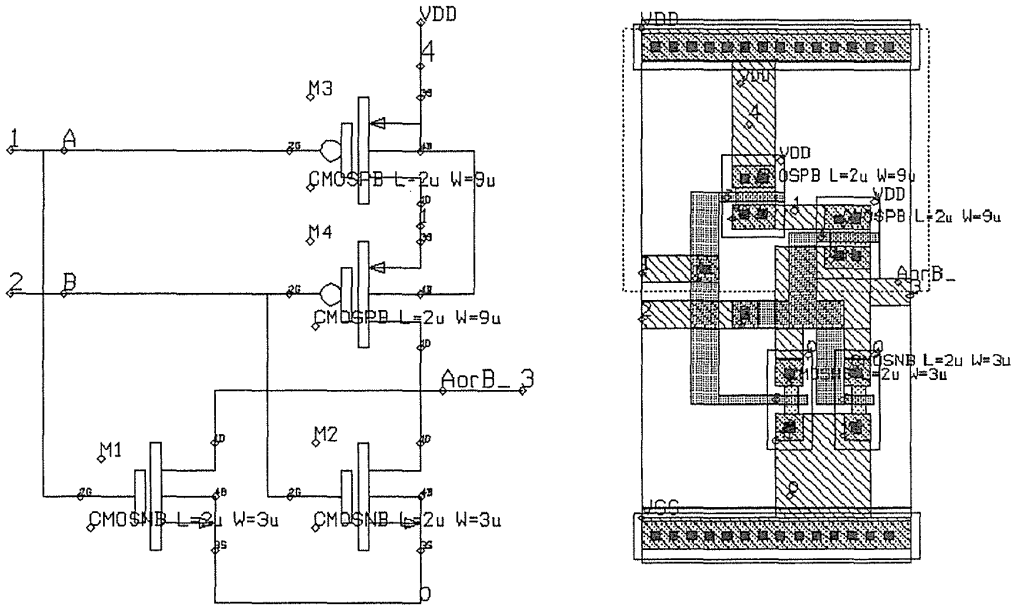



**Figure 8.14** Schematic and layout of a NOR gate.

(Fig. 8.15). Also shown in this figure are the added node and connector names. Figure 8.16 shows this OR gate cell with the NOR and NAND cells drawn as outlines. This figure reveals the location of the text and wires added to the cell. Let's discuss the

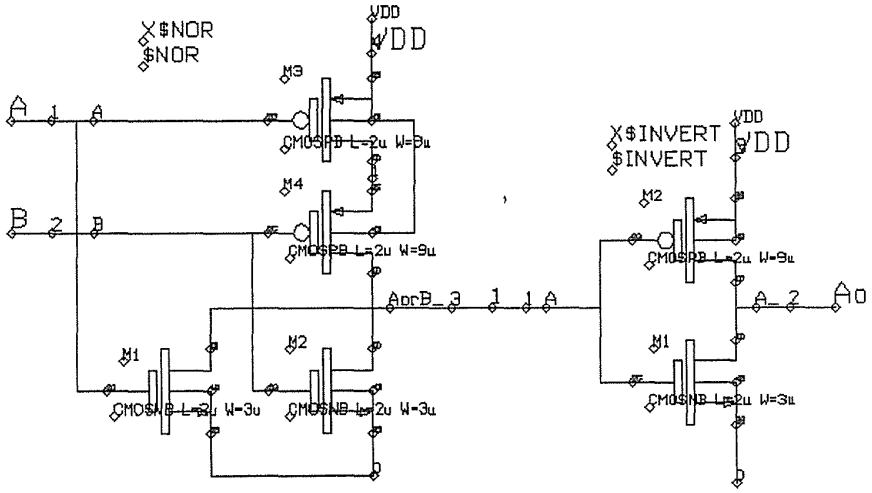



**Figure 8.15** Schematic of the OR gate.

Microchip Ex. 1055, p. 186
Microchip v. Aptiv
IPR2024-00494

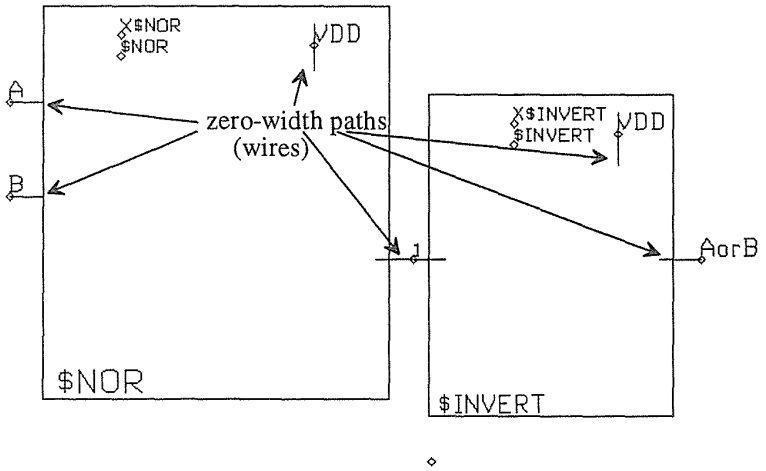



**Figure 8.16** The added zero-width paths and text to the OR gate cell.

labeling shown in this figure, keeping in mind that the first step in generating the OR gate schematic and layout was the addition of the lower ranking cells.

*Labeling Devices and Parameters*

The "devices" in this cell are labeled X$NOR and X$INVERT. Since the NOR and the INVERT cells are implemented in SPICE as subcircuits, the first letter in their name must be an X. This is similar to the MOSFET specification that the first letter, when labeling a MOSFET, has to be an M. The vertex of the parameter text is placed within the outline of the cell. The param text is a label telling LASICKT the name of the cell, for this case $NOR and $INVERT.

*Labeling Nodes*

The most important thing to remember, before labeling any of the nodes in a circuit, is that the vertex of the node text must lie on a box, polygon, or path. Node text placed on an underlying cell will have no effect unless it is placed exactly on a connector.

To connect *VDD* to the $NOR and $INVERT cells, we begin by placing a zero-width path on the schematic layer (layer 3) over the vertex of the power connector text. For the $NOR cell schematic, the power supply connection is node 4 on the connector text layer. For the $INVERT cell, the power supply connection is node 3 on the connector text layer. The next step is to place node name text, layer 4, on the added zero-width paths on the schematic layer. As shown in Fig. 8.15, it appears that the source of M3 in the $NOR gate cell is labeled twice as *VDD*. This is true. For each level of the schematic in the cell hierarchy, the *VDD* nodes must be labeled.

Microchip Ex. 1055, p. 187
Microchip v. Aptiv
IPR2024-00494

The next step in the generation of the schematic is to connect the cells together with, and label nodes on, zero-width paths. In this schematic, we have labeled the node connecting the $NOR cell to the $INVERT cell "1." We don't have to label this node. LASICKT will assign a virtual node number if nodetext is not placed on the zero-width wire. Note that the inputs to $OR are labeled A and B, while the output is labeled AorB. Also note that we did not label inputs, output, and power supply with connector text. If we want to use the $OR gate schematic in a higher ranking cell, the connector text must be added.

*Labeling the Layout*

The layout of the OR gate, with the cells drawn in outline mode, is shown in Fig. 8.17. Notice the added metal1 boxes in the layout. These boxes are placed over the connector text in the lower ranking cells. Note that node 1, in this figure, was not placed on a box; it was placed directly on the connector vertex of both lower ranking cells.

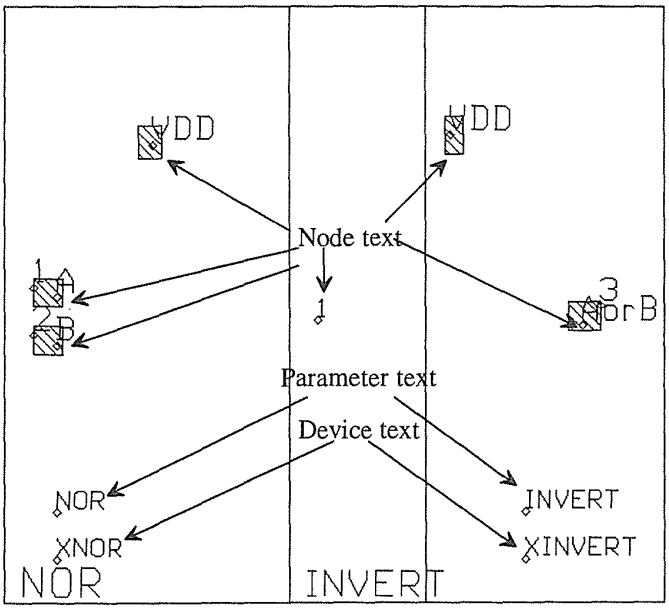



**Figure 8.17** Showing the labeling of the OR cell layout.

*Simulating the Operation of the OR Gate*

LASICKT was used to generate the SPICE circuit file, shown below, using OR.HDR for the header file. The simulation results, using this circuit file, are shown in Fig. 8.18.

```
* START OF OR.HDR
V1  VDD 0  DC 5V AC 0 0
V2  A 0  DC 0 AC 0 0 PULSE (0 5V 5ns 1ns 1ns 50ns 100ns)
V3  B 0  DC 0 AC 0 0 PULSE (0 5V 10ns 1ns 1ns 100ns 200ns)
.options reltol=0.1 abstol=10u vntol=10mv
```

Microchip Ex. 1055, p. 188
Microchip v. Aptiv
IPR2024-00494

```
.probe
.tran 1ns 200ns
.plot tran all
.print tran all
* END OF OR.HDR

.SUBCKT INVERT A A_ VDD
M1 A_ A 0 0 CMOSNB L=2u W=3u
M2 A_ A VDD VDD CMOSPB L=2u W=9u
.ENDS

.SUBCKT NOR A B AorB_ VDD
M3 1 A VDD VDD CMOSPB L=2u W=9u
M4 AorB_ B 1 VDD CMOSPB L=2u W=9u
M1 AorB_ A 0 0 CMOSNB L=2u W=3u
M2 AorB_ B 0 0 CMOSNB L=2u W=3u
.ENDS

* MAIN CIRCUIT
XINVERT 1 AorB VDD INVERT
XNOR A B 1 VDD NOR

SPICE models not shown
.END
```

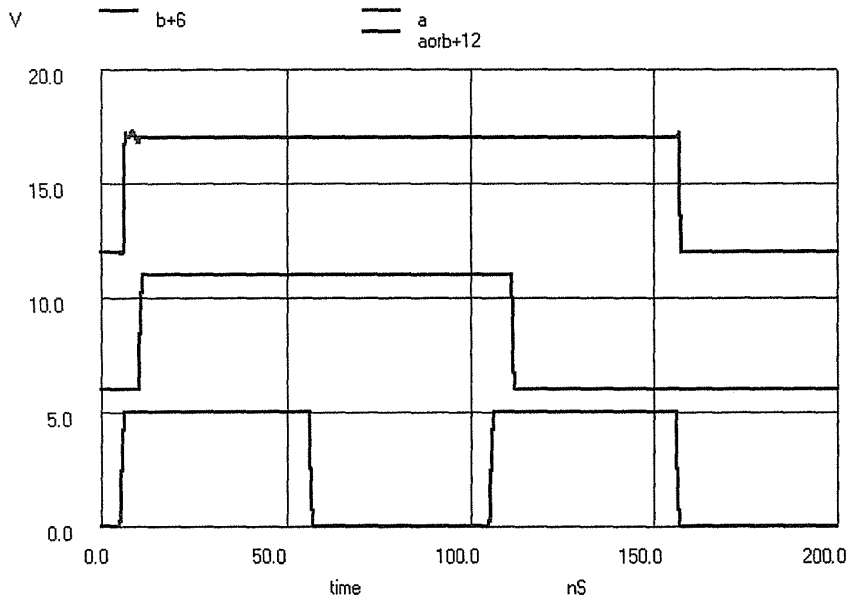



**Figure 8.18** Simulation results for the OR gate.

## REFERENCE

[1] Boyce, D.E. *LASICKT Help Manual*, Available by pressing F1 while running LASICKT.

Microchip Ex. 1055, p. 189
Microchip v. Aptiv
IPR2024-00494

Chapter

9

# Analog MOSFET Models

In this chapter we derive the AC small-signal model of the MOSFET. This model includes the transconductance, output resistance, and capacitances of the MOSFETs. The temperature and noise effects in MOSFETs will also be discussed.

## 9.1 Low-Frequency MOSFET Model

Before we derive the small-signal models for the MOSFET, it will be useful to review the equations derived in Ch. 6 from the BSIM model parameters. The threshold voltage of an n-channel MOSFET is given by

$$V_{THN} = VFB + PHI + K1 \cdot \sqrt{PHI + V_{SB}} - K2 \cdot (PHI + V_{SB}) \tag{9.1}$$

The drain current of the n-channel MOSFET in the saturation region, $V_{DS} > V_{GS} - V_{THN}$, is given by

$$I_D = \frac{MUZ \cdot C_{ox} \cdot W}{2 \cdot L}(V_{GS} - V_{THN})^2[1 + (\lambda_c + \lambda_m)(V_{DS} - V_{DS,sat})] \tag{9.2}$$

where $\lambda_c$ is the channel length modulation parameter and $\lambda_m$ is the mobility modulation parameter. We defined

$$\beta = MUZ \cdot C'_{ox} \cdot \frac{W}{L} = KP \cdot \frac{W}{L} \tag{9.3}$$

so that the drain current in the saturation region can be written as

$$I_D = \frac{\beta}{2}(V_{GS} - V_{THN})^2(1 + \lambda \cdot V_{DS}) \tag{9.4}$$

where we have assumed $V_{DS,sat}$ is approximately 0 and $\lambda = \lambda_c + \lambda_m$. The drain current of the n-channel MOSFET in the linear or triode region, $V_{DS} < V_{GS} - V_{THN}$, is given by

Microchip Ex. 1055, p. 190
Microchip v. Aptiv
IPR2024-00494

$$I_D = \beta \cdot \left[ (V_{GS} - V_{THN}) \cdot V_{DS} - \frac{V_{DS}^2}{2} \right] \tag{9.5}$$

When the MOSFET is operated in the subthreshold region, $V_{GS} < V_{THN}$, the drain current, assuming $V_{DS} > 100$ mV and $V_{GS} < V_{THN}$ - 100 mV, is given by

$$I_{D,weak} = KP \cdot \frac{W}{L} \cdot \left(\frac{kT}{q}\right)^2 e^{1.8} e^{q(V_{GS}-V_{THN})/N0 \cdot kT} = I_{D0} \cdot \frac{W}{L} \cdot e^{q(V_{GS}-V_{TH})/N0 \cdot kT} \tag{9.6}$$

### 9.1.1 Small-Signal Model of the MOSFET in Saturation

Consider the circuit shown in Fig. 9.1. The DC sources will be labeled with upper case letters and subscripts (e.g., $V_{GS}$), and the AC sources will have lower case letters with subscripts (e.g. $v_{gs}$). The sum of the DC and AC components will be labeled with lower case letters and upper case subscripts (e.g. $v_{GS}$). What we are trying to get at in Fig. 9.1 is what happens to the drain current when we apply an AC voltage, $v_{gs}$. We are assuming in the following that $V_{GS} >> v_{gs}$ or in other words the AC signals are small compared to the DC biasing voltages or currents. Since the MOSFET is in the saturation region, $V_{DS} > V_{GS} - V_{THN}$, the total (AC + DC) drain current is given by

$$i_D = i_d + I_D = \frac{\beta}{2}(\overbrace{V_{GS} + v_{gs}}^{v_{GS}} - V_{THN})^2 (1 + \lambda \cdot V_{DS}) \tag{9.7}$$

The forward transconductance, $g_m$, of the MOSFET is given by

$$g_m = \left[ \frac{\partial i_{DS}}{\partial v_{GS}} \right]_{V_{GS}=const}^{I_{DS}=const} = \beta(V_{GS} + v_{gs} - V_{THN})(1 + (\lambda_c + \lambda_m) \cdot V_{DS}) \tag{9.8}$$

Figure 9.2 shows a circuit model using this result. Notice that the DC sources are removed; that is, this circuit is an AC small-signal model of the MOSFET. We can think of the MOSFET as a voltage-controlled current source or a transconductance amplifier. Before we start making simplifying assumptions to make Eq. (9.8) more user friendly, let's examine how the transconductance will change with signal levels. When

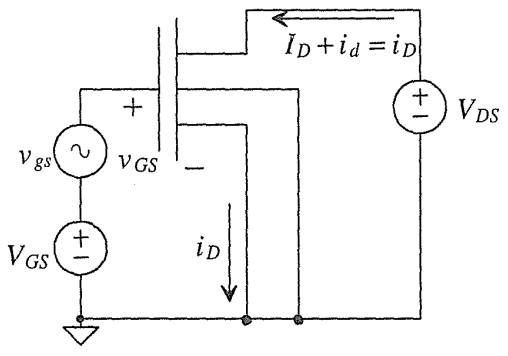



**Figure 9.1** Circuit used to determine the forward transconductance.

Microchip Ex. 1055, p. 191
Microchip v. Aptiv
IPR2024-00494

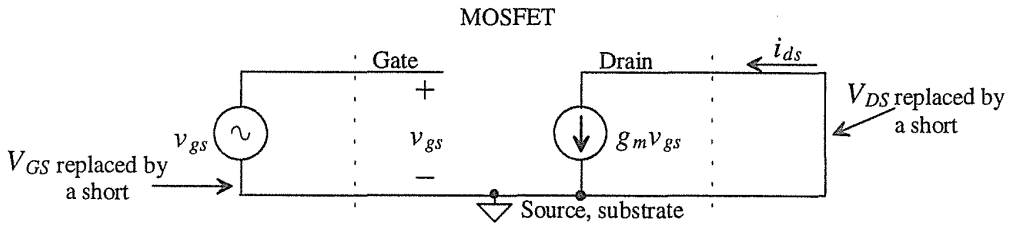



**Figure 9.2** Small-signal model of the circuit in Fig. 9.1.

$v_{gs}$ becomes comparable in amplitude (say, within a factor of 10) to $V_{GS}$, the transconductance variation can be noticeable in the gain of the amplifier. The result for a single-tone (single-frequency sinewave) input to an amplifier is a distorted output. The drain source voltage can modulate the transconductance as well. These problems make "good" (low distortion) large-amplitude voltage amplifiers difficult to design in CMOS. Using feedback can help to reduce the distortion. Another method is to use novel amplifier configurations.

If our MOSFET AC voltages are sufficiently small so that $v_{gs} << V_{GS}$ and the product of the AC and DC components of the drain-source voltage with the modulation parameters are less than 1, that is, $1 >> (\lambda_c + \lambda_m)(v_{ds} + V_{DS})$, then Eq. (9.8) may be written as

$$g_m = \beta(V_{GS} - V_{THN}) = \sqrt{2 \cdot \beta \cdot I_D} \tag{9.9}$$

From Fig. 9.2 we see that the RMS AC drain current is given by

$$i_d = g_m v_{gs} \tag{9.10}$$

The transconductance of the MOSFET operating in the weak inversion (subthreshold region) can be arrived at by first rearranging Eq. (9.6), or

$$\overbrace{v_{gs} + V_{GS}}^{v_{GS}} = \frac{kT \cdot N0}{q} \cdot \ln\left(\frac{I_{D,weak}}{I_{D0} \cdot \frac{W}{L}}\right) + V_{THN} \tag{9.11}$$

The transconductance is then given by

$$g_m^{-1} = \left[\frac{\partial v_{GS}}{\partial i_D}\right]_{I_D = const}^{V_{GS} = const} = \frac{\frac{kT \cdot N0}{q}}{I_{D,weak}} \Rightarrow g_m = \frac{I_D}{V_T} \tag{9.12}$$

where the designation of weak inversion has been dropped from $I_D$ and $V_T = \frac{kT \cdot N0}{q} = 26$ mV @ $T = 300$ K and $N0 = 1$. The conductance of the MOSFET in the subthreshold region increases linearly with $I_D$, while when operation is in the strong inversion region the conductance increases as the root of $I_D$. The model shown in Fig. 9.2 is used for both strong inversion and weak inversion cases.

Microchip Ex. 1055, p. 192
Microchip v. Aptiv
IPR2024-00494

**Example 9.1**

Consider the circuit shown below. Determine $i_d$ and an intuitive way to look at the circuit. Use CN20 BSIM model parameters of Ch. 6 or Appendix A.

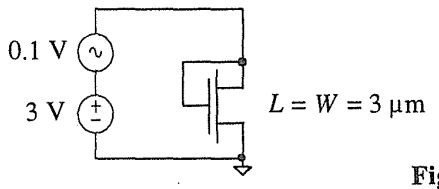



**Figure Ex9.1**

The substrate connection is not drawn, so we will assume that the substrate is connected to the most negative potential in the circuit, or in this case ground. Since $V_{DS} = V_{GS}$, the transistor is in the saturation region, $V_{DS} > V_{GS} - V_{THN}$. The threshold voltage, from Eq. (6.40), is

$$V_{THN} = -0.22 + 1.49\sqrt{0.75 - 0} - 0.315 \cdot (0.75 - 0) = 0.83 \text{ V}$$

The transconductance, using Eq. (9.9), is

$$g_m = 50 \frac{\mu A}{V^2} \cdot \frac{3\,\mu m}{3\,\mu m}(3 - 0.83) = 109 \frac{\mu A}{V}$$

and the RMS drain current is

$$i_d = g_m v_{gs} = (109 \frac{\mu A}{V}) \cdot (100 \text{ mV}) = 10.9\ \mu A$$

Notice that this drain current flows through both the DC and AC supplies in the circuit. An intuitive way to look at this circuit can be arrived at by observing that the ratio of the AC voltage across the MOSFET ($v_{gs}$) to the AC current through the MOSFET ($i_d$) is simply $1/g_m$. For an AC small-signal circuit we can replace the MOSFET with a resistor of value, in this case, of 9.2 kΩ. Note that this intuitive way of looking at the "diode-connected MOSFET" works for both weak and strong inversion. ■

Now let's investigate what happens when the source potential varies and th substrate, gate, and drain are at AC ground. Shown in Fig. 9.3 is the test circuit use to determine how the source-substrate potential variations affect the drain current. I this analysis, we will neglect the forward transconductance effects. These effects aris because $v_{sb} = -v_{gs}$. The body transconductance is given by

$$g_{mb} = -\left[\frac{\partial i_D}{\partial v_{sb}}\right]_{V_{DS}=const}^{I_D=const} = \overbrace{\beta(V_{GS} - V_{THN})(1 + (\lambda_c + \lambda_m)V_{DS})}^{g_m} \cdot \frac{\partial V_{THN}}{\partial v_{sb}} \qquad (9.13)$$

We know that due to the body effect the threshold voltage will not be constant wit changing substrate-source potential. Therefore, changes in $V_{THN}$ will modulate the drai current. From Eq. (6.4)

Microchip Ex. 1055, p. 193
Microchip v. Aptiv
IPR2024-00494

$$\frac{\partial V_{THN}}{\partial v_{sb}} = \frac{K1}{2\sqrt{PHI + V_{SB}}} - K2 = \eta \tag{9.14}$$

where $K1$ is essentially $\gamma$ of Ch. 5. Combining Eqs. (9.8), (9.13), and (9.14) gives

$$g_{mb} = g_m \cdot \eta \tag{9.15}$$

The factor $\eta$ describes how the threshold voltage changes with reverse body bias. Typical values for $\eta$ range from 0.6 to possibly a negative number for large source-substrate voltages. For large substrate-source bias voltages we will assume $\eta$ = 0. The small-signal model with the effects of the body-substrate current source is shown in Fig. 9.4. Notice that $\eta$ is a small-signal parameter that is a function of the DC substrate-source potential. Indeed, all of the small-signal parameters derived in this section are a function of DC biasing.

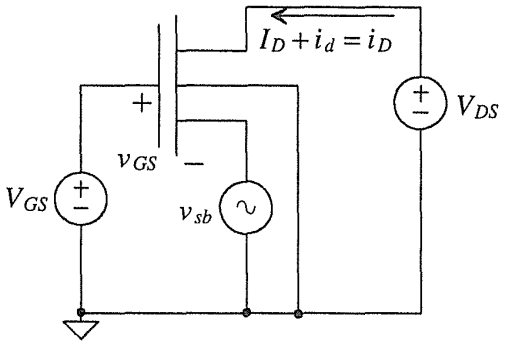



**Figure 9.3** Circuit used to determine the body-source transconductance.

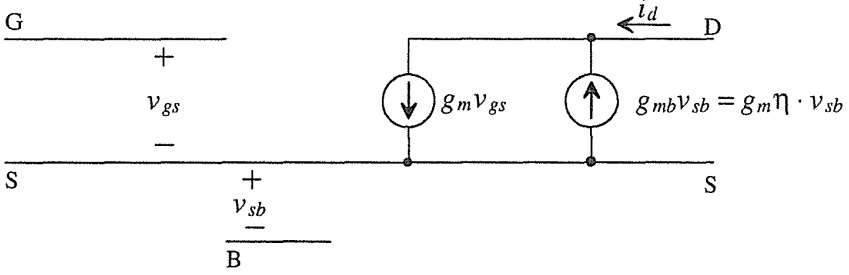



**Figure 9.4** Small-signal MOSFET with body-effect current source.

The output resistance of the MOSFET can be determined with the circuit shown in Fig. 9.5. A small AC test voltage, $v_{ds}$, is applied between the drain and the source. The conductance is determined again by

$$g_{ds} = r_o^{-1} = \left[\frac{\partial i_D}{\partial v_{DS}}\right]_{V_{DS}=const}^{I_D=const} = \frac{\beta}{2}(V_{GS} - V_{THN})^2(\lambda_c + \lambda_m) = I_D \cdot (\lambda_c + \lambda_m) \tag{9.16}$$

Microchip Ex. 1055, p. 194
Microchip v. Aptiv
IPR2024-00494

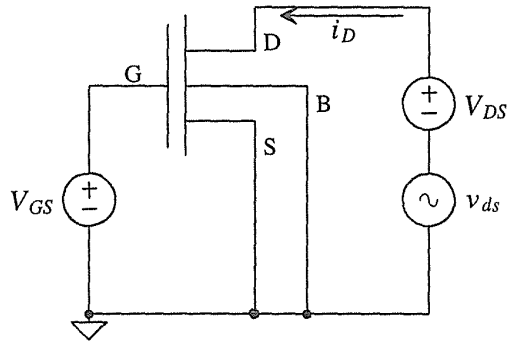



**Figure 9.5** Circuit used to determine output resistance of the MOSFET .

This equation combined with Eq. (9.9) gives the maximum gain of a MOSFET stage. Figure 9.6 shows the small-signal model for the MOSFET, including the output resistance. Assuming the source and substrate are connected together (no body effect, $v_{sb} = 0$), we that the maximum voltage gain, in the strong inversion region, is given by

$$\frac{v_{ds}}{v_{gs}} = \frac{-i_d \cdot r_o}{\frac{i_d}{g_m}} = -g_m \cdot r_o = -\frac{\sqrt{2\beta I_D}}{I_D(\lambda_c + \lambda_m)} = -\frac{\sqrt{2\beta}}{\sqrt{I_D} \cdot \lambda} \tag{9.17}$$

where

$$\lambda = \lambda_c + \lambda_m \tag{9.18}$$

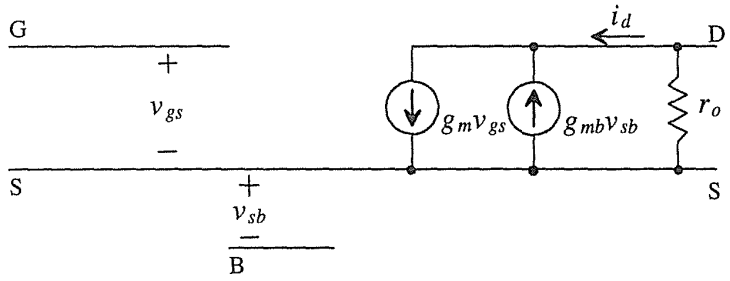



**Figure 9.6** Small-signal MOSFET with output resistance.

**Example 9.2**

Determine the output resistance of an n-channel MOSFET with $L = W = 10$ μm using the BSIM parameters when $V_{GS} = 3$ V. Verify the answer with SPICE.

From Ex. 9.1 $V_{THN} = 0.83$ V, assuming no body effect. The DC drain current is then $I_D = 118$ μA, assuming KP = 50 μA/V². $\lambda \approx \lambda_m = 0.06$ V$^{-1}$ from Ex. 6.4. The output resistance is $r_o = \frac{1}{0.06 \cdot 118 \times 10^{-6}} \approx 140\ k\Omega$. Using the BSIM model, SPICE gives, $I_D = 88$ μA and an $r_o = 285\ k\Omega$ at a $V_{DS} = 3$ V. These values are

Microchip Ex. 1055, p. 195
Microchip v. Aptiv
IPR2024-00494

very dependent on the value of $V_{DS}$ used. This example also illustrates how hand calculations give ballpark-type answers. ■

The small-signal output resistance, or channel resistance, of the MOSFET in the triode region is determined by

$$R_{ch}^{-1} = \left[\frac{\partial i_D}{\partial v_{DS}}\right]_{V_{DS}=const}^{I_D=const} = \frac{\partial}{\partial v_{DS}}\beta\left[(V_{GS}-V_{THN})v_{DS} - \frac{v_{DS}^2}{2}\right] \tag{9.19}$$

and thus,

$$R_{ch} = \frac{1}{\beta(V_{GS}-V_{THN}) - \beta V_{DS}} \approx \frac{1}{\beta(V_{GS}-V_{THN})} \text{ for } V_{DS} << V_{GS}-V_{THN} \tag{9.20}$$

## 9.2 High-Frequency MOSFET Model

To obtain the high-frequency model of the MOSFET, we will add the MOSFET capacitances to the low-frequency model of Fig. 9.6. The capacitances between the drain and source diffusion regions, discussed in Ch. 5, labeled $C_{db}$ and $C_{sb}$, and the capacitance of the gate over the field region, $C_{gb}$, will be added directly to the small-signal model. The capacitance between the gate and the drain is labeled $C_{gd}$ and the capacitance between the gate and source was labeled $C_{gs}$. Figure 9.7 shows the high-frequency model.

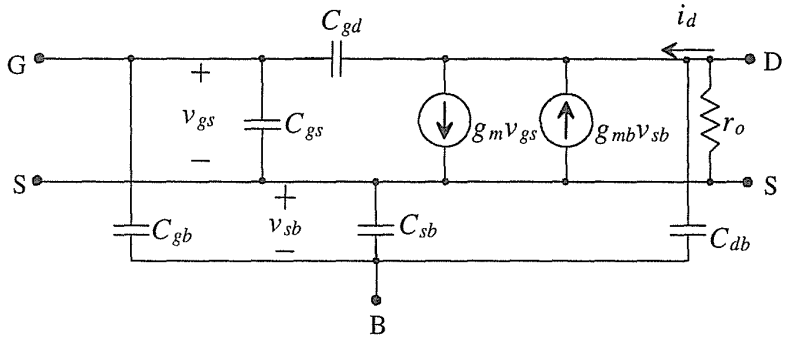



**Figure 9.7** Small-signal high-frequency MOSFET model.

**Example 9.3**

Estimate the capacitances in Fig. 9.7 for the layout (an n-channel MOSFET without substrate connection shown) in Fig. Ex9.3. Use the CN20 model parameters.

Let's start by calculating $C_{sb}$ and $C_{db}$. The size of the drain and source regions is 6 μm by 14 μm. Since the DC potentials of the drain and source is not specified, we will assume that $V_{DB} = V_{SB} = 0$ V. This will give the maximum possible depletion capacitance of these diffused regions. From the BSIM model parameters and Ch. 5

Microchip Ex. 1055, p. 196
Microchip v. Aptiv
IPR2024-00494

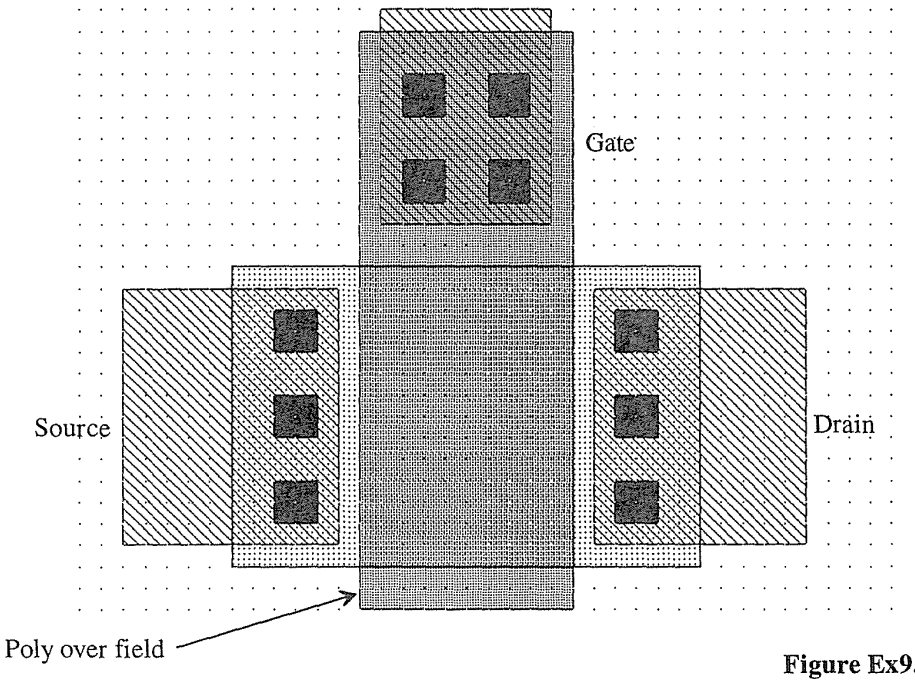



**Figure Ex9.3**

$$C_{sb} = C_{sb,bottom} + C_{sb,sidewall}$$

and

$$C_{db} = C_{db,bottom} + C_{db,sidewall}$$

where

$$C_{sb,bottom} = C_{db,bottom} = \frac{CJ \cdot AD}{\left(1 + \frac{V_{DB}}{PB}\right)^{MJ}} = \frac{(1.04 \times 10^{-4} \frac{F}{m^2}) \cdot (84\ \mu m^2)}{\left(1 + \frac{0}{0.8}\right)^{0.66}} = 8.8\ \text{fF}$$

and

$$C_{sb,sidewall} = C_{db,sidewall} = \frac{CJSW \cdot PD}{\left(1 + \frac{V_{DB}}{PBSW}\right)^{MJSW}} = \frac{(2.2 \times 10^{-10} \frac{F}{m}) \cdot (40\ \mu m)}{(1 + \frac{0}{0.8})^{0.18}} = 8.8\ \text{fF}$$

so that $C_{sb} = C_{db} = 17.6$ fF.

The capacitance of the poly over field region, that is, the poly that extends over the n+ above, $C_{gb}$, is estimated by $CGBO \cdot L = (3.8 \times 10^{-10} F/\text{m}) \cdot 10\ \mu\text{m} = 3.8$ fF. The gate-drain capacitance while in the saturation region is given by $C_{gd} = CGDO \cdot W$, $(3.8 \times 10^{-10}\text{F/m}) \cdot 14\ \mu\text{m} = 5.3$ fF. In the triode region, however, $C_{gd} = \frac{1}{2} C'_{ox} \cdot W \cdot L$ or $C_{gd} = \frac{1}{2} \cdot 800 \times 10^{-18} \frac{\text{F}}{\mu\text{m}^2} \cdot 10\mu\text{m} \cdot 14\mu\text{m} = 56$ fF.

Microchip Ex. 1055, p. 197
Microchip v. Aptiv
IPR2024-00494

Finally, the gate-source capacitance of the MOSFET in the saturation region is $C_{gs} = \frac{2}{3} \cdot C'_{ox} \cdot W \cdot L$, or $C_{gs} = 75$ fF. ■

### 9.2.1 Variation of Transconductance with Frequency

Consider the circuit shown in Fig. 9.8. Since the potential across $C_{db}$ and $C_{sb}$ is fixed, we can eliminate them from the following frequency analysis. The small-signal equivalent circuit of Fig. 9.8 is shown in Fig. 9.9. The AC gate-source voltage of the MOSFET is given by

$$v_{gs}(f) = v_s \cdot \frac{\frac{1}{j\omega(C_{gb}+C_{gs}+C_{gd})}}{Z_s + \frac{1}{j\omega(C_{gb}+C_{gs}+C_{gd})}} = v_s \cdot \frac{1}{1+j\omega \cdot Z_s(C_{gb}+C_{gs}+C_{gd})} \tag{9.21}$$

with the transconductance given by, assuming $g_m v_{gs} >> v_{gs} j\omega C_{gd}$,

$$g_m(f) = \frac{i_d}{v_s} = \frac{g_{m0}}{1+j\omega \cdot Z_s(C_{gb}+C_{gs}+C_{gd})} \tag{9.22}$$

where $g_{m0} = \frac{i_d}{v_{gs}(0)}$ is the low-frequency transconductance.

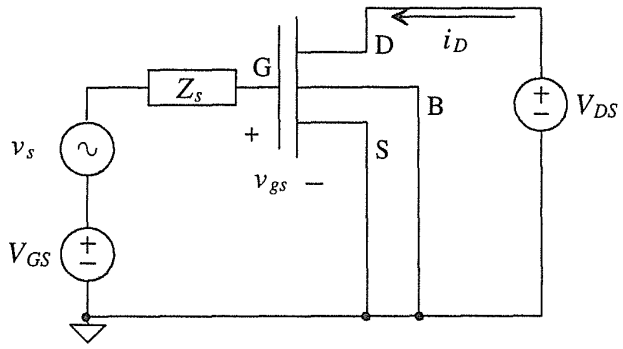



**Figure 9.8** Circuit used to determine frequency dependence of the forward transconductance.

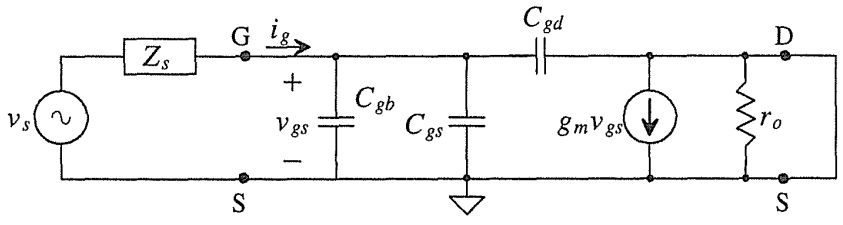



**Figure 9.9** Small-signal equivalent circuit of Fig. 9.8.

Microchip Ex. 1055, p. 198
Microchip v. Aptiv
IPR2024-00494

The current gain of a MOSFET can be defined by

$$\frac{i_d}{i_g} = \frac{g_m \cdot v_{gs}}{v_{gs} \cdot j\omega(C_{gb} + C_{gs} + C_{gd})} = \frac{KP \cdot \frac{W}{L} \cdot (V_{GS} - V_{THN})}{j\omega(C_{gb} + C_{gs} + C_{gd})} \tag{9.23}$$

A useful term used to describe the relative "speed" of a process is the unity current gain frequency, $f_T$, and is defined as the frequency where the magnitude of the MOSFET current gain is unity,

$$\left|\frac{i_d}{i_g}\right| = 1 = \frac{KP \cdot \frac{W}{L} \cdot (V_{GS} - V_{THN})}{2\pi f_T \cdot (C_{gb} + C_{gs} + C_{gd})} \tag{9.24}$$

or

$$f_T = \frac{KP \cdot W(V_{GS} - V_{THN})}{2\pi L(C_{gb} + C_{gs} + C_{gd})} \approx \frac{KP \cdot W}{2\pi L C_{gs}}(V_{GS} - V_{THN}) \tag{9.25}$$

This expression neglects the transit time of the carriers through the channel, which in general, for channel lengths within an order of magnitude of minimum length, is a negligible effect.

**Example 9.4**

Estimate $f_T$ for a minimum-size n- and p-channel device in CN20 with $V_{GS} - V_{THN} = 1$ V.

For the n-channel MOSFET with $L = 2$ μm and $W = 3$ μm

$$f_T \approx \frac{KP \cdot W}{2\pi L \cdot \frac{2}{3} W L C'_{ox}} = \frac{50\frac{\mu A}{V^2} \cdot 3\mu m}{2\pi \cdot 2\mu m \cdot \frac{2}{3} \cdot 2\mu m \cdot 3\mu m \cdot 800 \times 10^{-18}\frac{F}{\mu m^2}} = 3.74 \text{ GHz}$$

and for the p-channel with $KP = 17\frac{\mu A}{V^2}$, $f_T = 1.27$ GHz. ■

The most important observation we can make from this example is that the unity current gain frequency is independent of the MOSFET width, or

$$f_T = \frac{3 \cdot KP}{4\pi L^2 C'_{ox}}(V_{GS} - V_{THN}) \tag{9.26}$$

For high-speed design the minimum channel length should be used.

**Example 9.5**

For the circuit shown in Fig. EX9.5, determine the impedance looking into the source of the MOSFET. Neglect the body effect and assume that over the frequency range of interest $i_d \approx i_s$.

Since $i_d \approx i_s$ we know that $v_s \approx -v_{gs}$ and $i_d \approx -g_m v_s$. The impedance looking into the source is then given by

$$Z_{into\ source} = \frac{-v_s}{i_s} = \frac{1}{g_m} = \frac{1}{g_{m0}} \cdot [1 + j\omega R_s(C_{gb} + C_{gs} + C_{gd})]$$

Microchip Ex. 1055, p. 199
Microchip v. Aptiv
IPR2024-00494

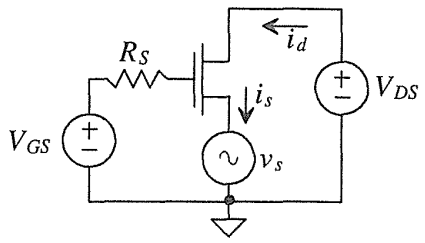



**Figure Ex9.5**

or

$$Z_{into\ source} = \frac{1}{g_{m0}} + j\omega\frac{R_s(C_{gb}+C_{gs}+C_{gd})}{g_{m0}} \tag{9.27}$$

or the impedance looking into the source of the MOSFET is a resistor of value $\frac{1}{g_{m0}}$ in series with an inductor of value $\frac{R_s(C_{gb}+C_{gs}+C_{gd})}{g_{m0}}$. ■

The results of this example have several important practical implications. First, the low-frequency resistance looking into the source of a MOSFET is $\frac{1}{g_{m0}}$ regardless of the impedance at the gate. When we are not concerned with the variation of the transconductance with frequency, we will write this resistance as $\frac{1}{g_m} = \frac{1}{\beta(V_{GS}-V_{THN})}$. When the MOSFET is driven with a resistive source impedance, the impedance looking into the source of the MOSFET becomes highly inductive, which will cause ringing in source-follower circuits driving capacitive loads. If the impedance at the gate of the MOSFET is inductive, this occurs in the cascade of two source followers; the output impedance of the second follower can become negative (replace $R_s$ in Eq. [9.26] with the inductor impedance $j\omega L$). A negative impedance is a source of power like a battery. Oscillations and poor step response are probable.

## 9.3 Temperature Effects in MOSFETs

The threshold voltage and the transconductance parameter change with temperature. The changes in the threshold voltage can be related to the temperature dependence of the surface inversion potential given by

$$\phi_s(T) = PHI(T) = \frac{PHI \cdot T}{T_0} - \frac{3kT}{q}\cdot\ln\left(\frac{T}{T_0}\right) - \frac{E_g(T_0)\cdot T}{T_0} + E_g(T) \tag{9.28}$$

where $E_g$, in eV (1 eV = $1.6\times10^{-19}$ J), is the silicon bandgap energy given by

$$E_g(T) = 1.16 - (702\times10^{-6})\cdot\frac{T^2}{T+1108} \tag{9.29}$$

The temperature coefficient of the threshold voltage over a range from –100 °C to 100 °C can be estimated as

$$TCV_{THN} = \frac{1}{V_{THN}}\cdot\frac{dV_{THN}}{dT} \approx -3,000\ \frac{ppm}{^\circ C} \tag{9.30}$$

where the threshold voltage is given by

Microchip Ex. 1055, p. 200
Microchip v. Aptiv
IPR2024-00494

$$V_{THN}(T) = V_{THN}(T_0)[1 + TCV_{THN} \cdot (T - T_0)] \tag{9.31}$$

A PSPICE simulation of a CN20 n-channel MOSFETs threshold voltage variation with temperature is shown in Fig. 9.10. At these low currents, the temperature dependence of the threshold voltage dominates the change in the drain current. We could have also defined an absolute change of $\approx -2.4\ mV/°C$ ($TCV_{THN} \cdot V_{THN}$), dependent on the magnitude of the threshold voltage. Note that SPICE2 or 3 from UC Berkeley does not have temperature- or noise-modeling capability with the BSIM1 model.

```
*** Top Level Netlist ***
M1_10u_10u    3 4 0 0 CMOSNB L=10u W=10u
Vdd     2 0     DC 5
Vgs     4 0     DC 0
VIMTR1 2 3    0V
***** Spice models and macro models *****
.MODEL CMOSNB NMOS LEVEL=4
+VFB=-9.73820E-01, LVFB=3.67458E-01,WVFB=-4.72340E-02
+ ... BSIM model parameters
+pbsw=0.8   mj=0.66036   mjsw=0.178543   wdf=0   dell=0
***** End of spice models and macro models *****
.probe
.DC Vgs .5 1.5 .05  temp -100 101 50
.end
```

The temperature dependence of the transconductance parameter is derived from the temperature dependence of the mobility where T is in kelvin given by

$$\mu(T) = \mu(T_0) \cdot \left(\frac{T}{T_0}\right)^{-1.5} \tag{9.32}$$

or

$$KP(T) = KP(T_0) \cdot \left(\frac{T}{T_0}\right)^{-1.5} \tag{9.33}$$

Figure 9.11 shows that at low-drain currents the effects of the surface mobility dominate the changes in the drain current, while at higher currents the mobility effects will dominate. For digital applications, the change in threshold voltage is usually negligible compared to the mobility changes; that is,, the mobility changing generally has a much greater impact on the propagation delay than does the threshold voltage. From Fig. 9.11 we can also see that at $V_{GS}$ = 1.8 V the drain current is essentially constant. It could be concluded that biasing the device at this gate-source voltage would create temperature- independent circuits. Indeed, this is true. However, variations in the process and a method to accurately bias at this voltage make this result difficult to implement in many cases. (One implementation uses the beta multiplier of Ch. 21.)

**Example 9.6**

One benefit of using MOSFET transistors in power applications is that the drain current decreases with increasing temperature, keeping the transistor from thermal runaway. Using Fig. 9.11, comment on this assumption.

Microchip Ex. 1055, p. 201
Microchip v. Aptiv
IPR2024-00494

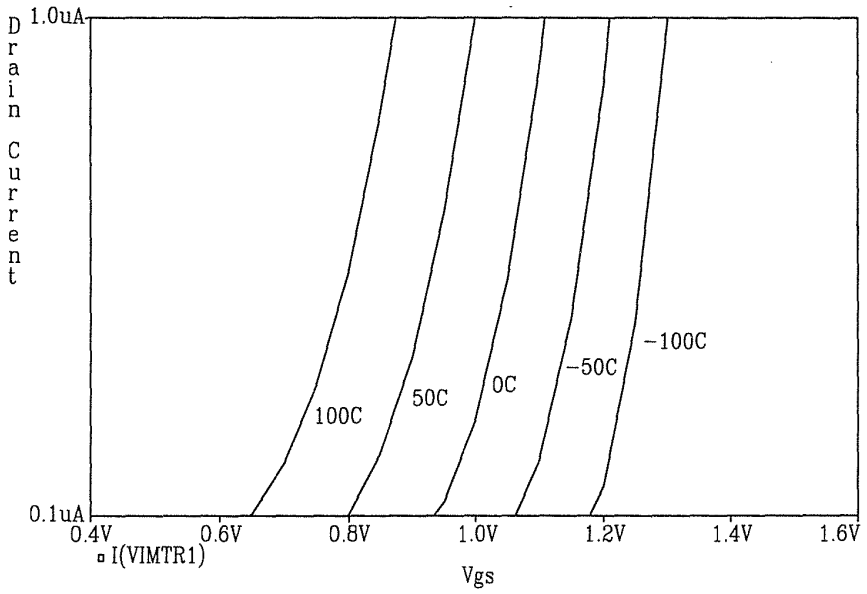



**Figure 9.10** Threshold voltage change with temperature for $V_{GS}$ approximately $V_{THN}$.

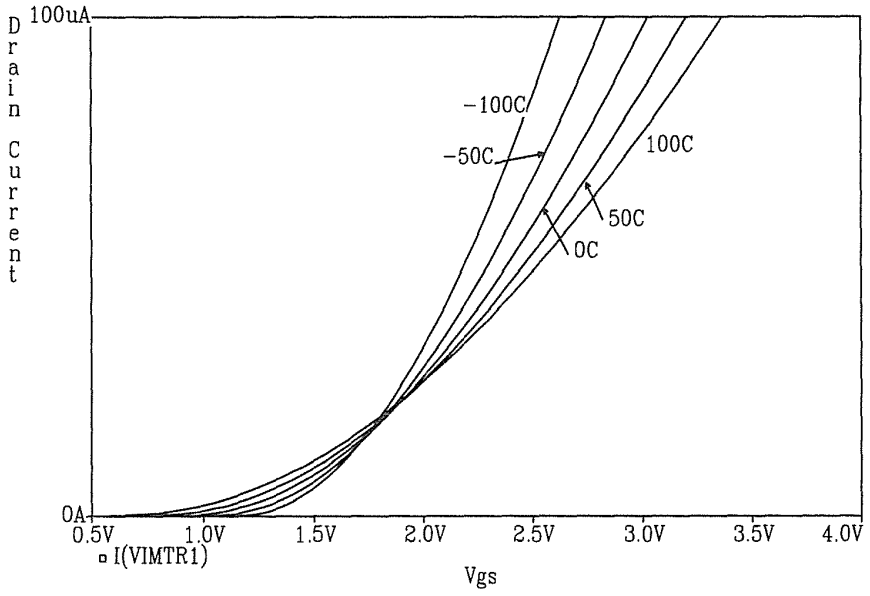



**Figure 9.11** Mobility effects when $V_{GS} >> V_{THN}$.

Microchip Ex. 1055, p. 202
Microchip v. Aptiv
IPR2024-00494

From Fig. 9.11 if we hold the gate-source voltage at 2.5 V, the drain current will decrease with increasing temperature. At $V_{GS}$ = 2.5 V, the variation of the mobility with temperature dominates. If we look at $V_{GS}$ = 1.25 V, we see that the drain current increases with increasing temperature, due to the surface inversion potential dominating the temperature characteristics. Therefore, unless $V_{GS}$ >> $V_{THN}$, the problem of thermal runaway still exists. ■

## 9.4 Noise in MOSFETs

The noise generators in MOSFETs are due to thermal and flicker (1/*f* noise, pronounced "one over f noise") noise. The RMS thermal noise current is generated by the effective channel resistance and by the parasitic drain ($R_D$), source ($R_S$), gate ($R_G$), and substrate ($R_B$) resistances, and is given by

$$\sqrt{\overline{i_{RD}^2}} = \sqrt{\frac{4kT}{R_D}} \tag{9.34}$$

$$\sqrt{\overline{i_{RS}^2}} = \sqrt{\frac{4kT}{R_S}} \tag{9.35}$$

$$\sqrt{\overline{i_{RG}^2}} = \sqrt{\frac{4kT}{R_G}} \tag{9.36}$$

$$\sqrt{\overline{i_{RB}^2}} = \sqrt{\frac{4kT}{R_B}} \tag{9.37}$$

where (see Sec. 7.3, $k = 1.38 \times 10^{-23}$ $W{\cdot}s/°K$) the temperature, $T$, is in kelvin, and the measurements are made over a bandwidth, $B$.

The thermal noise due to the channel resistance, modeled by a resistor of $\frac{2}{3} \cdot \frac{1}{g_m}$ while in the saturation region and $R_{CH}$ in the triode region, is given, in the saturation region, by

$$\sqrt{\overline{i_{therm}^2}} = \sqrt{4kT \cdot \frac{3g_m}{2}} = \sqrt{6kT \cdot \sqrt{2\beta \cdot I_D}} \tag{9.38}$$

This is a noise source placed between the drain and source of the MOSFET.

The 1/*f* noise can be modeled by an RMS noise source given by

$$\sqrt{\overline{i_{1/f}^2}} = \sqrt{\frac{KF \cdot I_D^{AF}}{f \cdot C'_{ox} \cdot L^2}} \tag{9.39}$$

where $KF$ is the flicker noise coefficient, a typical value is $10^{-25}$ $V^2F$, $I_D$ is the DC drain current, $AF$ is the flicker noise exponent, a value ranging from 0.5 to 2, and $f$ is the frequency variable we integrate over. A noise model for the MOSFET is shown in Fig. 9.12. All noise sources are RMS. In addition, the voltage or current gain in an

Microchip Ex. 1055, p. 203
Microchip v. Aptiv
IPR2024-00494

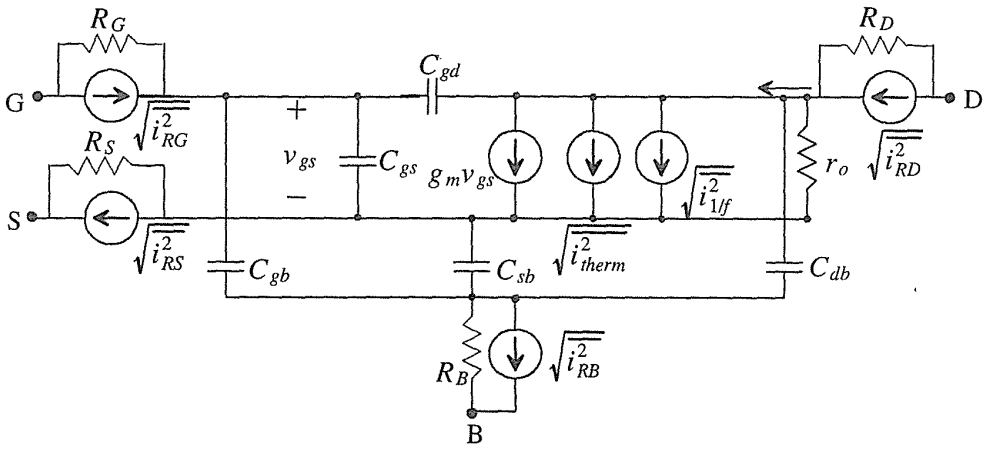



**Figure 9.12** Noise model for MOSFET.

amplifier can be used instead of the power gain. See Sec. 7.3 for a review of noise calculations.

For hand calculations, the thermal noise effects due to the parasitic resistances will be neglected. If this is the case, the only noise sources we will evaluate using hand calculations are the channel thermal and flicker noise contributions. Both of these noise sources can be reflected back to the gate of the MOSFET. The RMS magnitudes of the noise voltages in series with the gate terminal are given by

$$\sqrt{v^2_{therm}} = \frac{\sqrt{i^2_{therm}}}{g_m} \tag{9.40}$$

and

$$\sqrt{v^2_{1/f}} = \frac{\sqrt{i^2_{1/f}}}{g_m} \tag{9.41}$$

The body effect has been neglected. If the source is not connected to the substrate, then $g_m$ should be replaced with $g_m - g_{mb}$ in the above equations, including Eq. (9.36). Sometimes Eqs. (9.38) and (9.39) are used to get an intuitive glimpse of how MOSFET noise, on the input to an amplifier, will limit the minimum input signal amplitude.

**Example 9.7**

Estimate the RMS output noise for the circuit of Fig. Ex9.7 over the bandwidth of 1 to 1 kHz. Use CN20 process parameters with $L = W = 10$ μm and $KF = 10^{-25}$ $V^2F$ and $AF = 1.3$. Assume that the parasitic MOSFET resistances generate a negligible amount of thermal noise.

The DC gate voltage is approximately 3.3 V and is determined by solving

$$\frac{5 - V_{GS}}{10\text{k}} = \frac{\beta}{2}(V_{GS} - V_{THN})^2 = 25\frac{\mu\text{A}}{\text{V}^2}(V_{GS} - 0.83)^2$$

Microchip Ex. 1055, p. 204
Microchip v. Aptiv
IPR2024-00494

The drain current is 153 μA. The noise model for the circuit is shown above, following step 1 given in Ch. 7. Using superposition, we can determine the RMS output noise voltage from each noise source. Noting that the gate-source voltage and the output voltage are the same signal, we can write, for the thermal noise current source modeling, the noise generated by the 10 k resistor,

$$10\text{k}\left(\sqrt{\overline{i_{10k}^2}} + g_m v_{gs}\right) = -v_{gs}$$

or

$$v_{gs} = V_{out} = -\sqrt{\overline{i_{10k}^2}} \cdot \frac{\frac{1}{g_m} \cdot 10\text{k}}{10\text{k} + \frac{1}{g_m}}$$

This shows that the MOSFET behaves like a resistor of value $\frac{1}{g_m}$ when its gate and drain are tied together. Therefore, to save time and effort we would simply replace the current source above ($g_m v_{gs}$) with this noiseless resistor and perform the noise calculations.

We continue the noise analysis by determining the output RMS noise voltage from each source. Note that you cannot add the noise sources into one noise source, unless you first square each component. The thermal noise from the channel resistance is given by

$$\sqrt{\overline{i_{therm}^2}} = \sqrt{6 \cdot 1.38 \times 10^{-23} \cdot 300 \cdot \sqrt{4 \cdot 25\mu\text{A/V}^2 \cdot 153\mu\text{A}}} = 1.8 \frac{\text{pA}}{\sqrt{\text{Hz}}}$$

and since $\frac{1}{g_m} = 8.1\ \text{k}\Omega$

$$\sqrt{\overline{v_{therm,out}^2}} = 1.8\frac{\text{pA}}{\sqrt{\text{Hz}}} \cdot 10\text{k}||8.1\text{k} = 8.1\ \frac{\text{nV}}{\sqrt{\text{Hz}}}$$

while the *1/f* (flicker) noise is given by

$$\sqrt{\overline{i_{1/f,d}^2}} = \sqrt{\frac{10^{-25} \cdot (150\mu\text{A})^{1.3}}{f \cdot (10)^2 \cdot (800 \times 10^{-18})}} = \frac{3.7\ \text{nA}}{\sqrt{f}}$$

with output RMS noise voltage due 1/*f* noise as

$$\sqrt{\overline{v_{1/f,out}^2}} = 3.7\frac{\text{nA}}{\sqrt{f}} \cdot 10\text{k}||8.1\text{k} = 16.6\ \frac{\mu\text{V}}{\sqrt{f}}$$

The output thermal RMS noise from the 10 kΩ resistor is given by

$$\sqrt{\overline{v_{10k}^2}} = 10\text{k}||8.1\text{k} \cdot \sqrt{\frac{4kT}{10\text{k}}} = 5.8\frac{\text{nV}}{\sqrt{\text{Hz}}}$$

We square each RMS component and integrate over the bandwidth of interest. The total mean squared output noise voltage is given by

Microchip Ex. 1055, p. 205
Microchip v. Aptiv
IPR2024-00494

$$\overline{v_{on}^2} = \int_{f_L}^{f_H} \left( \overbrace{65.6\,\frac{\text{aV}^2}{\text{Hz}}}^{thermal} + \overbrace{\frac{276\times10^{-12}\text{V}^2}{f}}^{1/f} + \overbrace{33.6\frac{\text{aV}^2}{\text{Hz}}}^{res.\,thermal} \right) df$$

or evaluating the integral

$$\overline{v_{on}^2} = (65.6\,\frac{\text{aV}^2}{\text{Hz}} + 33.6\,\frac{\text{aV}^2}{\text{Hz}}) \cdot (f_H - f_L) + \overbrace{276\ \text{pV}^2 \cdot (\ln f_H - \ln f_L)}^{1/f}$$

The total RMS output noise voltage over the bandwidth of 1 to 1 kHz is given by $\sqrt{\overline{v_{on}^2}} = 43.7\ \mu\text{V}$. The RMS output noise is dominated by the $1/f$ noise. From the results above, we observe that the same amount of $1/f$ noise exists in a bandwidth of 1 to 1 kHz as a bandwidth of 1 MHz to 1 GHz, or a bandwidth of once per day to 1,000 times a day. ■

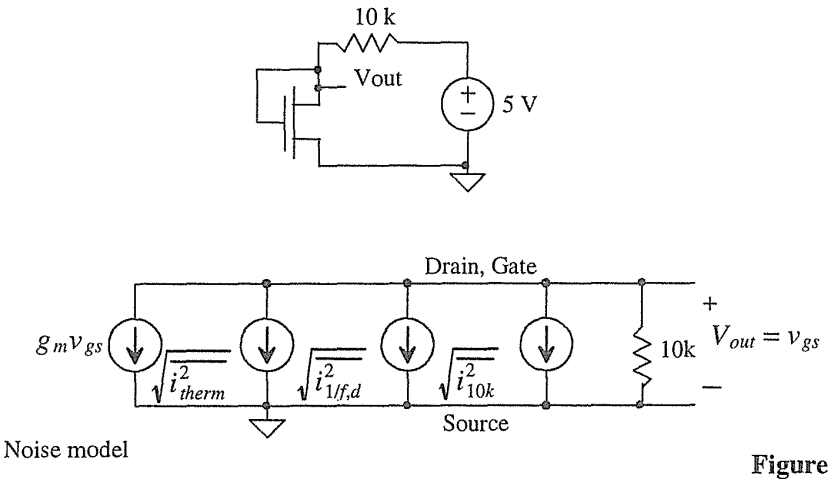



**Figure Ex9.7**

## REFERENCES

[1] F. Maloberti, "Layout of Analog and Mixed Analog-Digital Circuits," in J. E. Franca and Y. Tsividis, eds., *Design of Analog-Digital VLSI Circuits for Telecommunications and Signal Processing*, 2nd ed., Prentice-Hall, 1994. ISBN 0-13-203639-8.

[2] C. D. Motchenbacher and F. C. Fitchen, *Low-Noise Electronic Design,* John Wiley and Sons, 1973. ISBN 0-471-61950-7.

[3] H. L. Krauss, C. W. Bostian, and F. H. Raab, *Solid State Radio Engineering,* John Wiley and Sons, 1980. ISBN 0-471-03018-X.

**Microchip Ex. 1055, p. 206**
**Microchip v. Aptiv**
**IPR2024-00494**

## PROBLEMS

Unless otherwise stated, use the CN20 process.

**9.1** Sketch the p-channel equivalent to Fig. 9.6 using the p-channel variables, $v_{sg}$ and $v_{bs}$. Are these small-signal voltages positive or negative numbers? Refer to Appendix A.

**9.2** Plot the drain current against $V_{GS}$ for the MOSFET shown in Fig. P9.2 using a SPICE DC sweep. What is the approximate slope of a line tangent to this curve at $V_{GS} = 2$ V. What does the reciprocal of this slope correspond to?

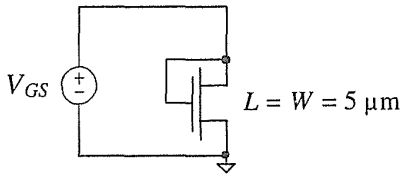



**Figure P9.2**

**9.3** Determine the AC drain current for the circuit given in Ex. 9.1 if $L = 2$ μm and $W = 5$ μm with $V_{GS} = 1.5$ V and $v_{gs} = 10$ mV. Verify your answer with an AC SPICE simulation.

**9.4** Estimate, using hand calculations, the AC drain current that flows in the following circuit. The MOSFET has $W = 15$ μm and $L = 5$ μm.

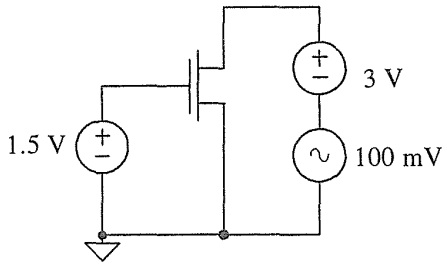



**Figure P9.4**

**9.5** Verify the answer in Problem 4 using SPICE.

**9.6** Repeat Problem 4 with $V_{GS} = 3$ V and $V_{DS} = 1.5$ V ($v_{ds}$ is still 100mV).

**9.7** Plot the drain current against $V_{SG}$ for the MOSFET shown in Fig. P9.7 using a SPICE DC sweep.

**9.8** Estimate the AC drain current that flows in the circuit of Fig. P9.8. The p-channel MOSFET has $W = 70$ μm and $L = 5$ μm. How does the AC drain current change if $VDD = 0$, that is, the source of the p-channel MOSFET is connected to ground?

Microchip Ex. 1055, p. 207
Microchip v. Aptiv
IPR2024-00494

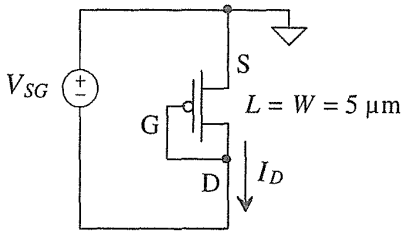



**Figure P9.7**

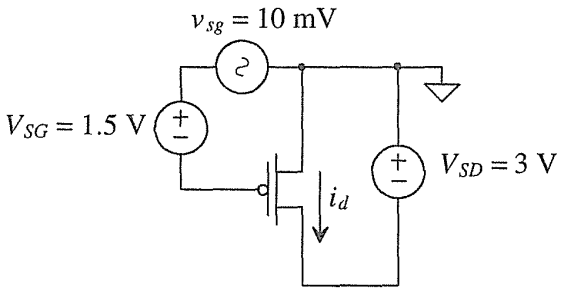



**Figure P9.8**

**9.9** Verify the answer in Problem 8 using SPICE.

**9.10** Repeat Problem 8 for the circuit of Fig. P9.10 and verify the answer with SPICE.

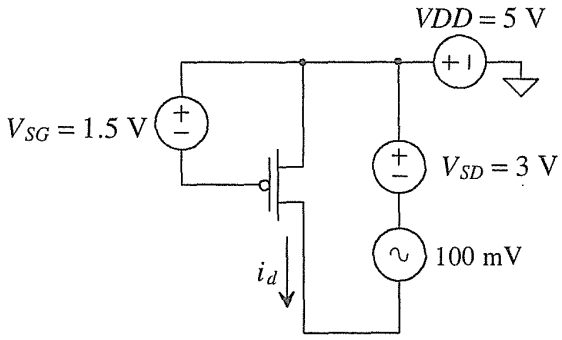



**Figure P9.10**

**9.11** Repeat Ex. 9.3 where the layout shown in Fig. Ex9.3 is a p-channel MOSFET in an n-well.

**9.12** Figure P9.12 shows the test setup used to determine a MOSFET $f_T$. Using this setup and SPICE, verify the results of Ex. 9.4. Remember that the MOSFET must be biased in the saturation region. How are the drain current and gate current related at the frequency $f_T$ and at DC?

**9.13** For wide operating range in analog circuits $V_{GS} - V_{THN}$ is made very small. Comment on how this affects the speed of a circuit.

Microchip Ex. 1055, p. 208
Microchip v. Aptiv
IPR2024-00494

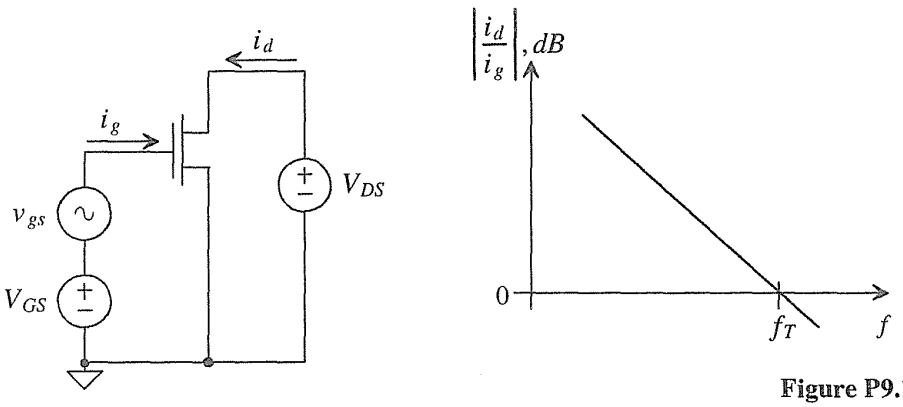



**Figure P9.12**

**9.14** Does the n-channel MOSFET threshold voltage increase or decrease with increasing temperature? Some CMOS processes have MOSFETs with threshold voltages around 0.5 V. Comment on the problems associated with decreasing the threshold voltage in a CMOS process down to a few hundred millivolts. Use drawings to illustrate the problems. (*Hint*: Consider temperature effects and subthreshold current.)

**9.15** Verify, with SPICE, the RMS output noise voltage of 43.7 μV given in Ex. 9.7.

**9.16** Determine the RMS output noise in the following circuit over the frequency range from 1 to 10 kHz. Verify your answer with SPICE. Neglect parasitic MOSFET resistances and assume that the device capacitances have little effect on the noise performance over this frequency range.

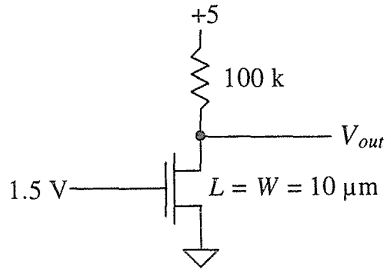



**Figure P9.16**

Microchip Ex. 1055, p. 209
Microchip v. Aptiv
IPR2024-00494

Chapter

10

# The Digital Model

In this chapter we present the digital model of the MOSFET. At this point, the student should feel relatively comfortable with simulating and laying out CMOS circuits and the parasitics associated with the CMOS process. The transition into digital circuit design should be relatively straightforward.

## 10.1 The Digital MOSFET Model

Consider the MOSFET circuit shown in Fig. 10.1. Initially, the MOSFET is off, $V_{GS}$ = 0, and the drain of the MOSFET is at *VDD*. If the gate of the MOSFET is taken instantaneously from 0 to *VDD*, a current given by

$$I_D = \frac{KP_n}{2} \cdot \frac{W}{L} \cdot (VDD - V_{THN})^2 = \frac{\beta}{2} \cdot (VDD - V_{THN})^2 \qquad (10.1)$$

initially flows through the MOSFET. Point A in Fig. 10.2 shows the operating point of the MOSFET prior to switching for *VDD* = 5 V. After switching takes place, the operating point moves to point B and follows the curve $V_{GS}$ = *VDD* down to $I_D \approx 0$ and $V_{DS}$ = 0, point C.

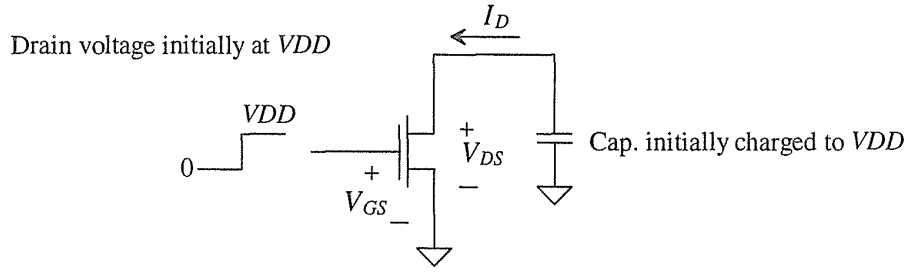



**Figure 10.1** MOSFET switching circuit.

Microchip Ex. 1055, p. 210
Microchip v. Aptiv
IPR2024-00494

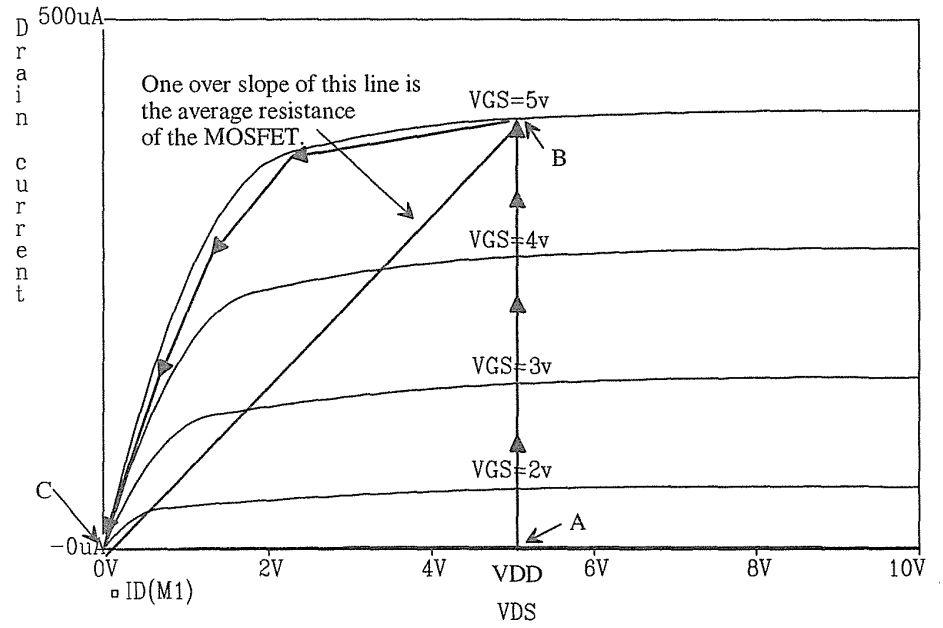



**Figure 10.2** Diagram used to determine average resistance of a MOSFET during switching.

An estimate for the resistance between the drain and source of the MOSFET is given by the reciprocal slope of the line BC in Fig. 10.2, or

$$R_n = \frac{VDD}{\frac{KP_n}{2}\frac{W}{L}\cdot(VDD - V_{THN})^2} = R_n' \cdot \frac{L}{W} \tag{10.2}$$

The MOSFET is modeled by the circuit shown in Fig. 10.3. When $V_{GS} > VDD/2$, the switch is closed while $V_{GS}$ values less than $VDD/2$ the switch is open. In the derivation of this model, we assumed that the input step transition occurred in zero time; that is, the risetime was zero, so that the point at which the switch was opened or closed was well defined. In practice, we will not encounter a zero risetime pulse; therefore, the model has limitations. Nevertheless, the model works remarkably well in designing and analyzing digital circuits, giving results that are within a factor of two of simulation or measurement in general applications.

Application of the BSIM model parameters to this digital model can, to a first order, predict the increase in effective resistance for short channel devices. For short channel devices, the drain current increases linearly with $V_{GS}$ rather than as the square. Characterization of these MOSFETs usually results in a lower value of MUZ, accounting for the mobility degradation effects present. The digital model resistance can be written in terms of the BSIM model parameters by

$$R_n = \frac{2L \cdot VDD}{MUZ \cdot C_{ox}' \cdot W \cdot (VDD - V_{THN})^2} = R_n' \cdot \frac{L}{W} \tag{10.3}$$

Microchip Ex. 1055, p. 211
Microchip v. Aptiv
IPR2024-00494

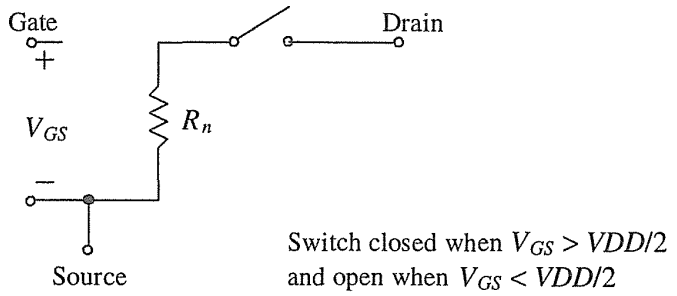



**Figure 10.3** Simple digital MOSFET model.

### 10.1.1 Capacitive Effects

At this point, we need to add the capacitances of the switching MOSFET to our model of Fig. 10.3. Consider the MOSFET shown in Fig. 10.4 with capacitance $C_{ox}/2$ between the gate-drain and the gate-source electrodes. This is the capacitance when the MOSFET is in the triode region. In our development of the digital MOSFET model, we will neglect the depletion capacitances of the source and drain implants to substrate. When the input pulse transitions from 0 to *VDD*, the output transitions from *VDD* to 0. The current through $C_{gd}$ (= $C_{ox}/2$), assuming a linear transition, is given by

$$I = C_{gd} \cdot \frac{dV_{gd}}{dt} = \frac{C_{ox}}{2} \cdot \frac{VDD-(-VDD)}{\Delta t} = C_{ox} \cdot \frac{VDD}{\Delta t} = C_{ox} \cdot \frac{dV_{DS}}{dt} \qquad (10.4)$$

The voltage across $C_{gd}$ changes by $2 \cdot VDD$. The current that flows through this capacitance is the drain current of the MOSFET in Fig. 10.4. We can break $C_{gd}$ into a component from the gate to ground and from the drain to ground of value $2C_{gd}$ or $C_{ox}$. The complete model of a switching MOSFET is shown in Fig. 10.5.

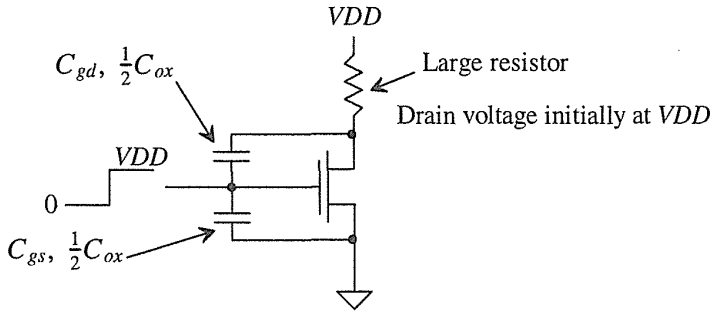



**Figure 10.4** MOSFET switching circuit with capacitances.

Microchip Ex. 1055, p. 212
Microchip v. Aptiv
IPR2024-00494

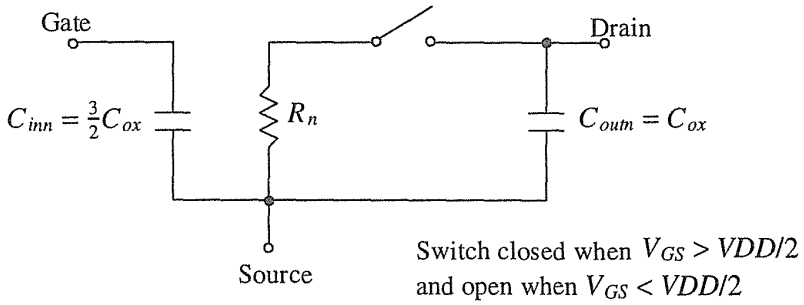



**Figure 10.5** Simple digital MOSFET model.

### 10.1.2 Process Characteristic Time Constant

An important question we can answer at this point is, "What is the intrinsic switching speed of a MOSFET?" Looking at Figs. 10.4 and 10.5, we can see an intrinsic time constant of $R_nC_{ox}$. That is, if the drain is charged to *VDD* as in Fig. 10.4 and the input switches from 0 to *VDD*, the output voltage will decay with a time constant of $R_nC_{ox}$. For an n-channel transistor, this is given by

$$\tau_n = R_nC_{ox} = \frac{2L \cdot VDD}{KP_nW(VDD - V_{THN})^2} \cdot C'_{ox}WL = \frac{2L^2C'_{ox} \cdot VDD}{KP_n \cdot (VDD - V_{THN})^2} \qquad (10.5)$$

Notice that the "speed" of a process increases as the square of the channel length and that it is independent of the channel width, *W*. Also note that the larger *VDD*, the faster the process. This is very similar to the unity current gain frequency, $f_T$, we discussed in the last chapter.

---

**Example 10.1**

Estimate the process characteristic time constants for CN20, both n- and p-channel devices, using the BSIM model parameters.

We can start the solution of this problem by finding $R'_n$ and $R'_p$ using Eq. (10.3). For the n-channel,

$$R_n = R'_n \cdot \frac{L}{W} = \frac{2 \cdot VDD}{MUZ \cdot C'_{ox}(VDD - V_{THN})^2} \cdot \frac{L}{W} = \frac{2 \cdot 5 \cdot (L/W)}{\left(598\frac{\text{cm}^2}{\text{V·s}}\right)\left(800\frac{\text{aF}}{\mu\text{m}^2}\right)\left(\frac{10^8\mu\text{m}^2}{\text{cm}^2}\right)(5 - 0.83)^2}$$

$$= 12\ \text{k}\Omega \cdot \frac{L}{W}$$

and for the p-channel

$$R_p = R'_p \cdot \frac{L}{W} = \frac{2 \cdot 5 \cdot \frac{L}{W}}{\left(211\frac{\text{cm}^2}{\text{V·s}}\right)\left(800\frac{\text{aF}}{\text{cm}^2}\right)\left(\frac{10^8\mu\text{m}^2}{\text{cm}^2}\right)(5 - 0.92)^2} \approx 36\ \text{k}\Omega \cdot \frac{L}{W}$$

Microchip Ex. 1055, p. 213
Microchip v. Aptiv
IPR2024-00494

The process characteristic time constants for minimum length devices are given by

$$\tau_n = R_n C_{ox} = 12\text{k}\cdot\frac{2\ \mu\text{m}}{W}\cdot\left(800\frac{\text{aF}}{\mu\text{m}^2}\right)W(2\ \mu\text{m}) = 38\ \text{ps}$$

and

$$\tau_p = R_p C_{ox} = 3\tau_n = 114\ \text{ps}\ \blacksquare$$

This example has several practical results. The resistance of the n-channel MOSFET is three times smaller than that of the p-channel MOSFET, resulting in a factor of three differences in the time constants. This is because the mobility of the electrons is three times greater than the mobility of holes in the CN20 process. Also note that the effective resistances calculated in this example, for *VDD* = 5V, do not change, so that we can use these in the coming chapters.

### 10.1.3 Delay- and Transition Times

Before we go any further in the discussion of the digital models, let's define delay and transition times in logic circuits. Consider Fig. 10.6. The top trace represents the input to a logic gate, while the bottom trace represents the output. Note that there was no logic inversion between the input and output; however, the following definitions apply equally well to the case when there is an inversion. The input rise- and falltime are labeled $t_r$ and $t_f$, respectively. The output rise- and falltime are labeled $t_{LH}$ and $t_{HL}$, respectively. The delay-time between the 50 percent points of the input and the output are labeled $t_{PLH}$ and $t_{PHL}$ depending on whether the output is changing from a high to a low or from a low to a high. These definitions are extremely important in characterizing the time-domain characteristics of digital circuits.

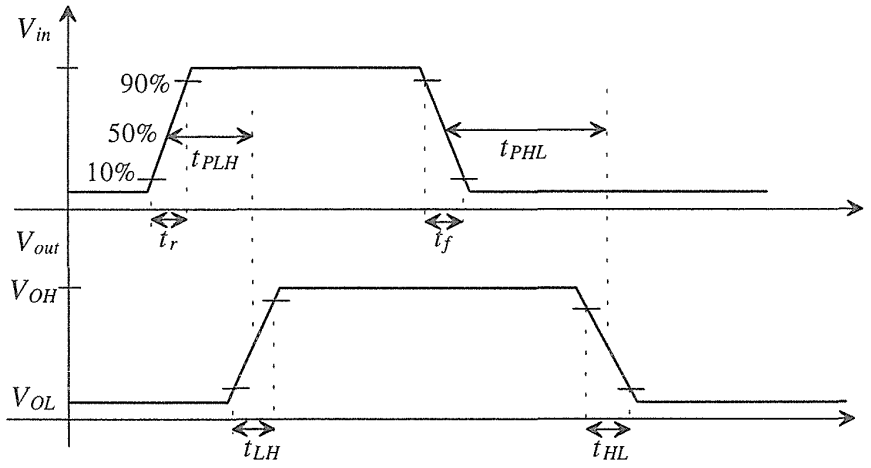



**Figure 10.6** Definition of delays and transition times.

Microchip Ex. 1055, p. 214
Microchip v. Aptiv
IPR2024-00494

For the simple *RC* circuit shown in Fig. 10.7, the delay-time is given by

$$t_{delay} = 0.7RC \tag{10.6}$$

and the rise- or falltime is given by

$$t_{rise} = 2.2RC \tag{10.7}$$

For our simple digital model of Fig. 10.5, we will assume that the propagation delay-time, whether high to low or low to high, is given by one time constant, or

$$t_{PHL},\ t_{PLH} \approx R_{n,p} \cdot C_{tot} \tag{10.8}$$

and the output rise- and falltimes are given by

$$t_{HL},\ t_{LH} \approx 2R_{n,p} \cdot C_{tot} \tag{10.9}$$

where $C_{tot}$ is the total capacitance from the drain of the MOSFET to ground and $R_{n,p}$ is the effective resistance of the n- or p-channel MOSFET, respectively. These models do not give exact results. The models are useful for determining approximate delay and transition times, usually to within a factor of two.

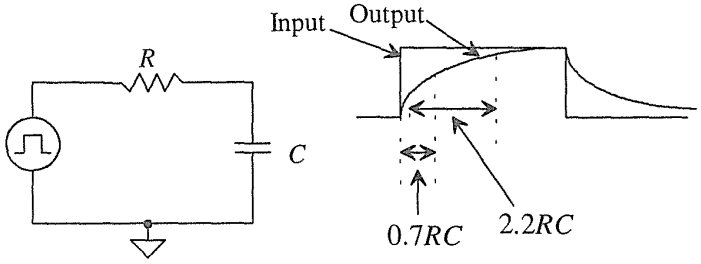



**Figure 10.7** Delay- and risetime for a simple RC circuit.

**Example 10.2**

Using hand calculations, estimate the risetime and delay-time of the following circuits (Fig. 10.8). Compare your results to SPICE simulations.

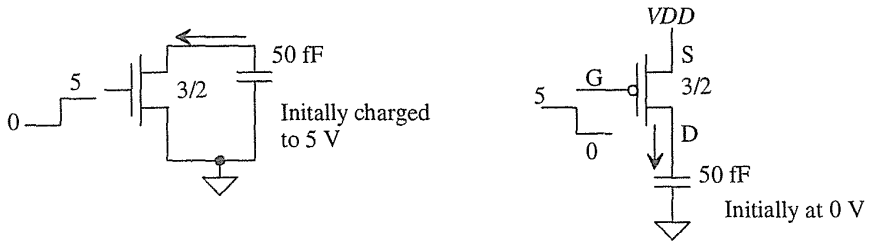



**Figure 10.8** Circuits used in Example 10.2.

Microchip Ex. 1055, p. 215
Microchip v. Aptiv
IPR2024-00494

The effective resistance of the n-channel MOSFET, from Ex. 10.1, is $R_n$ = 12k $\cdot\frac{2\mu m}{3\mu m}$ = 8 kΩ and for the p-channel $R_p$ = 24 kΩ. $C_{ox}$ is given by 2μm·3μm·800 aF/μm² = 4.8 fF. The circuit with the digital models drawn is shown in Fig. 10.9. The capacitance, $C_{ox}$, between the drain and source of the p-channel MOSFET is drawn from the drain to ground rather than from the drain to the source (*VDD*). Electrically, there is no difference in the circuits. Conceptually, it is easier to see that $C_{ox}$ is in parallel with the 50 fF capacitor when the circuit is drawn in this way.

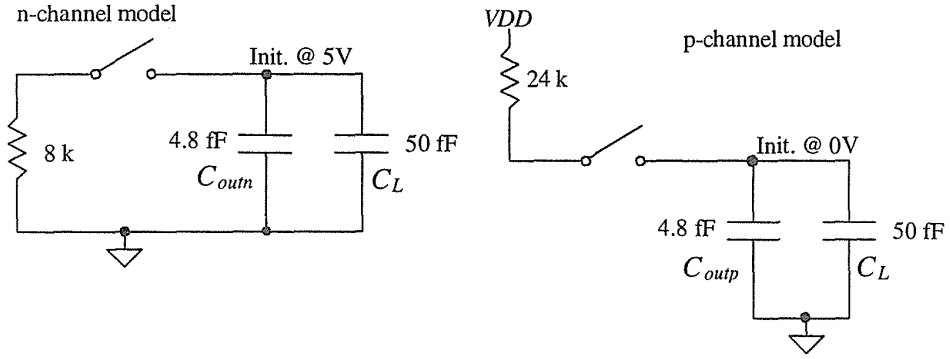



**Figure 10.9** Models used to determine switching times in Example 10.2.

The hand calculations of the delay-time for the n-channel transistor, $t_{PHL}$, is 438 ps, while the falltime, $t_{HL}$, is 877 ps. For the p-channel $t_{PLH}$ = 1.3 ns and $t_{LH}$ = 2.6 ns. SPICE simulation results are shown in Fig. 10.10. The PSPICE netlist is shown below. Notice that the .OPTIONS (or .OPTION) statement was used to help with convergence problems (see Ch. 6). ■

```
*** Top Level Netlist ***
C1      1 0 50f  IC=5
C2      2 0 50f  IC=0
M1      1 3 0 0 CMOSNB  L=2u W=3u
M2      2 4 5 5 CMOSPB  L=2u W=3u
V1      5 0     DC 5 AC 0 0
V2      3 0     DC 0 AC 0 0 PULSE(0 5 1n 1p)
V3      4 0     DC 0 AC 0 0 PULSE(5 0 1n 1p)

.MODEL CMOSNB NMOS LEVEL=4
.... BSIM model parameters of Appendix A
.MODEL CMOSPB PMOS LEVEL=4
.... BSIM model parameters of Appendix A

.OPTION ABSTOL=1u ITL4=100 RELTOL=0.01 VNTOL=.1mv
.probe
.tran 1n 5n 0 .01n uic
.end
```

Microchip Ex. 1055, p. 216
Microchip v. Aptiv
IPR2024-00494

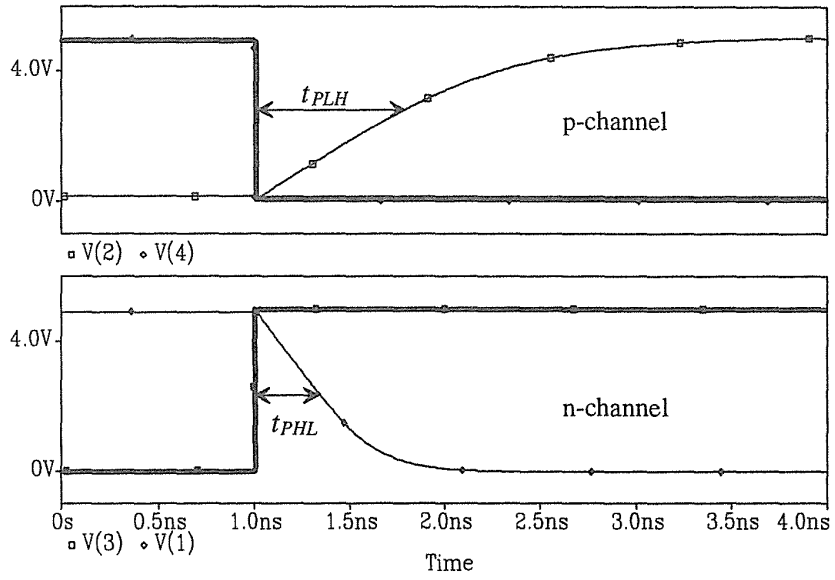



**Figure 10.10** Simulation results for Example 10.2.

## 10.2 Series Connection of MOSFETs

Consider the series connection of MOSFETs shown in Fig. 10.11. The input to this circuit, *I*, is passed to the output *Z* when *A* = *B* = *C* = *VDD* = logic "1." If *A*, *B*, or *C* is at ground (= logic "0"), the output is in the high-impedance state, that is, not a logic 0 or 1. Series connection of MOSFETs occurs frequently in CMOS digital circuit design. In this section, we will analyze the DC and transient behavior of a string of MOSFETs.

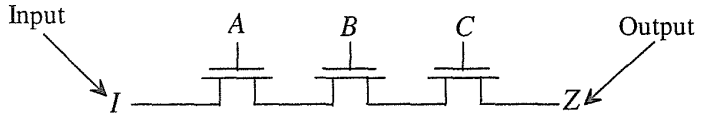



**Figure 10.11** Series connection of MOSFETs.

### 10.2.1 DC Behavior of Series-Connected MOSFETs

To illustrate the DC operation of series-connected MOSFETs, let's use Fig. 10.11 and assume that *I* = *A* = *B* = *C* = *VDD* (see Fig. 10.12a). The maximum voltage we can pass from M1 to M2 is *VDD* - $V_{THN}$ (with body effect). In fact, this is the largest voltage we can pass to the output *Z* without turning any of the MOSFETs off. Now consider Fig. 10.12b where the input is now a logic low (= 0 V). The output Z can swing all the way down to zero. In other words, the n-channel string passes 0V well and *VDD* with a threshold drop.

Microchip Ex. 1055, p. 217
Microchip v. Aptiv
IPR2024-00494

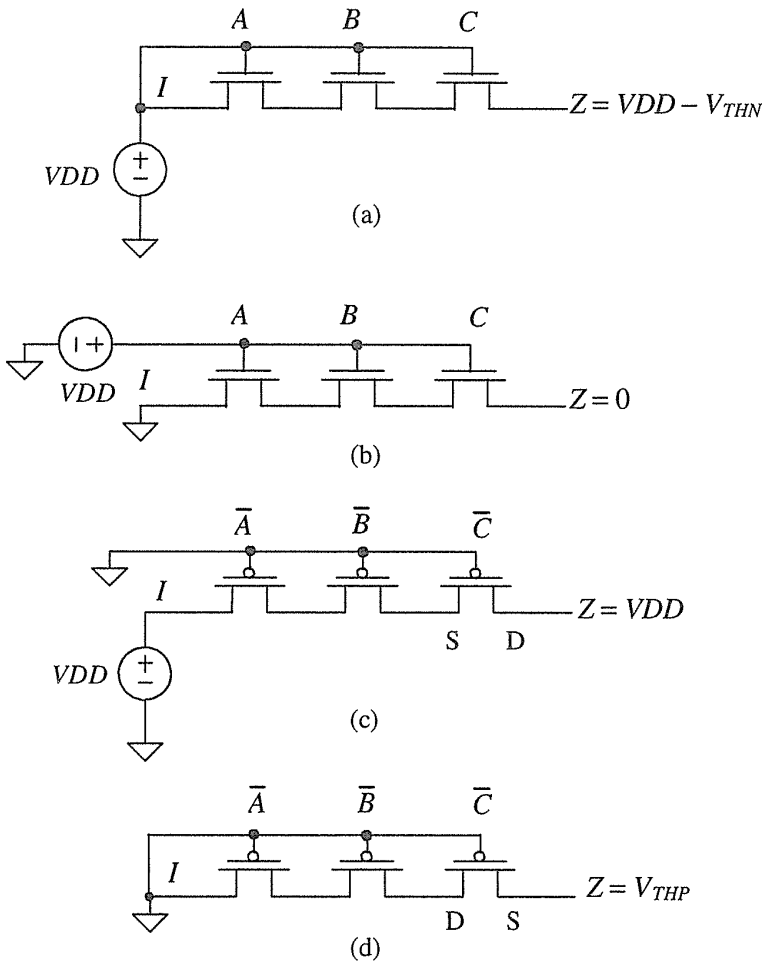



**Figure 10.12** DC operation of series-connected MOSFETs.

A p-channel series connection of MOSFETs is shown in Figs. 10.12c and d. Notice that $A$, $B$, and $C$ are now active low; that is, the p-channel MOSFETs turn on when $\overline{A} = \overline{B} = \overline{C} = 0$. The p-channel string can pass a logic high without a voltage drop. However, the minimum voltage through a p-channel string is $V_{THP}$ (with body effect).

**Example 10.3**

Describe the logic function of the circuit shown in Fig. 10.13. What are the minimum and maximum voltages at the output of the circuit?

The logical output of the circuit, $Z$, is the input $I$ (that is, $I$ is passed to the output) when $A = B = 1$ and $\overline{C} = 0$. Otherwise the output is in the

Microchip Ex. 1055, p. 218
Microchip v. Aptiv
IPR2024-00494

high-impedance state. (At least one of the switches is off.) The output variable $Z$ can vary in voltage from $V_{THP}$ to $VDD$ - $V_{THN}$. For this reason, this circuit is of little practical value. ■

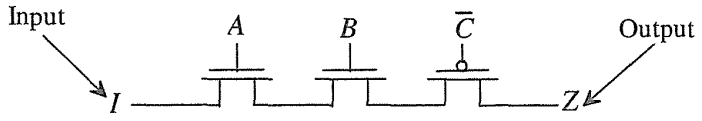



**Figure 10.13** Circuit for Example 10.3.

### 10.2.2 Delay Through Series-Connected MOSFETs

Also of importance in series-connected MOSFETs is the delay. Consider the circuit and its equivalent model shown in Fig. 10.14. In the following analysis we will assume that the capacitance at any internal node is approximated by

$$C_n = C_{inn} + C_{outn} = 2.5C_{ox} \tag{10.10}$$

For a large number of transistors, the series connection of MOSFETs behaves like an RC transmission line with a delay given by Eq. (2.11), or

$$t_d = 0.35C_nR_nl^2 \tag{10.11}$$

where $l$ is the number of MOSFETs in the series connection. Making the appropriate substitutions into this equation gives

$$t_d = 0.35 \cdot 2.5 \cdot C_{ox} \cdot R_n \cdot l^2 \approx C_{ox}R_n \cdot l^2 = \tau_n \cdot l^2 \tag{10.12}$$

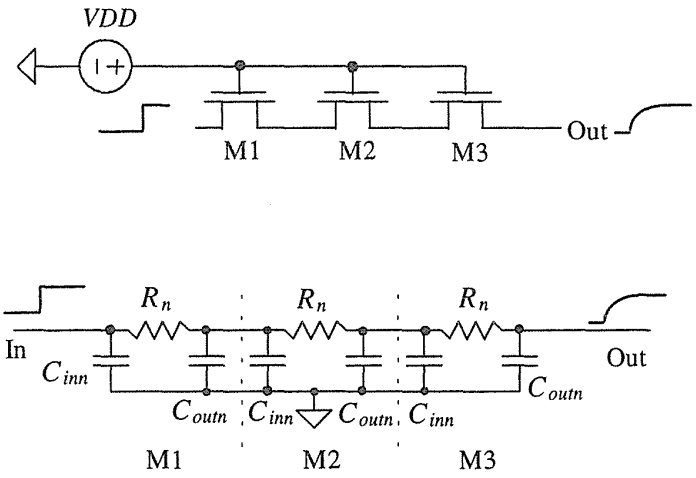



**Figure 10.14** Modeling delay through series-connected MOSFETs.

Microchip Ex. 1055, p. 219
Microchip v. Aptiv
IPR2024-00494

**Example 10.4**

Estimate and simulate the delay through ten n-channel and ten p-channel MOSFETs. Assume minimum size ($L$ = 2 μm and $W$ = 3 μm) devices. Use the CN20 parameters.

The digital model resistances of the n- and p-channel MOSFETs are

$$R_n = 12\text{k} \cdot \frac{2\ \mu\text{m}}{3\ \mu\text{m}} = 8\ \text{k}\Omega \text{ and } R_p = 36\text{k} \cdot \frac{2\ \mu\text{m}}{3\ \mu\text{m}} = 24\ \text{k}\Omega$$

The oxide capacitance of either MOSFET is $C_{ox} = C'_{ox}LW = 800\ aF \cdot 2 \cdot 3 = 4.8$ fF. Using Eq. (10.12), the delay through the series connection of ten n-channel MOSFETs is

$$t_d = C_{ox}R_n l^2 = 4.8\ \text{fF} \cdot 8\text{k} \cdot (10)^2 = 3.8\ \text{ns}$$

while the delay through ten p-channel MOSFETs is

$$t_d = C_{ox}R_p l^2 = 4.8\ \text{fF} \cdot 24\text{k} \cdot (10)^2 = 11.52\ \text{ns}$$

The simulation results are shown in Fig. 10.15. The delay of the n-channel string of MOSFETs is greatest when the string is passing a logic 1, while the delay through the p-channel string is greatest passing a logic 0. Notice that the output of the n-channel string only reaches approximately 3.5 V ($VDD - V_{THN}$), while the output of the p-channel goes down to 1.7 V ($V_{THP}$ with body effect). The delay through the n-channel (p-channel) string is less for an input going from a high to a low (low to a high). Note that the location of the delay-times in Fig. 10.15 is somewhat arbitrary. ■

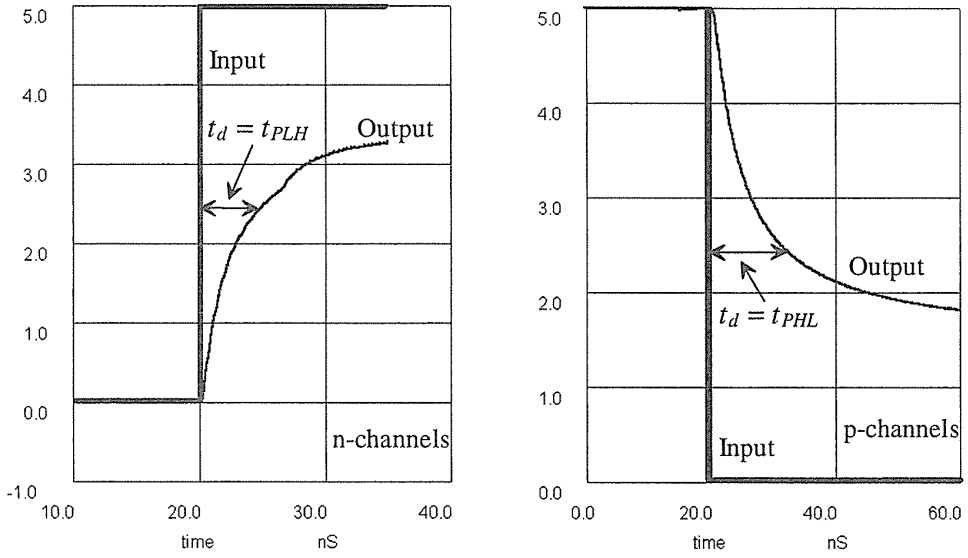



**Figure 10.15** Delay simulations for Example 10.4 (note different time scales).

Microchip Ex. 1055, p. 220
Microchip v. Aptiv
IPR2024-00494

## REFERENCE

[1] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI-Design Techniques for Analog and Digital Circuits,* McGraw-Hill Publishing Company, 1990. ISBN 0-07-023253-9.

## PROBLEMS

**10.1** Repeat Ex. 10.2 for devices with sizes 10/2 (i.e., 10 μm by 2 μm) and a capacitor of value 150 fF. Use the BSIM SPICE models.

**10.2** Calculate $R'_n$ and $R'_p$ for the CMOS14TB process. What are the process characteristic time constants?

**10.3** Simulate the operation of the circuit shown in Fig. 10.13. Pulse (transient analysis) the input from 0 to 5 V and back to 0. Show and explain the resulting circuit output. Note that the output node cannot be floating. Connect a 100MEG (not 100M) resistor from the output node to ground.

**10.4** Estimate the delay through seven n-channel MOSFETs connected in series, similar to Fig. 10.11. The size of the MOSFETs is 10/2.

**10.5** Repeat Problem 4 using p-channel MOSFETs.

**10.6** Use SPICE to verify the answer to Problem 4.

**10.7** Estimate $t_{PHL}$ and $t_{PLH}$ for the circuits shown in Fig. 10.8 when the capacitor is increased to 1 pF. Verify your hand calculations with SPICE.

**10.8** The schematic of a standard 10 to 1 scope probe is shown in Fig. P10.8a. The capacitance per foot of the scope cable is approximately 30 pF/ft. The simplified schematic of the probe is shown in Fig. P10.8b. We can use the simplified approximation of the scope probe shown in Fig. 10.8c when calculating probe-loading effects. Repeat Problem 7 using the scope loading of Fig. 10.8c. Note that measuring signals on-chip requires special probes that do not load the MOSFETs on the die. Measuring signals off-chip requires an on-chip buffer to isolate the logic from the off-chip capacitance.

**10.9** Repeat Ex. 10.2 using the CMOS14TB process with MOSFETs having a width of 0.9 μm and a length of 0.6 μm.

## Possible Student Projects

This section lists some possible student projects for fabrication through the MOSIS service. Generally, two to four student projects should be implemented on one chip. MOSIS will return to the MOSIS liaison (generally, the course instructor) four copies of each chip design submitted.

In addition to the design rule checked designs, in TLC format, the student should turn in (1) one sheet of paper showing the logic level diagrams *and* pin

Microchip Ex. 1055, p. 221
Microchip v. Aptiv
IPR2024-00494

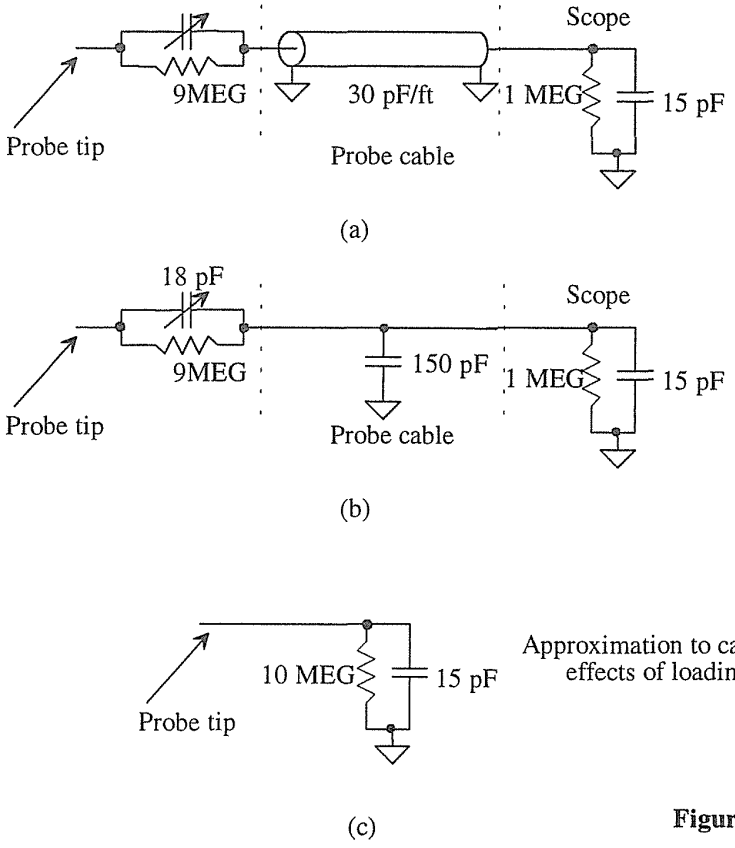



**Figure P10.8**

connections so that whoever is evaluating the chip can quickly determine functionality, and (2) final reports that consist of a block diagram, schematic diagram, layout information, hand calculations, SPICE simulations, and clear explanations of the operation of the circuit.

The report (two copies) is due with the TLC design file on a floppy (the name of the TLC cell to be fabricated should be printed clearly on the floppy disk label). Each group should hand in one disk, while each student should turn in his or her own report. Each designer is solely responsible for his or her designs; it is not a group effort. However, it is the responsibility of each designer to ensure that no fatal problems exist on the chip (e.g., *VDD* shorted to *GND*). Therefore, each student should review the design of the other student's project for his or her own benefit.

1. Quad 2-input MUX
2. Clock doubling circuit using exclusive OR gate
3. Octal buffer with tristate outputs

Microchip Ex. 1055, p. 222
Microchip v. Aptiv
IPR2024-00494

4. SR flipflop with tristate outputs
5. Edge triggered T flipflop
6. Edge triggered D flipflop
7. Schmitt trigger
8. 1 of 4 decoder
9. 4-bit dynamic shift register
10. Bi-CMOS OR gate
11. Bi-CMOS AND gate
12. 2-bit adder with carryout
13. Current-starved VCO with center frequency of 20 MHz
14. 2-bit bidirectional transceiver
15. PE gate to implement $X = \overline{A + BCD + EF}$
16. One-shot whose output pulse width is determined by external RC
17. An NMOS super buffer for driving a 20 pF load
18. An NMOS output driver for driving a 20 pF load

Advanced projects

19. A 64-bit static RAM which will include storage cell, addressing and decoding circuitry, buffers, write/read enable, and chip select.
20. Design of a charge pump (voltage generator). The input to the charge pump is *VDD* (= 5 V) and the output is –3 V. The circuit should be fully simulated. The reference, oscillator, and feedback should be fully simulated and discussed in the final report.
21. Design of a 64-bit DRAM, which will include storage cell, addressing and decoding circuitry, buffers, write/read enable, and chip select.
22. A DPLL which will take a 1 MHz input and generate a 4 MHz output. The output should follow the input for frequency changes from 900 kHz to 1.1 MHz. The report should discuss the transient properties of the DPLL, as well as a detailed design of the phase detector, VCO, and loop filter. The entire design should be monolithic; that is, no external components should be used.

Microchip Ex. 1055, p. 223
Microchip v. Aptiv
IPR2024-00494

Part

II

# CMOS Digital Circuits

Microchip Ex. 1055, p. 224
Microchip v. Aptiv
IPR2024-00494

Chapter

11

# The Inverter

The CMOS inverter is a basic building block for digital circuit design. As Fig. 11.1 shows, the inverter performs the logic operation of $A$ to $\overline{A}$. When the input to the inverter is connected to ground, the output, in accord with the digital models in the last chapter, is pulled to 5 V through the p-channel transistor. When the input terminal is connected to *VDD*, the output is pulled to ground through the n-channel MOSFET. The CMOS inverter has several important characteristics that are addressed in this chapter: Its output voltage swings from *VDD* to ground unlike other logic families that never quite reach the supply levels; the static power dissipation of the CMOS inverter is practically zero; the inverter can be sized to give equal sourcing and sinking capabilities; and the logic switching threshold can be set by changing the size of the device.

This chapter concentrates on the DC switching characteristics of the inverter and the transition times associated with driving capacitive loads and RC transmission lines, but it also addresses other types of inverters available in the CMOS process.

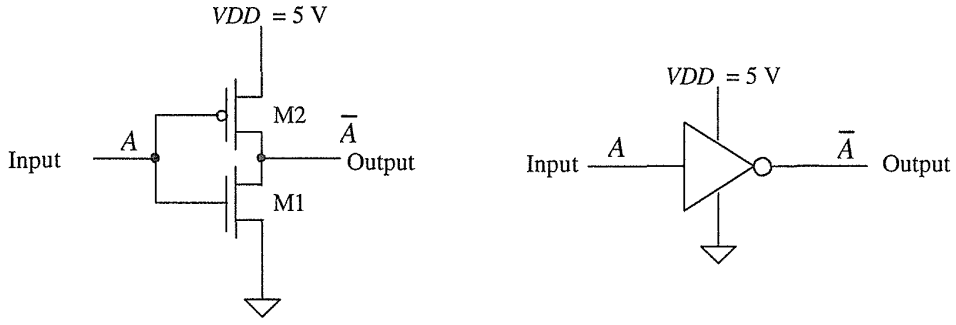



**Figure 11.1** The CMOS inverter, schematic, and logic symbol.

Microchip Ex. 1055, p. 225
Microchip v. Aptiv
IPR2024-00494

## 11.1 DC Characteristics

Consider the inverter shown in Fig. 11.2 and the associated transfer characteristic plot. In region 1 of the transfer characteristics, the input voltage is sufficiently low (typically not much greater than the threshold voltage of M1), so that M1 is off and M2 is on ($V_{SG} >> V_{THP}$). As $V_{in}$ is increased, both M2 and M1 turn on (region 2). Increasing $V_{in}$ further causes M2 to turn off and M1 to turn on fully, as shown in region 3.

The maximum output "high" voltage is labeled $V_{OH}$ and the minimum output "low" voltage, $V_{OL}$. Points A and B on this curve are defined by the slope of the transfer curves equaling –1. Input voltages less than or equal to the voltage $V_{IL}$, defined by point A, are considered a logic low on the input of the inverter. Input voltages greater than or equal to the voltage $V_{IH}$, defined by point B, are considered a logic high on the input of the inverter. Input voltages between $V_{IL}$ and $V_{IH}$ do not define a valid logic voltage level. Ideally, the difference in $V_{IL}$ and $V_{IH}$ is zero; however, this is never the case in real logic circuits.

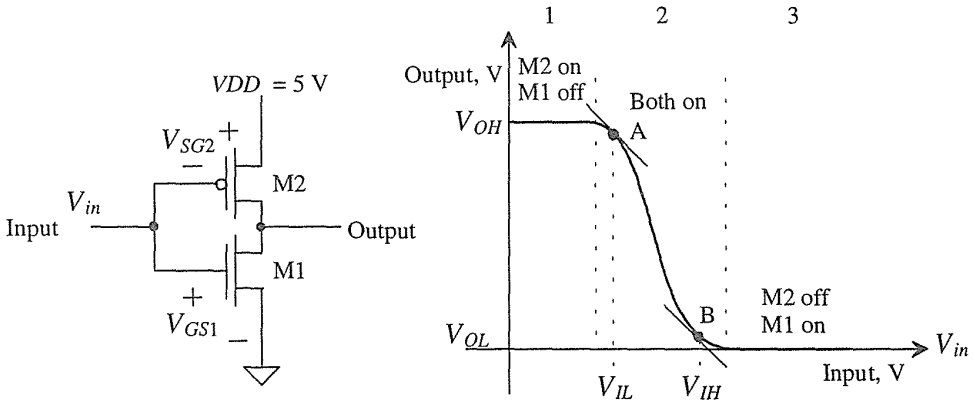



**Figure 11.2** The CMOS inverter transfer characteristics.

**Example 11.1**

Using SPICE, plot the transfer characteristics for the following inverter. From the plot, determine $V_{IH}$, $V_{IL}$, $V_{OH}$, and $V_{OL}$.

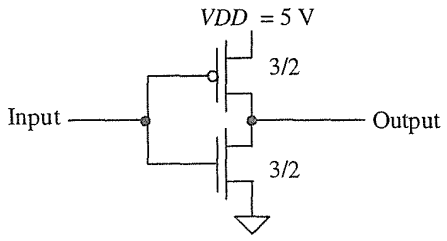



**Figure Ex11.1**

Microchip Ex. 1055, p. 226
Microchip v. Aptiv
IPR2024-00494

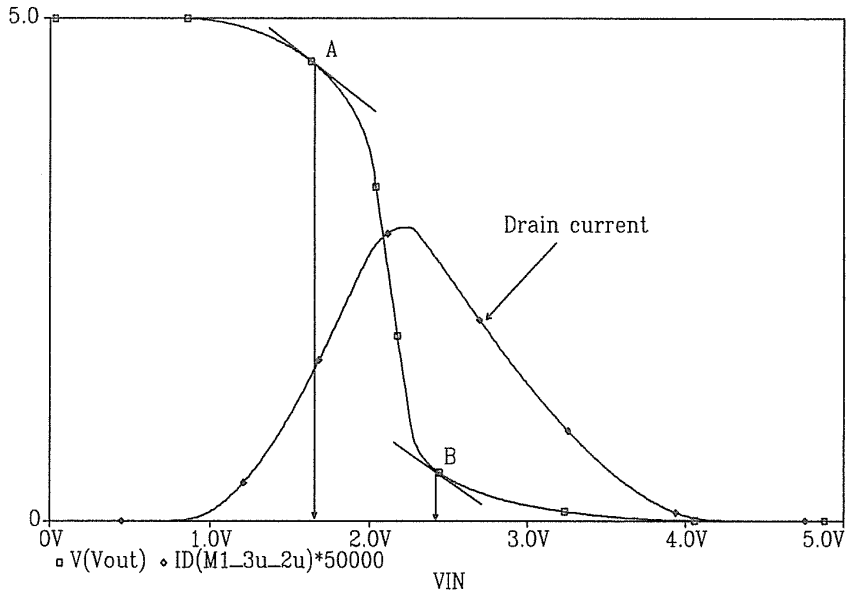



**Figure 11.3** Transfer characteristics of a minimum-size inverter used in Example 11.1.

The results of the PSPICE simulation are shown in Fig. 11.3. The netlist for the simulation is shown below. The plot shows that at point A, $V_{IL}$ is approximately 1.7 V and at point B, $V_{IH}$ is approximately 2.4 V. The output voltages, $V_{OH}$ and $V_{OL}$, are 5 V and 0 V, respectively. Figure 11.3 also shows the scaled drain current of the CMOS inverter. Notice that current flows only when the MOSFETs are switching. ■

```
*** Top Level Netlist ***
M1_3u_2u  Vout 2 0 0 CMOSNB  L=2u W=3u AD=42p AS=42p PD=26u PS=26u
M2_3u_2u  Vout 2 Vdd Vdd CMOSPB  L=2u W=3u AD=42p AS=42p PD=26u PS=26u
VDD     Vdd 0  DC 5
VIN     2 0    DC 0

.MODEL CMOSNB NMOS LEVEL=4
+vfb=-9.73820E-01, lvfb=3.67458E-01,wvfb=-4.72340E-02
... see Appendix A for complete listing of BSIM model parameters
.MODEL CMOSPB PMOS LEVEL=4
+ vfb=-2.65334E-01, lvfb=6.50066E-02, wvfb=1.48093E-01
... see Appendix A for complete listing of BSIM model parameters
.probe
.DC Vin 0 5 .01
.end
```

### 11.1.1 Noise Margins

The noise margins of a digital gate or circuit indicate how well the gate will perform under noisy conditions. The noise margin for the high logic levels is given by

Microchip Ex. 1055, p. 227
Microchip v. Aptiv
IPR2024-00494

$$NM_H = V_{OH} - V_{IH} \tag{11.1}$$

and the noise margin for the low logic levels is given by

$$NM_L = V_{IL} - V_{OL} \tag{11.2}$$

For $VDD$ = 5 V the ideal noise margins are 2.5 V; that is, $NM_L = NM_H = VDD/2$.

**Example 11.2**

For the minimum-size inverter in Ex. 11.1 determine the noise margins. Comment on making the noise margins closer to ideal.

Using Eqs. (11.1) and (11.2), $NM_H = 5 - 2.4 = 2.6$ and $NM_L = 1.7 - 0 = 1.7$. The high noise margin is almost a whole volt greater than the lower noise margin. This is mainly because the inverter switching point, $V_{SP}$, is approximately 2.2 V instead of the ideal case of 2.5 V or $VDD/2$. This is discussed further in the next section. ■

### 11.1.2 Inverter Switching Point

Consider the transfer characteristics of the basic inverter as shown in Fig. 11.4. Point C corresponds to the point on the curve when the input voltage is equal to the output voltage. At this point, the input (or output) voltage is called the inverter switching point voltage, $V_{SP}$, and both MOSFETs in the inverter are in the saturation region. Since the drain current in each MOSFET must be equal, the following is true:

$$\frac{\beta_n}{2}(V_{SP} - V_{THN})^2 = \frac{\beta_p}{2}(VDD - V_{SP} - V_{THP})^2 \tag{11.3}$$

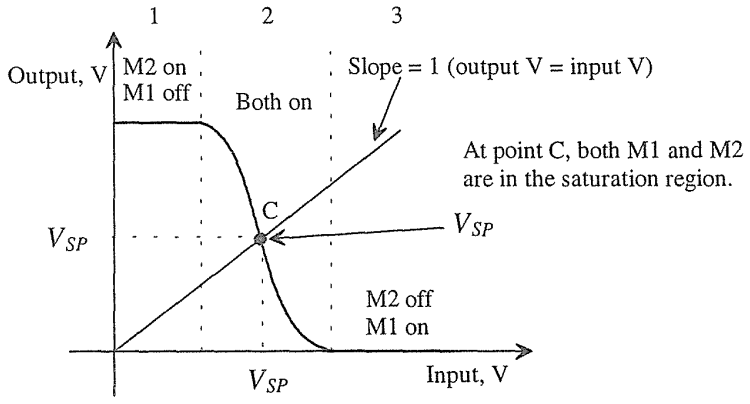



**Figure 11.4** Transfer characteristics of the inverter showing the switching point.

Solving for $V_{SP}$ gives

Microchip Ex. 1055, p. 228
Microchip v. Aptiv
IPR2024-00494

$$V_{SP} = \frac{\sqrt{\frac{\beta_n}{\beta_p}} \cdot V_{THN} + (VDD - V_{THP})}{1 + \sqrt{\frac{\beta_n}{\beta_p}}} \tag{11.4}$$

**Example 11.3**

Estimate $\beta_n$ and $\beta_p$ so that the switching point voltage of a CMOS inverter is 2.5 V. Assume CN20 parameters and $VDD$ = 5 V.

Solving Eq. (11.4) with $V_{SP}$ = 2.5 V for the ratio $\beta_n/\beta_p$ gives a value of approximately unity. That is,

$$\beta_n = \beta_p = KP_n \frac{W_1}{L_1} = KP_p \frac{W_2}{L_2}$$

Since $KP_n = 3KP_p$, the width of the p-channel transistor must be three times the width of the n-channel, assuming equal-length MOSFETs. For $V_{SP}$ = 2.5 V, this requires

$$W_2 = 3W_1$$

which is also the requirement for making $R_n = R_p$. ■

**Example 11.4**

Show, using SPICE, transfer curves for the CMOS inverter with transconductance ratios $\beta_n/\beta_p$ of 3, 1, and 1/3. Explain what changing the inverter ratio does to the transfer characteristics.

Assuming a channel length of 2 μm for the ratio of 3 $W_1$ = $W_2$ = 3 μm (one solution), for the ratio of 1, set $W_1$ = 3 μm and $W_2$ = 9 μm; for the ratio of 1/3, $W_1$ = 3 μm and $W_2$ = 27 μm works. Using a simulation similar to that in Ex. 11.1 gives the curves shown in Fig. 11.5. Notice that increasing the transconductance ratio causes $V_{SP}$ to move toward $V_{THN}$. Inverters are often sized for a specific switching point voltage in digital CMOS circuit design. ■

## 11.2 Switching Characteristics

The switching behavior of the inverter can be generalized by examining the parasitic capacitances and resistances associated with the inverter. Consider the inverter shown in Fig. 11.6 with its equivalent digital model. Although the model is shown with both switches open, in practice one of the switches is closed, keeping the output connected to $VDD$ or ground. Notice that the effective input capacitance of the inverter is

$$C_{in} = \frac{3}{2}(C_{ox1} + C_{ox2}) = C_{inn} + C_{inp} \tag{11.5}$$

The effective output capacitance of the inverter is simply

$$C_{out} = C_{ox1} + C_{ox2} = C_{outn} + C_{outp} \tag{11.6}$$

Microchip Ex. 1055, p. 229
Microchip v. Aptiv
IPR2024-00494

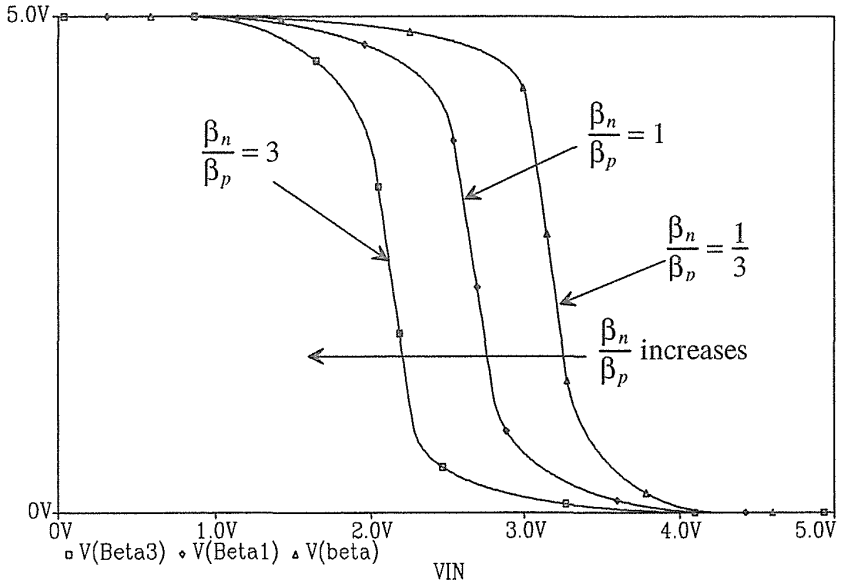



**Figure 11.5** Sizing of the CMOS inverter.

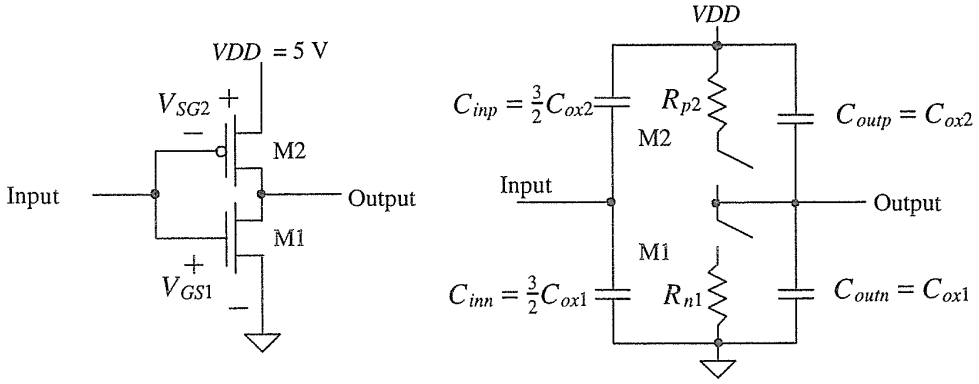



**Figure 11.6** The CMOS inverter switching characteristics using the digital model.

Microchip Ex. 1055, p. 230
Microchip v. Aptiv
IPR2024-00494

The intrinsic propagation delays of the inverter are

$$t_{PLH} = R_{p2} \cdot C_{out} \tag{11.7}$$

$$t_{PHL} = R_{n1} \cdot C_{out} \tag{11.8}$$

**Example 11.5**

Estimate and simulate the intrinsic propagation delays of the minimum-size inverter.

For the minimum-size inverter $C_{ox1} = C_{ox2} = 3\ \mu m \cdot 2\ \mu m \cdot 800\ aF/\mu m^2 = 4.8\ fF$. $R_{n1} = 12k \cdot 2\ \mu m/3\ \mu m = 8\ k\Omega$, while $R_{p2} = 24\ k\Omega$. The propagation delay times $t_{PHL} = 77$ ps and $t_{PLH} = 230$ ps. The simulation results are shown in Fig. 11.7. ■

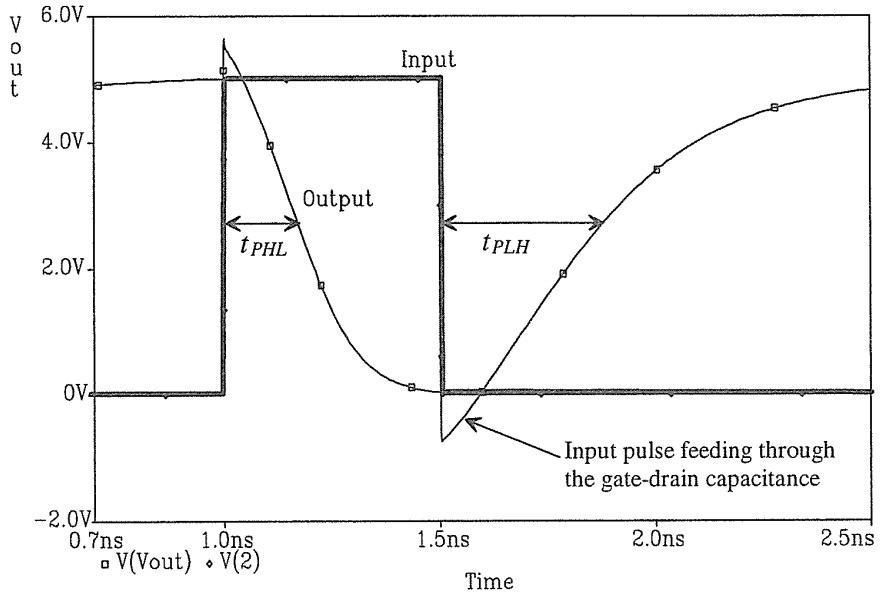



**Figure 11.7** Intrinsic inverter delay.

The propagation delays for an inverter driving a capacitive load are

$$t_{PLH} = R_{p2} \cdot C_{tot} = R_{p2} \cdot (C_{out} + C_{load}) \tag{11.9}$$

and

$$t_{PHL} = R_{n1} \cdot C_{tot} = R_{n1} \cdot (C_{out} + C_{load}) \tag{11.10}$$

where $C_{tot}$ is the total capacitance on the output of the inverter, that is, the sum of the output capacitance of the inverter, any capacitance of interconnecting lines, and the input capacitance of the following gate(s).

Microchip Ex. 1055, p. 231
Microchip v. Aptiv
IPR2024-00494

**Example 11.6**

Estimate and simulate the propagation delay of a minimum-size inverter driving a 100 fF capacitor.

The schematic of the minimum-size inverter driving a 100 fF load and the logic symbol of the inverter are shown in Fig. 11.8. The sizes adjacent to the inverter correspond to the ratio of the p-channel width to the n-channel width, assuming the lengths of the MOSFETs are the same size. Usually, the lengths are the minimum size available, which for CN20 is 2 μm. The total capacitance, $C_{tot}$, on the output of the inverter is the sum of $C_{out}$, the load capacitance and any interconnecting capacitance. In this case $C_{tot}$ = 109.6 fF, assuming no interconnecting capacitance. The propagation delay times are then $t_{PHL}$ = 877 ps and $t_{PLH}$ = 2.63 ns. This can be compared to the simulation results of Fig. 11.9.

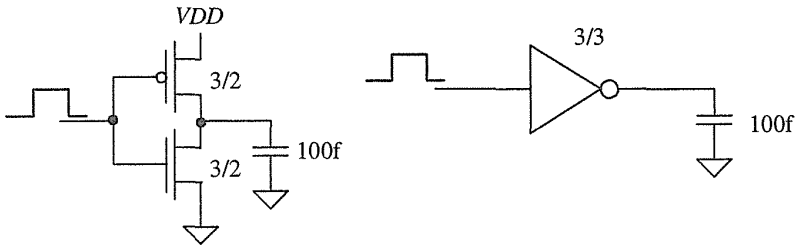



**Figure 11.8** Inverter driving a 100 fF load capacitance in Ex. 11.6.

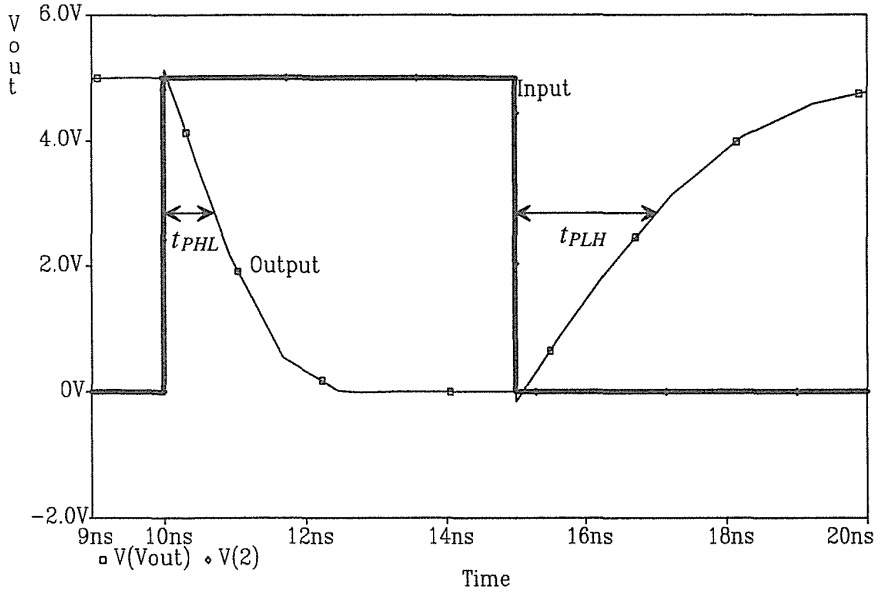



**Figure 11.9** Simulation results of minimum-size inverter driving 100 fF.

Microchip Ex. 1055, p. 232
Microchip v. Aptiv
IPR2024-00494

Notice that in the simulations, a zero risetime input results in delays somewhat less than hand calculations indicate. ■

At this point, the question "How do we size the inverter so that $t_{PHL} = t_{PLH}$?" can be answered. If $R_{n1} = R_{p2}$, the delay times are equal. This is equivalent to making $W_2 = 3W_1$, which was the same requirement used in the previous section for making $V_{SP} = VDD/2$.

### 11.2.1 The Ring Oscillator

The odd number of inverters of the circuit shown in Fig. 11.10 forms a closed loop with positive feedback and is called a ring oscillator. The oscillation frequency is given by

$$f_{osc} = \frac{1}{n \cdot (t_{PHL} + t_{PLH})} \tag{11.11}$$

assuming the inverters are identical and $n$ is the number (odd) of inverters in the ring oscillator. Since the ring oscillator is self-starting, it is often added to a test portion of a wafer to give an indication of the speed of a particular run.

Consider the case when a minimum-size inverter is used. Under these conditions, $C_{tot}$ is given by

$$C_{tot} = \overbrace{2C_{ox}}^{C_{out}} + \overbrace{3C_{ox}}^{C_{in}} = 5C_{ox} \tag{11.12}$$

where $C_{ox} = 2\ \mu m \cdot 3\ \mu m \cdot C'_{ox}$, so that

$$t_{PHL} + t_{PLH} = (R_{n1} + R_{p2})C_{tot} = (12k + 36k)\frac{2}{3} \cdot 5C_{ox} = 160k \cdot C_{ox} \tag{11.13}$$

Also consider the case when the inverters are sized to give equal propagation times. For the delays to be identical, $W_2$ must equal $3W_1$, which leads to a larger oxide capacitance for $M_2$, or

$$C_{ox2} = 3C_{ox1} \tag{11.14}$$

Therefore, $C_{tot}$ is given by

$$C_{tot} = \overbrace{4C_{ox}}^{C_{out}} + \overbrace{6C_{ox}}^{C_{in}} = 10C_{ox} \tag{11.15}$$

and the propagation delays are given by

$$t_{PHL} + t_{PLH} = \left(12k\frac{2}{3} + 36k\frac{2}{9}\right)10C_{ox} = 160k \cdot C_{ox} \tag{11.16}$$

which is the same as that given in Eq. (11.13). Although the effective resistance of the p-channel was reduced by a factor of three, the capacitance of $M_2$ was increased by a factor of three. In general, the ring oscillator frequency is dependent on $W$, although much less than one would expect. Also note that only five inverters were used. In practice, in order to keep the oscillation frequency in the tens of MHz range, the number of inverters used is 31 (for CN20).

Microchip Ex. 1055, p. 233
Microchip v. Aptiv
IPR2024-00494

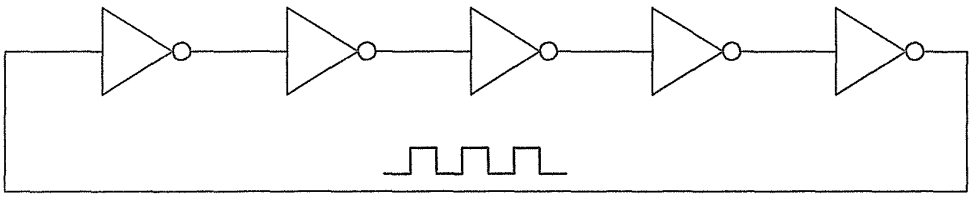


**Figure 11.10** A five-stage ring oscillator.

### 11.2.2 Dynamic Power Dissipation

Consider the CMOS inverter driving a capacitive load shown in Fig. 11.11. Each time the inverter changes states, it must either supply a charge to $C_{tot}$ or sink the charge stored on $C_{tot}$ to ground. If a square pulse is applied to the input of the inverter with a period T and frequency, $f_{clk}$, the average amount of current that the inverter must pull from *VDD*, recalling that current is being supplied from *VDD* only when the p-channel is on, is

$$I_{avg} = \frac{Q_{Ctot}}{T} = \frac{VDD \cdot C_{tot}}{T} \tag{11.17}$$

The average dynamic power dissipated by the inverter is

$$P_{avg} = VDD \cdot I_{avg} = \frac{C_{tot} \cdot VDD^2}{T} = C_{tot} \cdot VDD^2 \cdot f_{clk} \tag{11.18}$$

Notice that the power dissipation is a function of the clock frequency. A great deal of effort is put into reducing the power dissipation in CMOS circuits. One of the major advantages of dynamic logic (Ch. 15) is its lower power dissipation.

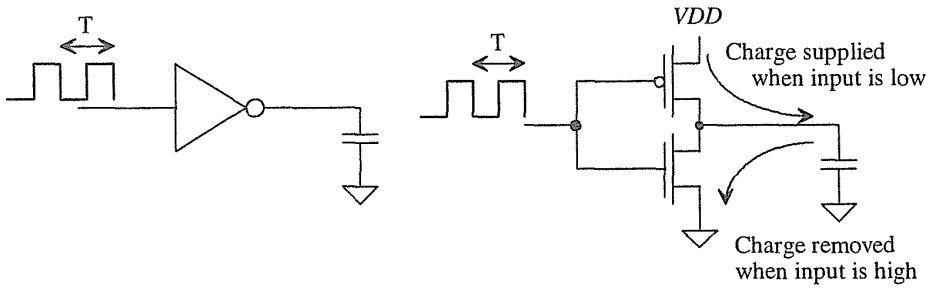



**Figure 11.11** Dynamic power dissipation of the CMOS inverter.

Microchip Ex. 1055, p. 234
Microchip v. Aptiv
IPR2024-00494

To characterize the speed of a digital process, a term called the power delay product (*PDP*) is often used. The *PDP*, measured in joules, is defined by

$$PDP = P_{avg} \cdot (t_{PHL} + t_{PLH}) \tag{11.19}$$

These terms can be determined from the ring oscillator circuit of the previous section. The *PDP* is frequently used to compare different technologies or device sizes; for example, a GaAs process can be compared with a 0.8 μm CMOS process. Although the GaAs process may have a lower propagation delay, the power dissipation may be larger and result in a larger *PDP*.

**Example 11.7**

Estimate the *PDP* of CN20, using hand analysis of a five-stage ring oscillator with $W_n = W_p = 10$ μm. Simulate the oscillator with SPICE and compare the results to the hand calculations.

The effective resistances of the n- and p-channel MOSFETs are

$$R_{n1} = 12\text{k} \cdot \frac{2\ \mu\text{m}}{10\ \mu\text{m}} = 2.4\ \text{k}\Omega$$

$$R_{p2} = 36\text{k} \cdot \frac{2\ \mu\text{m}}{10\ \mu\text{m}} = 7.2\ \text{k}\Omega$$

The input capacitance of any inverter is

$$C_{in} = C_{inn} + C_{inp} = \frac{3}{2} C'_{ox}(W_n L_n + W_p L_p) = \frac{3}{2} \cdot 800\ \text{aF} \cdot (10 \cdot 2 + 10 \cdot 2) = 48\ \text{fF}$$

the output capacitance is

$$C_{out} = C_{outn} + C_{outp} = C'_{ox}(W_n L_n + W_p L_p) = 32\ \text{fF}$$

The total capacitance on the output of any inverter is the sum of its own output capacitance and the input capacitance of the next (identical) stage. This is given by

$$C_{tot} = C_{out} + C_{in} = 80\ \text{fF}$$

thus

$$t_{PHL} + t_{PLH} = (R_{n1} + R_{p2}) C_{tot} = (2.4\text{k} + 7.2\text{k}) \cdot 80\,\text{fF} = 768\ \text{ps}$$

The oscillator frequency, from Eq. (11.11), is then

$$f_{osc} = \frac{1}{5 \cdot 768\ \text{ps}} = 260\ \text{MHz}$$

The SPICE simulation results are shown in Fig. 11.12. SPICE gives an $f_{osc}$ of approximately 300 MHz.

Normally, the *PDP* is determined with the minimum-size devices; for CN20 that would be $W_1 = W_2 = 3$ μm. However, for this example, larger than minimum-

Microchip Ex. 1055, p. 235
Microchip v. Aptiv
IPR2024-00494

size devices are specified. The average power dissipated per inverter, using Eq. (11.18), is

$$P_{avg} = 80 \text{ fF} \cdot (5)^2 \cdot (260 \text{ MHz}) = 520 \ \mu\text{W}$$

The power delay product, using hand calculations, is 400 fJ (femto-joules). SPICE simulation gives a *PDP* of 330 fJ. ■

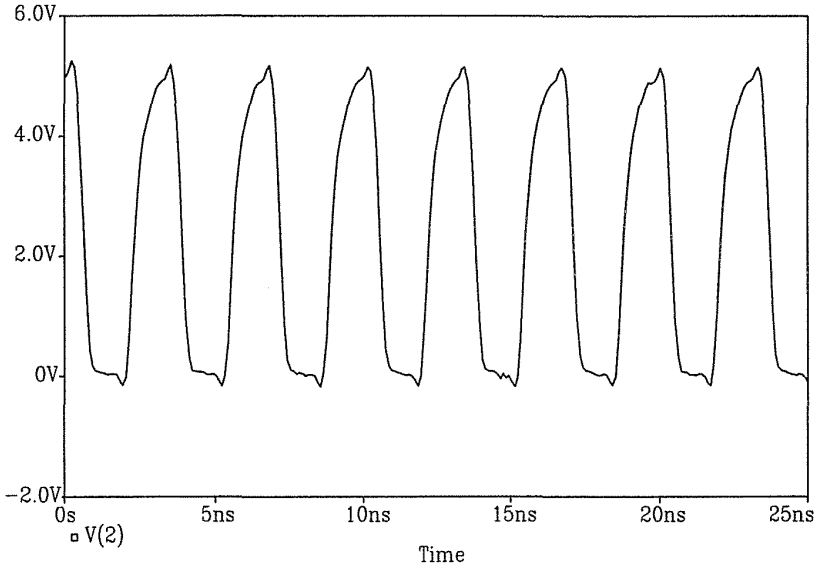



**Figure 11.12** SPICE simulation of the five-stage ring oscillator of Ex. 11.7.

## 11.3 Layout of the Inverter

If care is not taken when laying out CMOS circuits, the parasitic devices present can cause a condition known as latch-up. Once latch-up occurs, the inverter output will not change with the input; that is, the output may be stuck in a logic state. To correct this problem, the power must be removed. Latch-up is especially troubling in output driver circuits. Manufacturers of integrated circuits often use NMOS inverters (discussed later in this chapter) for output drivers, thus eliminating the possibility of latch-up.

### 11.3.1 Latch-up

Figure 11.13 illustrates two methods of laying out a minimum-size inverter. The cross-sectional view in Fig. 11.14 shows both the n-channel and the p-channel MOSFETs that make up an inverter. Notice first that in Fig. 11.7, the output pulse feeds through the gate-drain capacitance. This causes the output to change in the same direction as the input before the inverter starts to switch. This feedthrough and the parasitic bipolar transistors cause the latch-up.

Microchip Ex. 1055, p. 236
Microchip v. Aptiv
IPR2024-00494

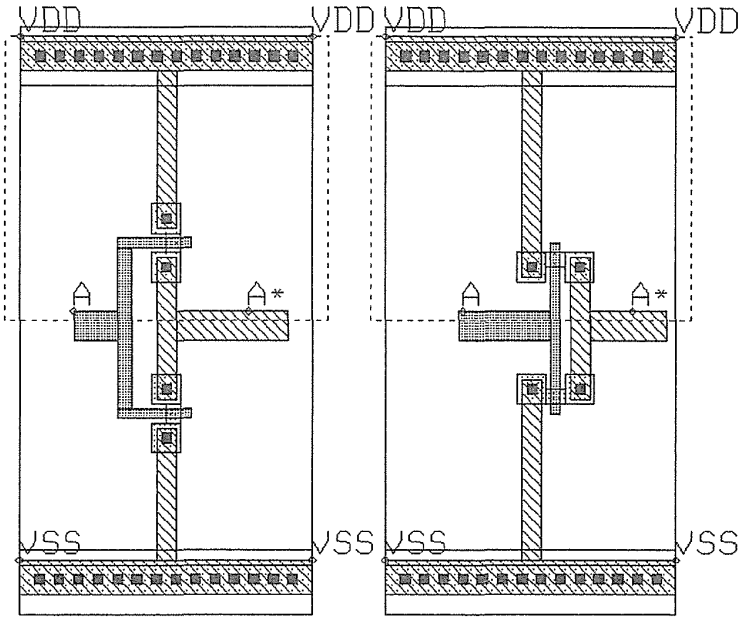



**Figure 11.13** Two inverter layout styles.

In Fig. 11.14, the emitter, base, and collector of transistor Q1 are the source of the p-channel, the n-well, and the substrate, respectively. Transistor Q2's collector, base, and emitter are the n-well, substrate, and source of the n-channel transistor. Resistors RW1 and RW2 represent the effects of the resistance of the n-well, and resistors RS1 and RS2 represent the resistance of the substrate. The capacitors C1 and C2 represent the drain implant depletion capacitance, that is, the capacitance between the drains of the transistors and the source and substrate. The parasitic circuit resulting from the inverter layout is shown in Fig. 11.15.

If the output of the inverter switches fast enough, the pulse fed through C2 (for positive going inputs) can cause the base-emitter junction of Q2 to become forward biased. This then causes the current through RW2 and RW1 to increase, causing Q1 to turn on. When Q1 is turned on, the current through RS1 and RS2 increases, causing Q2 to turn on harder. This positive feedback will eventually cause Q2 and Q1 to turn on fully and remain that way until the power is removed and reapplied. A similar argument can be made for negative-going inputs feeding through C1.

Several techniques reduce the latch-up problem. The first technique is to slow the rise- and falltimes of the logic gates, reducing the amount of signal fed through C1 and C2. Reducing the areas of M1 and M2's drains lowers the size of the depletion capacitance and the amount of signal fed through. Probably the best method of reducing latch-up effects is to reduce the parasitic resistances RW1 and RS2. If these resistances are zero, Q1 and Q2 never turn on. The value of these resistances, as seen

Microchip Ex. 1055, p. 237
Microchip v. Aptiv
IPR2024-00494

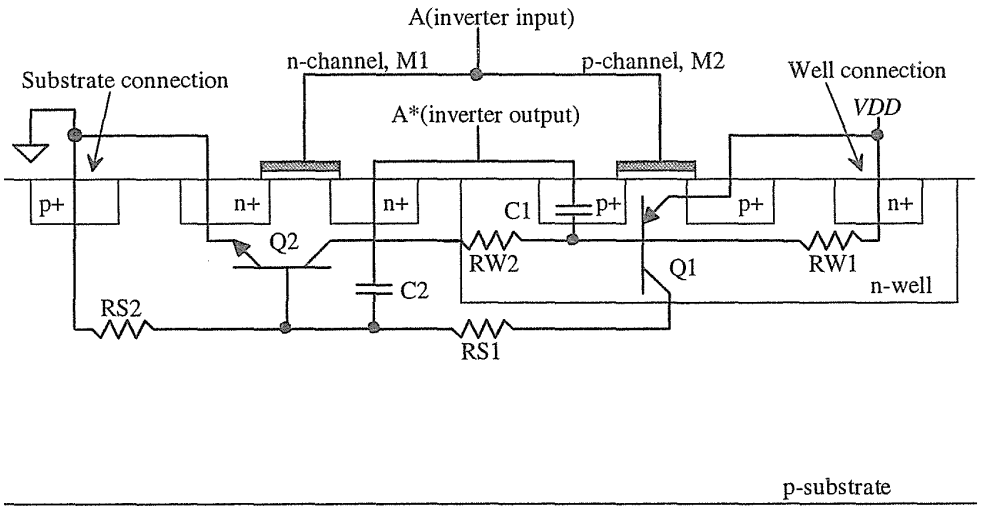



**Figure 11.14** Cross-sectional view of an inverter showing parasitic bipolar transistors and resistors.

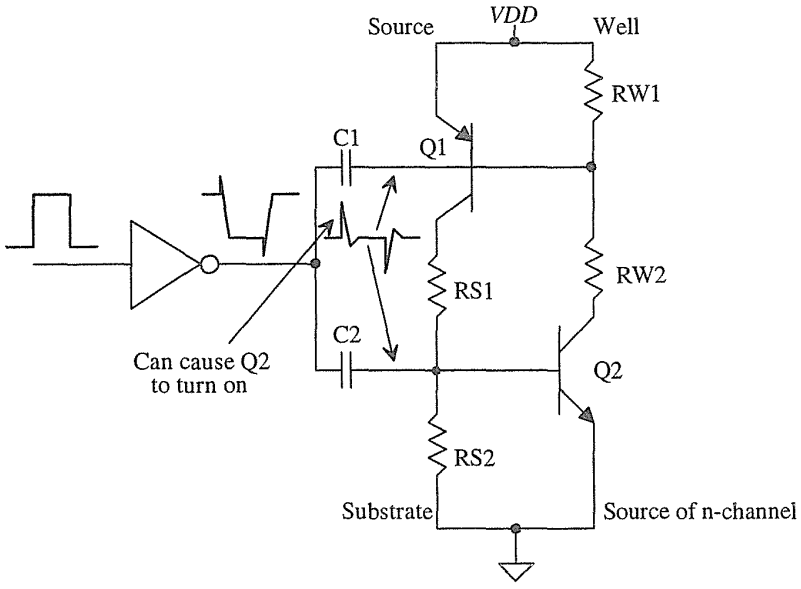



**Figure 11.15** Schematic used to describe latch-up.

Microchip Ex. 1055, p. 238
Microchip v. Aptiv
IPR2024-00494

in Fig. 11.14, is a strong function of the distance of the well and substrate contacts. Simply put, the closer these contacts are to the inverter, the fewer are the chances the inverter will latch up. These contacts should be not only close, but also plentiful. Placing substrate and well contacts between the p- and n-channel MOSFETs provides a low-resistance connection to *VDD* and ground, significantly helping to reduce latchup (see Fig. 11.16 for a simple layout example). Placing n+ and p+ areas between or around circuits reduces the amount of signal reaching a given circuit from another circuit. These diffusions are sometimes called guard rings. Notice that poly cannot be used to connect the gates of the MOSFETs, since poly over the n+ or p+ will be interpreted as a MOSFET. Therefore, metal2 is used to connect the p- and the n-channel MOSFETs together. The cost for reducing the possibility of latch-up is a more complicated layout in a larger area.

Large MOSFETs, required to drive off-chip loads, are especially susceptible to latch-up because of the large drain depletion capacitances. The only way to design latch-up free output drivers is to use only one type of MOSFET, in most cases an n-channel. Eliminating the n-well and the p-channel transistor eliminates the possibility of latch-up. This will be discussed further in the next section.

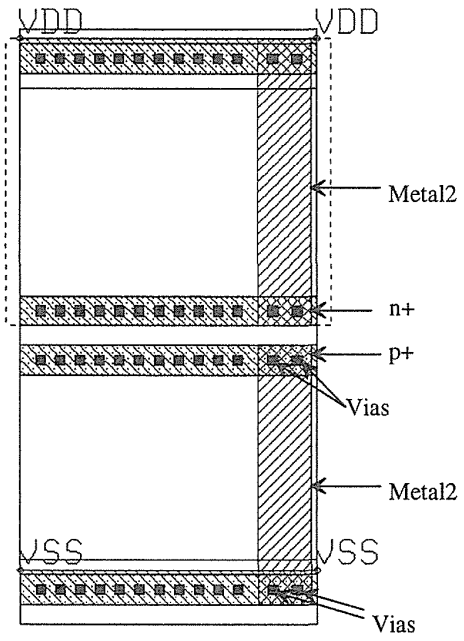



**Figure 11.16** Alternative standard-cell frame used for better latch-up protection.

Microchip Ex. 1055, p. 239
Microchip v. Aptiv
IPR2024-00494

## 11.4 Sizing for Large Capacitive Loads

Designing a circuit to drive large capacitive loads with minimum delay is important when driving off-chip loads. Consider the inverter string driving a load capacitance, labeled $C_{load}$ and shown in Fig. 11.17. If a single inverter were to drive $C_{load}$, the delay times would be

$$t_{PHL} + t_{PLH} = (R_n + R_p) \cdot (C_{out} + C_{load}) \tag{11.20}$$

If, moving toward the load, cascading $N$ inverters are used, each inverter larger than the previous by a factor $A$ (that is, the width of each MOSFET is multiplied by $A$), a minimum delay can be obtained as long as $A$ and $N$ are picked correctly. Each inverter's input capacitance is also larger than the previous inverter's input capacitance by a factor of $A$. If the load capacitance is equal to the input capacitance of the last inverter multiplied[1] by $A$, then

$$\text{Input C of final inverter} = C_{in1} \cdot A^N = C_{loaa} \tag{11.21}$$

where $C_{in1}$ is the input capacitance of the first inverter. Rearranging Eq. (11.21) gives

$$A = \left[ \frac{C_{load}}{C_{in1}} \right]^{\frac{1}{N}} \tag{11.22}$$

The total delay of the inverter string is given by

$$(t_{PHL} + t_{PLH})_{total} = \underbrace{(R_{n1} + R_{p1})(C_{out1} + AC_{in1})}_{\text{First-stage delay}} + \underbrace{\frac{(R_{n1} + R_{p1})}{A} \cdot (AC_{out1} + A^2 C_{in1})}_{\text{Second-stage delay}} \ldots \tag{11.23}$$

where $R_{n1}$ and $R_{p1}$ are the effective resistances of the first inverter and $C_{out1}$ is the output capacitance of the first inverter. As the inverters are increased in size by $A$, their capacitances, both input and output, increase by $A$ while their resistances decrease by a factor $A$. The equation (11.23) can be written as

$$(t_{PHL} + t_{PLH})_{total} = \sum_{k=1}^{N} (R_{n1} + R_{p1})(C_{out1} + AC_{in1}) = N(R_{n1} + R_{p1})(C_{out1} + AC_{in1}) \tag{11.24}$$

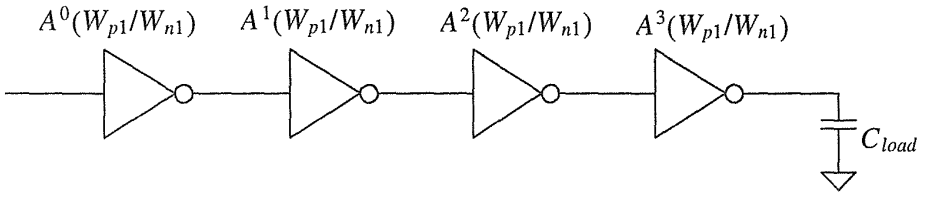



**Figure 11.17** Cascade of inverters used to drive a large load capacitance.

[1] Consider this as if the load capacitance were simulating the input capacitance of the next inverter (if there was another inverter).

Microchip Ex. 1055, p. 240
Microchip v. Aptiv
IPR2024-00494

or with the help of Eq. (11.22):

$$(t_{PHL}+t_{PLH})_{total}=N(R_{n1}+R_{p1})\left[C_{out1}+\left(\frac{C_{load}}{C_{in1}}\right)^{\frac{1}{N}}\cdot C_{in1}\right] \tag{11.25}$$

The minimum delay can be found by taking the derivative of this equation with respect to $N$, setting the result equal to zero, and solving for $N$. Taking the derivative of Eq. (11.25) with respect to $N$ gives

$$(R_{n1}+R_{p1})C_{out1}+(R_{n1}+R_{p1})C_{in1}\left[\left(\frac{C_{load}}{C_{in1}}\right)^{\frac{1}{N}}+N\cdot\left(\frac{C_{load}}{C_{in1}}\right)^{\frac{1}{N}}\frac{\ln(C_{load}/C_{in1})}{-N^2}\right]=0 \tag{11.26}$$

The first term in this equation is the intrinsic delay of the first inverter in our cascade of inverters. If we assume that this delay is small, solving this equation for $N$ gives

$$N=\ln\frac{C_{load}}{C_{in1}} \tag{11.27}$$

Eqs. (11.27) and (11.22) are used to design a cascade of inverters in order to drive a large capacitance. Note that the larger the first inverter, the fewer the number of inverters needed to drive a given capacitive load. Logic families like the 74HCXX series use fairly large MOSFETs throughout the entire chip. This allows driving large capacitances, say >50 pF, with typically two or three buffer stages. In very-large-scale integration (VLSI) design where the MOSFETs are generally close to minimum size, the number of stages can be greater than this. The following example illustrates output buffer design in its simplest form.

**Example 11.8**

Estimate $t_{PHL}$ + $t_{PLH}$ for the inverter shown in Fig. Ex11.8 driving a load capacitance of 20 pF. Design a buffer to drive the load capacitance with a minimum delay. Compare the propagation delays of both circuits using SPICE.

The total propagation delay of the unbuffered inverter is given by

$$t_{PHL}+t_{PLH}=\left(12\text{k}\frac{2}{3}+36\text{k}\frac{2}{9}\right)\cdot\left(\overbrace{2\cdot 3\cdot 800\text{ aF}+2\cdot 9\cdot 800\text{ aF}}^{C_{out1}=19.2\text{ fF}}+\overbrace{20\text{ pF}}^{C_{load}}\right)=320\text{ ns !}$$

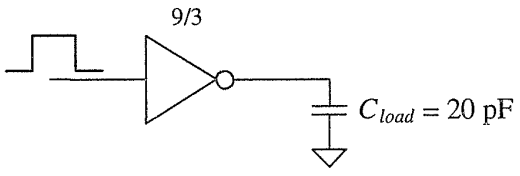



**Figure Ex11.8**

Microchip Ex. 1055, p. 241
Microchip v. Aptiv
IPR2024-00494

Designing a buffer begins with determining $C_{in1}$. For the present case $C_{in1} = \frac{3}{2}C_{out1} = 28.8$ fF. The number of inverters using Eq. (11.27) is

$$N = \ln\left(\frac{20 \text{ pF}}{28.8 \text{ fF}}\right) = 6.54 \rightarrow 7 \text{ stages}$$

In order to maintain the same logic, that is, an inversion of the input signal, we will use seven inverters. In practice, the difference in delay between six and seven inverters is negligible. If we did not want a logic inversion, we would use six stages. The area factor is then

$$A = \left[\frac{20 \text{ pF}}{28.8 \text{ fF}}\right]^{\frac{1}{7}} = 2.55$$

The total delay, using Eq. (11.25), is then

$$(t_{PHL} + t_{PLH})_{total} = 7(16\text{k})(19.2 \text{ fF} + 2.55 \cdot 28.8 \text{ fF}) = 10.4 \text{ ns}$$

or over 30 times faster. Since the p-channel width is three times that of the n-channel width, the propagation delay times, $t_{PHL}$ and $t_{PLH}$, are equal, or

$$t_{PHL} = t_{PLH} = 5.2 \text{ ns}$$

A schematic of the design is shown in Fig. 11.18. The actual sizes were changed to a number close to that given using the value of $A$ calculated above to make the layout easier. Notice that the first inverter is the same inverter shown above. The SPICE simulation results are shown in Fig. 11.19. Note that the unbuffered inverter does not fully charge the capacitor since the input of the inverter changes back to zero volts 15 ns after it changes to *VDD*. ■

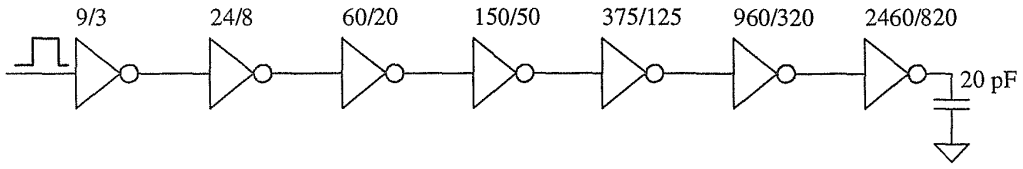



**Figure 11.18** Buffer designed in Ex. 11.8.

It should be clear that, although this technique results in the least delay in driving the 20 pF load, the MOSFETs needed are very large. In many applications, the minimum delay through a buffer is not required. A specification that the delay be less than some value is given. Consider the following example.

Microchip Ex. 1055, p. 242
Microchip v. Aptiv
IPR2024-00494

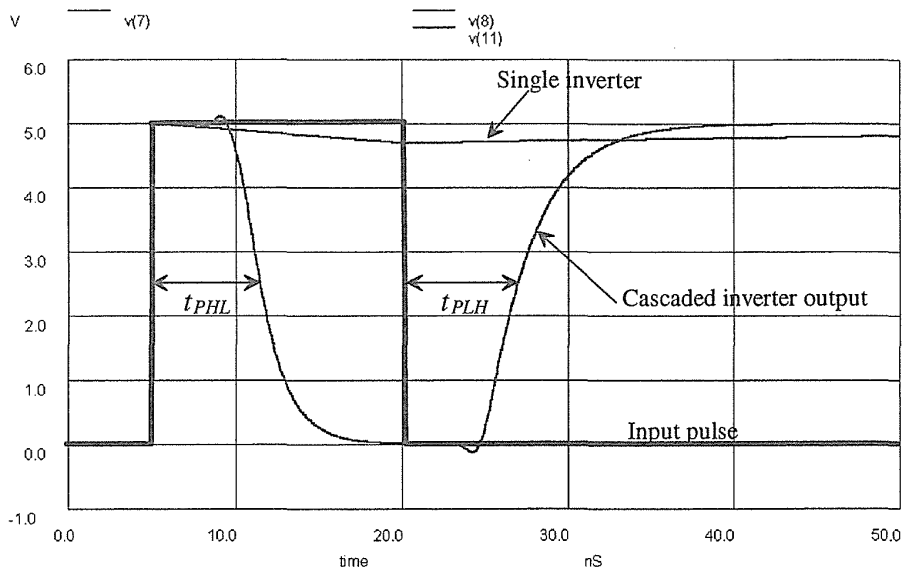



**Figure 11.19** Simulation results from Ex. 11.8.

**Example 11.9**

Redesign the buffer of Ex. 11.8 so that the delay, $t_{PHL} + t_{PLH}$, is less than 15 ns. (The minimum delay was 10.4 ns in Ex. 11.8.)

In order to maintain the logic inversion, either three or five stages should be used. Let's begin by trying three stages. The area factor for three stages is given by

$$A = \left[\frac{20 \text{ pF}}{28.8 \text{ fF}}\right]^{1/3} = 8.86$$

The delay is calculated using Eq. (11.25) and is given by

$$t_{PHL} + t_{PLH} = 3(16\text{k})(19.2 \text{ fF} + 8.86 \cdot 28.8 \text{ fF}) = 13.2 \text{ ns}$$

or

$$t_{PHL} = t_{PLH} = 6.6 \text{ ns}$$

The resulting buffer is shown in Fig. 11.20. The layout size of this buffer is significantly smaller than the buffer designed in Ex. 11.8, while the increase in delay is modest. ■

*Layout of Large MOSFETs*

The time and effort it takes to lay out the large MOSFETs used in an output buffer can be greatly reduced using cell hierarchy. As a simple example, let's lay out a 250/2

Microchip Ex. 1055, p. 243
Microchip v. Aptiv
IPR2024-00494

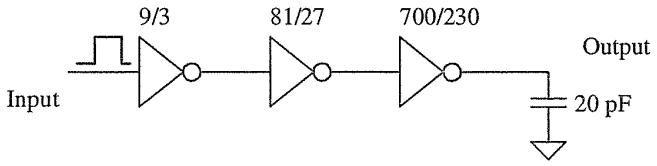



**Figure 11.20** Buffer design of Ex. 11.9.

n-channel MOSFET. We begin by creating a cell, called NAA25X2 (n-active-area 25 by 2), with a rank of 1 and shown in Fig. 11.21.

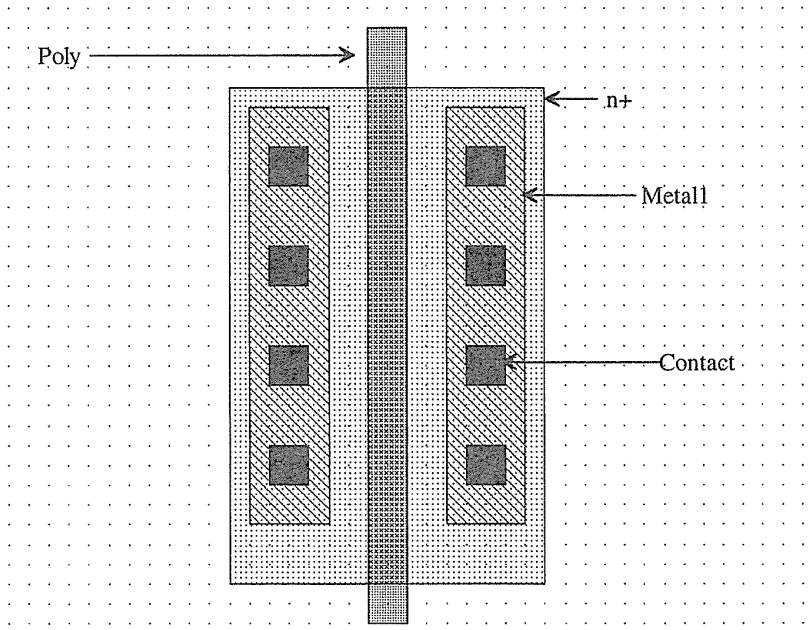



**Figure 11.21** Layout of an n-channel MOSFET measuring 25 μm (width) by 2 μm (length).

The next step is to create a cell called N250X2 with a rank of 2. We then set the object (using the **obj** command button in LASI) to the cell NAA25X2 and select the **Add** command button. Figure 11.22 shows four NAA25X2 cells added to the N250X2 cell. The "trick" when placing the NAA25X2 cells is to overlap the source/drain areas. As discussed in Ch. 5, this sharing of areas reduces the depletion capacitance to substrate of the drain/source implants. Figure 11.23a shows the layout of the 250 by 2 n-channel MOSFET with the sources, drains, and gates of each of the individual MOSFETs connected together. The only thing missing from this layout is the connection to the substrate. (Not providing well and substrate connections for the

Microchip Ex. 1055, p. 244
Microchip v. Aptiv
IPR2024-00494

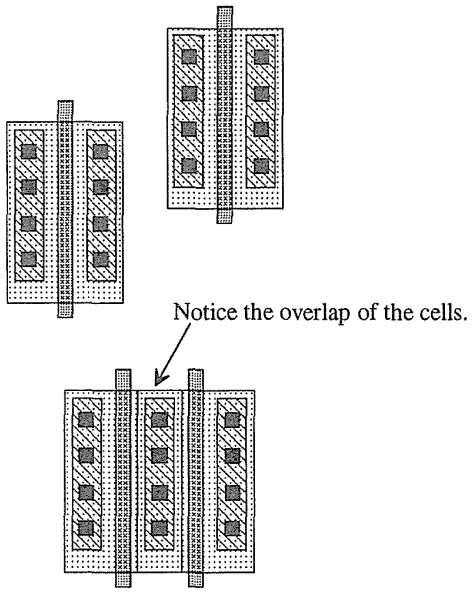



**Figure 11.22** Placing basic cells to form a large MOSFET.

MOSFETs is a *fatal* layout error.) Since the standard-cell frame, SFRAME (with a rank of 1) in the CN20 setups provided with LASI, provides these connections, we could add this frame to our basic layout. The result is shown in Fig. 11.23b.

### 11.4.1 Distributed Drivers

Consider the driver circuit shown in Fig. 11.24a containing 11 inverters. If all of the inverters shown in the figure are the same size, the delay from the input to the output is

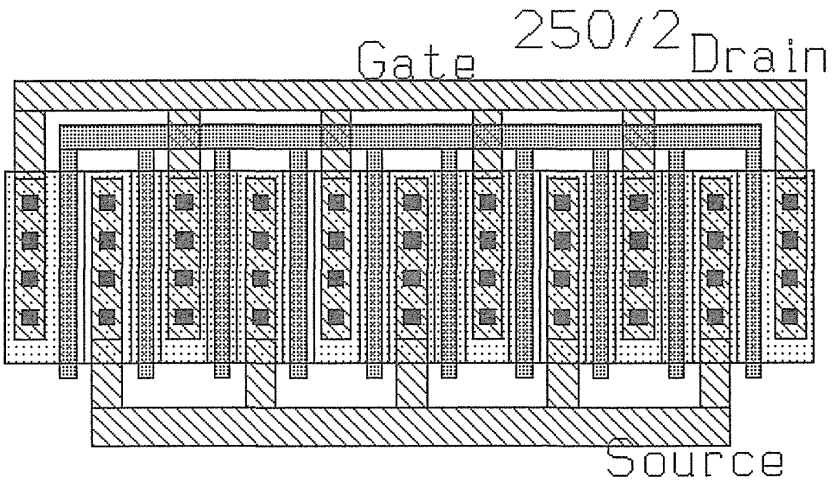
$$t_{PHL} + t_{PLH} = (R_n + R_p)(C_{out} + 10C_{in}) \tag{11.28}$$

Now consider the circuit shown in Fig. 11.24b with 13 inverters. Again, assuming all inverters are the same size, the delay from the input to the output is

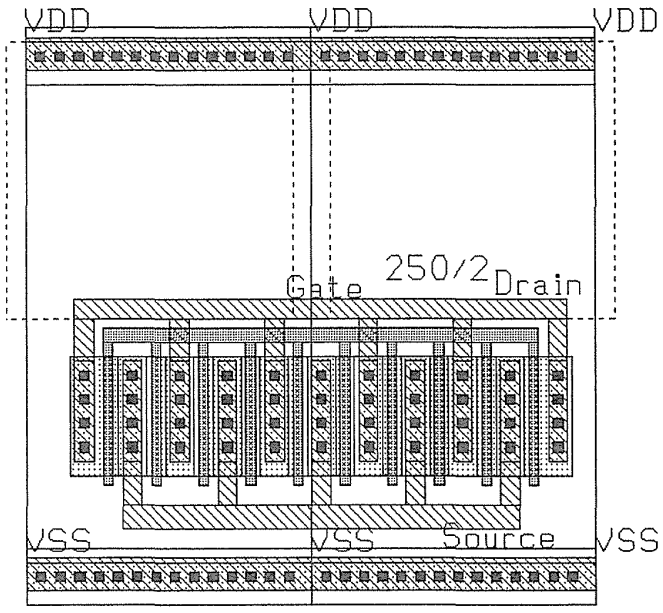
$$t_{PHL} + t_{PLH} = (R_n + R_p)[(C_{out} + 2C_{in}) + (C_{out} + 5C_{in})] = (R_n + R_p)[2C_{out} + 7C_{in}] \tag{11.29}$$

which is less delay than the circuit with 11 inverters. Often, distributing the signal into different paths can reduce the propagation delay. At this point we can ask the question, "Why not make the first inverter in the circuits of Fig. 11.24 really large so that it has small effective resistances for driving the ten inverters quickly?" The answer is simply that as we increase the size of an inverter, we also increase its input capacitance. In SPICE simulations, we use ideal voltage sources to drive the first gate in our circuit. In practice, this inverter is driven from another gate somewhere on the chip. Increasing the size will slow the propagation delay-time of the gate driving this inverter.

Microchip Ex. 1055, p. 245
Microchip v. Aptiv
IPR2024-00494




(a)




(b)

**Figure 11.23** (a) Layout of a 250 by 2 n-channel MOSFET and (b) using a standard-cell frame.

Microchip Ex. 1055, p. 246
Microchip v. Aptiv
IPR2024-00494

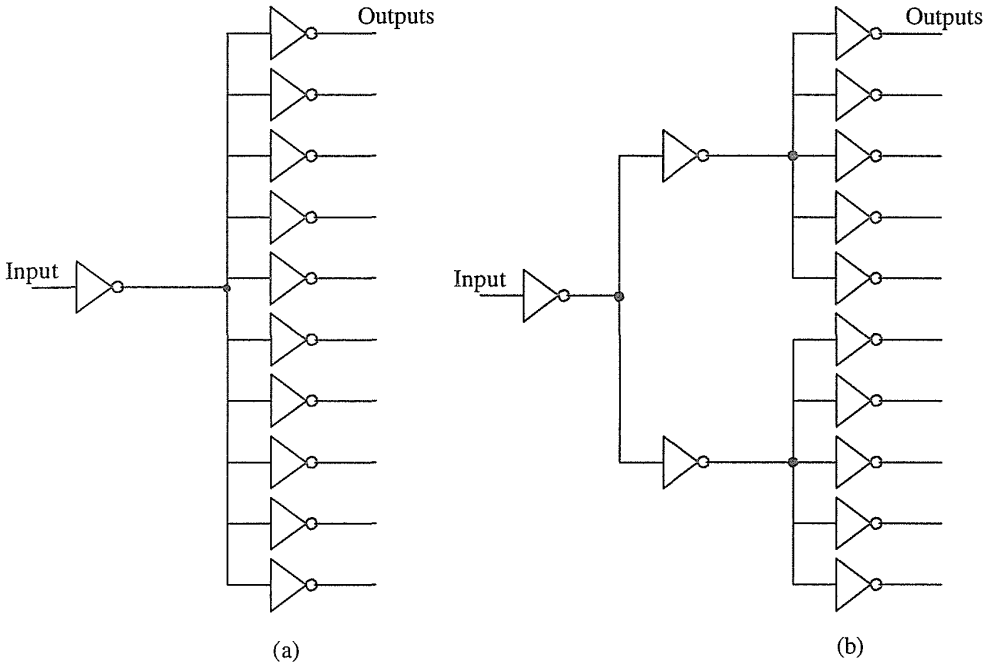



**Figure 11.24** Distributed drivers.

### 11.4.2 Driving Long Lines

Often when designing large systems, a signal may need to be driven across the chip. In some cases, for example, dynamic random-access memory (DRAM), the signal must be transmitted over a line that has a large parasitic resistance and capacitance. We need to develop a method of determining the delay through this line using hand calculations. This will lend insight to the design and help to determine exactly how to design the driver (or drivers).

Consider the driver circuit shown in Fig. 11.25. The inverter is driving an RC transmission line with resistance/unit length, $r$, capacitance/unit length, $c$, and unit length, $l$. We can estimate the delay from the input to voltage across the capacitor by adding the delays. This is given by

$$t_{PHL} + t_{PLH} = (R_n + R_p)(C_{out} + c \cdot l + C_{load}) + 0.35 \cdot rcl^2 + (r \cdot l)(C_{load}) \qquad (11.30)$$

where the first term in this equation is the delay associated with the inverter driving the total capacitance at its output to ground. The second term is the delay through the line, while the last term is an estimate of the delay associated with driving a capacitive load through the line's resistance. The most common method of reducing the delay through the line is to place buffer stages at different locations along the line. This effectively breaks the line up and can lower the overall delay. If $C_{load}$ is a major contributor to the delay, a buffer can be inserted between the $RC$ line and $C_{load}$ to reduce the delay.

Microchip Ex. 1055, p. 247
Microchip v. Aptiv
IPR2024-00494

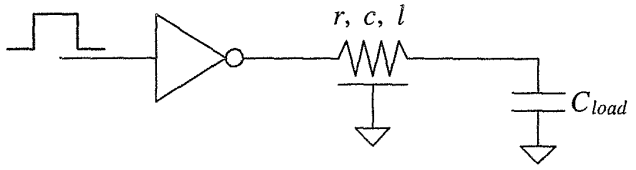



**Figure 11.25** Driving an RC transmission line.

## 11.5 Other Inverter Configurations

Three other inverter configurations are shown in Fig. 11.26. The inverter shown in Fig. 11.26a is an NMOS-only inverter, useful in avoiding latch-up. The inverters shown in Fig. 11.26b and c use a p-channel load, which is, in general, most useful in logic gates with a large number of inputs (more on this in the next chapter). In general, the selection of the MOSFET sizes follows the 4 to 1 rule; that is, the resistance ($R_n$ or $R_p$) of the load is made four times larger than the resistance of M1. The output logic low will never reach 0 V for these inverters, and thus the noise margins are poorer than the basic CMOS inverter of Fig. 11.1. Also, DC power will be dissipated when the output logic level is a low since a drain current will flow through the inverters. The output high level of the inverter of Fig. 11.26c will reach *VDD*, while the other inverter's output high level will be a threshold voltage drop below *VDD*. It might be concluded that the power dissipation of the inverters shown in Fig. 11.26 is greater than the basic CMOS inverter. However, since the input capacitance of these inverters is less than the basic CMOS inverter and the output voltage swing is reduced, the inverter with the greatest power dissipation is determined by the operating frequency. At high operating frequencies, the basic CMOS inverter dissipates the most power.

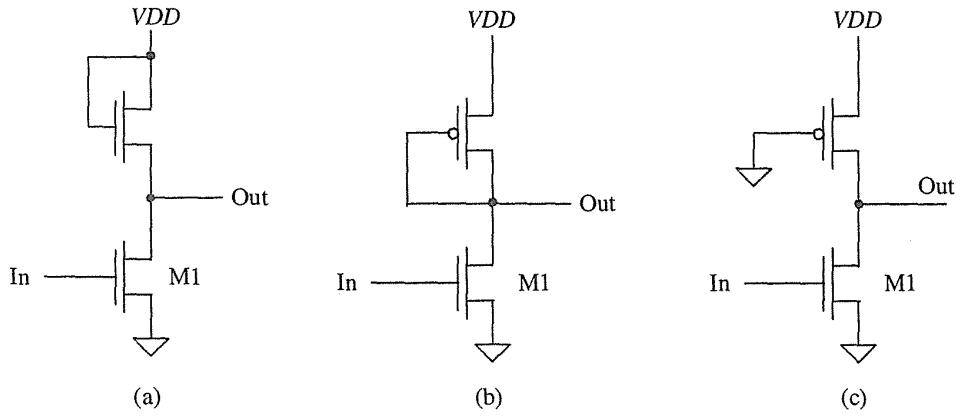



**Figure 11.26** Other inverter configurations.

Microchip Ex. 1055, p. 248
Microchip v. Aptiv
IPR2024-00494

### 11.5.1 N-Channel Only Output Drivers

Because of the susceptibility of the basic CMOS inverter to latch-up, output drivers consisting of only n-channel MOSFETs are used. Figure 11.27 shows the basic "NMOS super buffer." When the input signal is low, M1 and M4 are off while M2 and M3 are on. The output is pulled to ground through M2. A high on the input to the buffer causes M1 and M4 to turn on pulling the output to $VDD - V_{THN}$, assuming the input high-signal amplitude is *VDD*.

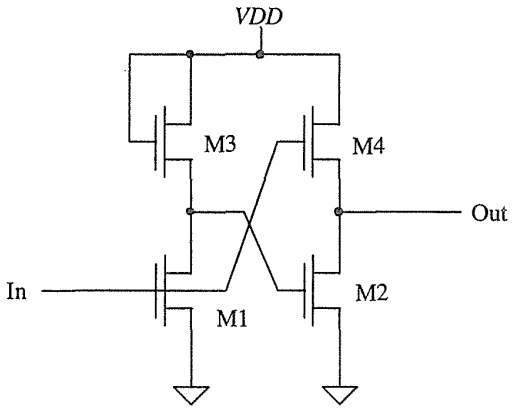



**Figure 11.27** NMOS super buffer.

The reduced output voltage of the NMOS-only output buffer can be improved using the circuit of Fig. 11.28. The inverter driving the gate of M2 uses an on-chip generated DC voltage of nominally *VDD* + 2 V. This allows the output signal to reach

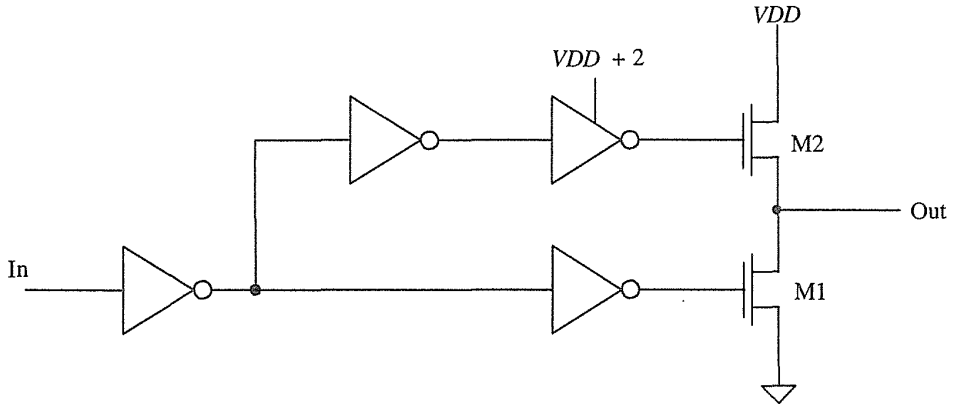



**Figure 11.28** Alternative output buffer.

Microchip Ex. 1055, p. 249
Microchip v. Aptiv
IPR2024-00494

*VDD*. Thus, the output swings from 0 to *VDD* similar to the CMOS output buffer. Note that with the addition of an enabling logic gate, the gates of M1 and M2 can be held at ground, forcing the output into the high-impedance (Hi-Z) state. This is sometimes referred to as tri-state output since the output can be a 1, 0, or Hi-Z.

### 11.5.2 Inverters with Tri-State Outputs

Two configurations used in the design of an inverter with tri-state outputs are shown in Fig. 11.29. A high on the *S* input allows the circuit to operate normally, that is, as an inverter. A low on the S input forces the output into the Hi-Z, or high-impedance state. These circuits are useful when data are shared on a communication bus. The logic symbol of the tri-state inverter is also shown in Fig. 11.29.

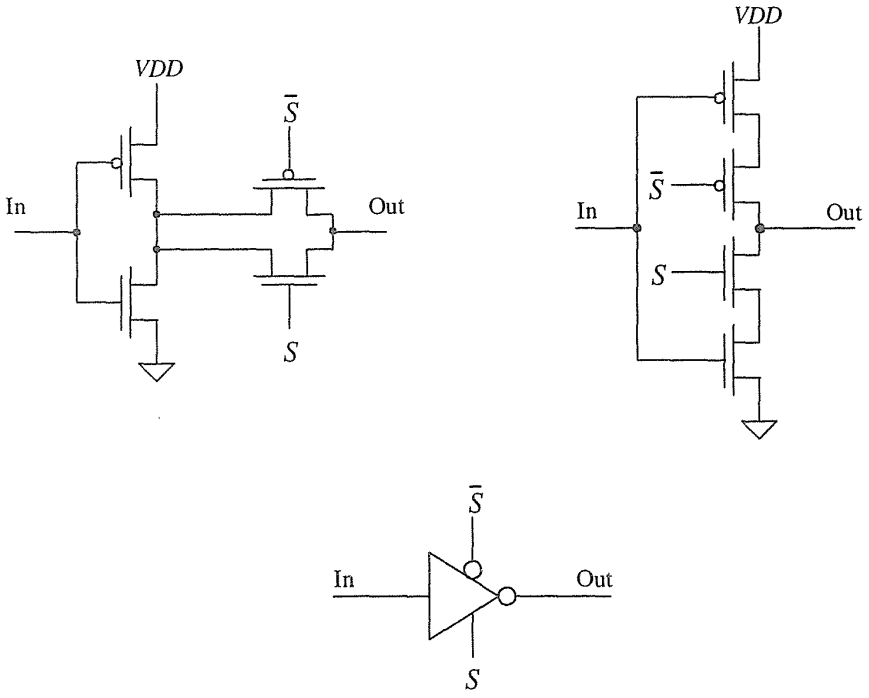



**Figure 11.29** Circuits and logic symbol for the tri-state inverter.

### 11.5.3 The Bootstrapped NMOS Inverter

Consider the modified version of the NMOS inverter of Fig. 11.26 shown in Fig. 11.30. This inverter configuration is called the bootstrapped NMOS inverter. It is used wher the output voltage must swing up to *VDD*. To understand the operation, consider firs the case when input to the inverter is a logic high. The MOSFET M1 is on, and the output is pulled down to approximately

Microchip Ex. 1055, p. 250
Microchip v. Aptiv
IPR2024-00494

$$V_{OL} = (VDD - 2V_{THN}) \cdot \frac{R_{n1}}{R_{n1} + R_{n2}} \tag{11.31}$$

For the device sizes given, $V_{OL}$ is approximately 1/2 V. Next consider the case when the input transitions from a high to a low. The MOSFET M4 is used as a capacitor. The idea is to capacitively couple the output pulse to the gate of M2. The result is an increase in the gate potential above *VDD*, allowing M2 to fully turn on (pulling the output up by its bootstraps). To understand the operation, consider the circuit shown in Fig. 11.31. MOSFET M4 is replaced with a capacitor $C_4$ (= $W_4 \cdot L_4 \cdot C'_{ox}$). When M1 shuts off, with the input going low, a capacitive voltage divider exists between the output and the gate of M2 given by

$$\text{Change in M2's gate voltage} = (VDD - V_{OL}) \cdot \frac{C_4 + C_{inn2}}{C_4 + C_{inn2} + C_{inn3}} \tag{11.32}$$

Without bootstrapping, M3 and M4, the gate of the M2 is tied to *VDD* and the output is limited to $VDD - V_{THN}$. If the gate of M2 is bootstrapped up to *VDD* + 2 V, then M2 fully turns on and the output goes to *VDD*. Therefore, the change in M2's gate potential should be > 2 V. In general, the size ($W \cdot L$) of M4 should be ten times larger than the size of M3. This is equivalent to saying that $C_4$ should be > $10 \cdot C_{inn3}$. Note that this is a dynamic effect. The gate of M2 under DC conditions is $VDD - V_{THN}$, while the output high is $VDD - 2V_{THN}$ or lower than the nonbootstrapped inverter of Fig. 11.26a.

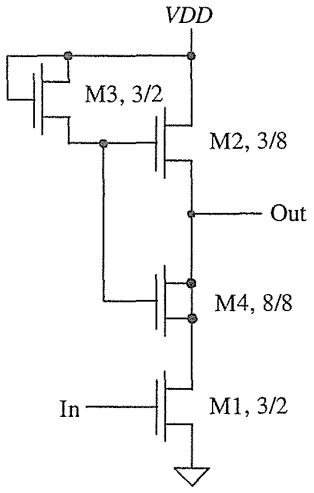



**Figure 11.30** Bootstrapped NMOS inverter.

**Example 11.10**

Simulate the operation of the inverter shown in Fig. 11.30 using SPICE.

The simulation results are shown in Fig. 11.32. Notice how the output doesn't go all the way to ground or *VDD*. We can decrease the size (*W/L*) of M2 (increase

Microchip Ex. 1055, p. 251
Microchip v. Aptiv
IPR2024-00494

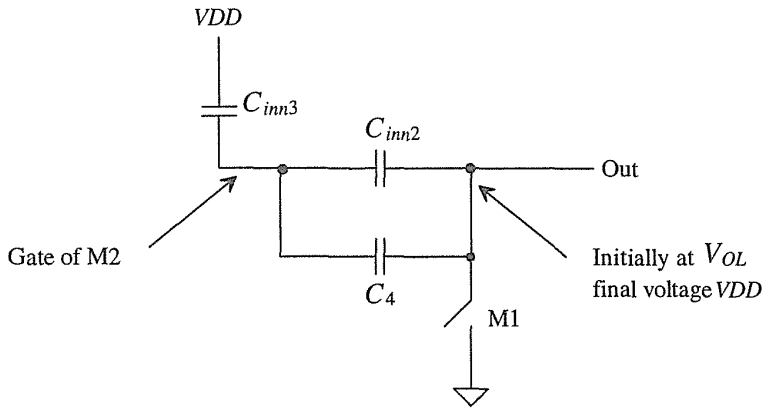



**Figure 11.31** Circuit used to illustrate the bootstrapping effect.

the size of its switching resistance) in Fig. 11.30 to make the output go closer to ground. The price we pay for this is an increase in $t_{PLH}$. Also, the bootstrapped inverter will swing up to *VDD* if we increase the size of the capacitor M4. However, since this capacitor is charged through M3, the resulting increase in charging time will cause the maximum practical operating frequency of the gate to decrease. ■

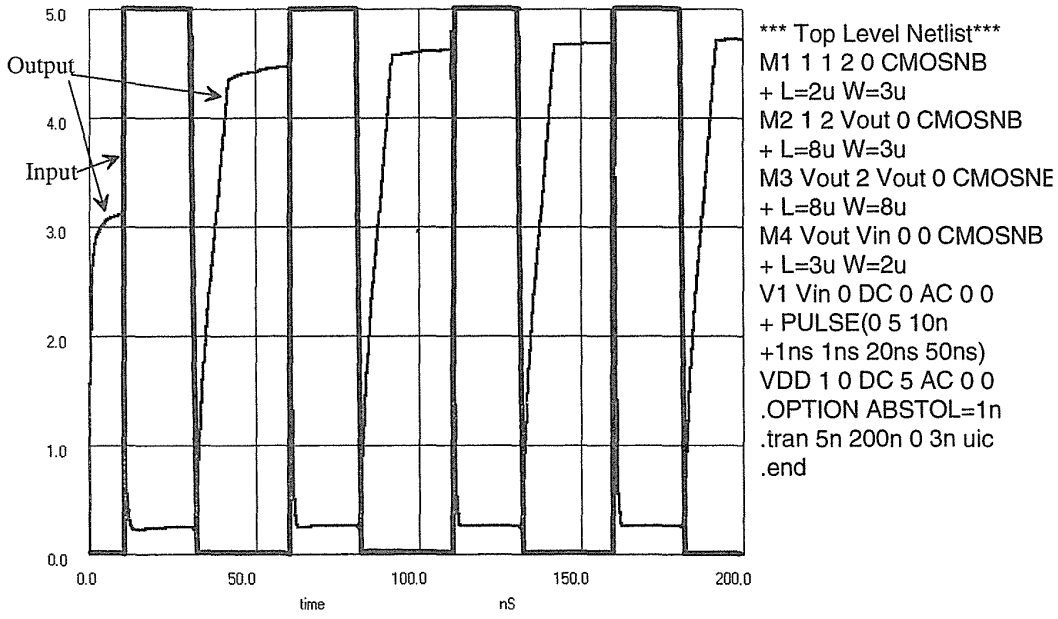



**Figure 11.32** Simulation results for Example 11.10.

Microchip Ex. 1055, p. 252
Microchip v. Aptiv
IPR2024-00494

## REFERENCES

[1] R. L. Geiger, P. E. Allen and N. R. Strader, *VLSI-Design Techniques for Analog and Digital Circuits,* McGraw-Hill Publishing Co., 1990. ISBN 0-07-023253-9.

[2] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design,* Addison- Wesley, 2nd ed., 1993. ISBN 0-201-53376-6.

## PROBLEMS

Use the CN20 process for the following problems unless otherwise stated.

**11.1** Design and simulate the DC characteristics of an inverter with $V_{SP}$ approximately equal to $V_{THN}$. Estimate the resulting noise margins for the design.

**11.2** Repeat Ex. 11.6 for MOSFETs with $W = 10$ μm and a load capacitance of 1 pF.

**11.3** Estimate the oscillation frequency of a 31-stage ring oscillator using minimum-size inverters.

**11.4** Lay out the standard-cell frame of Fig. 11.16. Explain how the added implants help to reduce latch-up.

**11.5** Design and simulate the operation of a buffer to drive a 50 pF capacitive load from an inverter with size of 150/50. The $t_{PHL} + t_{PLH}$ should be less than 10 ns.

**11.6** Repeat Ex. 11.9, using a minimum delay of 20 ns, where the first inverter in the series is minimum size, that is, 3/2 (p-channel) and 3/2 (n-channel).

**11.7** Design and simulate the delay of a minimum-size inverter driving a 1 mm poly line terminated with a 1 pF capacitor.

**11.8** Lay out an inverter with a size of 450/150 using the standard-cell frame of Ch. 5.

**11.9** Simulate the operation and explain the results for the NMOS super buffer shown in Fig. 11.27.

**11.10** Repeat Ex. 11.10 if M4's size is increased to 20/20.

**11.11** Repeat Ex. 11.5 using minimum-size (0.9/0.6) MOSFETs in the CMOS14TB process.

**11.12** Repeat Ex. 11.6 using minimum-size MOSFETs in CMOS14TB.

**11.13** Sketch the cross-sectional views, at the positions indicated, for the layout shown in Fig. P11.13.

Microchip Ex. 1055, p. 253
Microchip v. Aptiv
IPR2024-00494

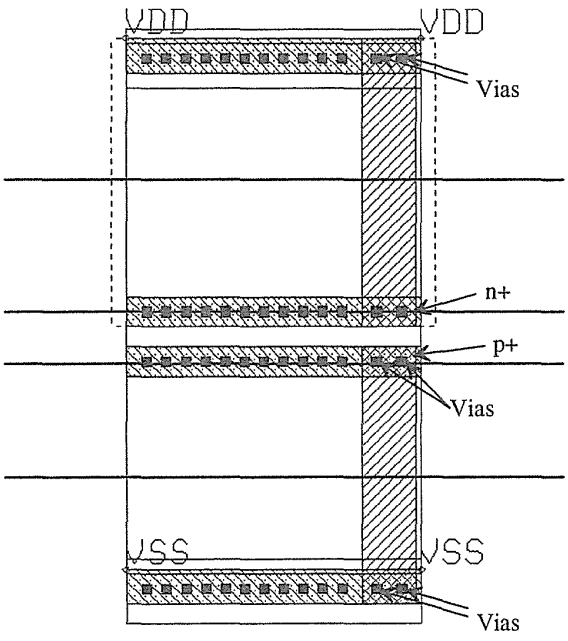



**Figure P11.13**

Microchip Ex. 1055, p. 254
Microchip v. Aptiv
IPR2024-00494

Chapter

12

# Static Logic Gates

In this chapter we discuss the DC characteristics, dynamic behavior, and layout of CMOS static logic gates. Static logic means that the output of the gate is always a logical function of the inputs and always available on the outputs of the gate regardless of time. We begin with the NAND and NOR gates.

## 12.1 DC Characteristics of the NAND and NOR Gates

The two basic input NAND and NOR gates are shown in Fig. 12.1. Before we get into the operation, notice that each input into the gate is connected to both a p- and an n-channel transistor similar to the inverter of the last chapter. We will make use of the results of Ch. 11 to explain the operation of these gates.

### 12.1.1 DC Characteristics of the NAND Gate

The NAND gate of Fig. 12.1 requires both inputs to be high before the output will switch low. Let's begin our analysis by determining the voltage transfer curve of a gate with p-channel MOSFETs that have $W = W_p$, $L = L_p$, and n-channel MOSFETs with $W = W_n$, $L = L_n$. If both inputs of the gate are tied together, the gate behaves like an inverter.

To determine the gate switching point voltage, $V_{SP}$, we must remember that two MOSFETs in parallel behave like a single MOSFET with a width equal to the sum of the individual widths. For the two parallel p-channel MOSFETs in Fig. 12.1, we can write

$$W_3 + W_4 = 2W_p \tag{12.1}$$

again assuming that all p-channel transistors are of the same size. The transconductance parameters can also be combined into the transconductance parameter of a single MOSFET, or

$$\beta_3 + \beta_4 = 2\beta_p \tag{12.2}$$

Microchip Ex. 1055, p. 255
Microchip v. Aptiv
IPR2024-00494

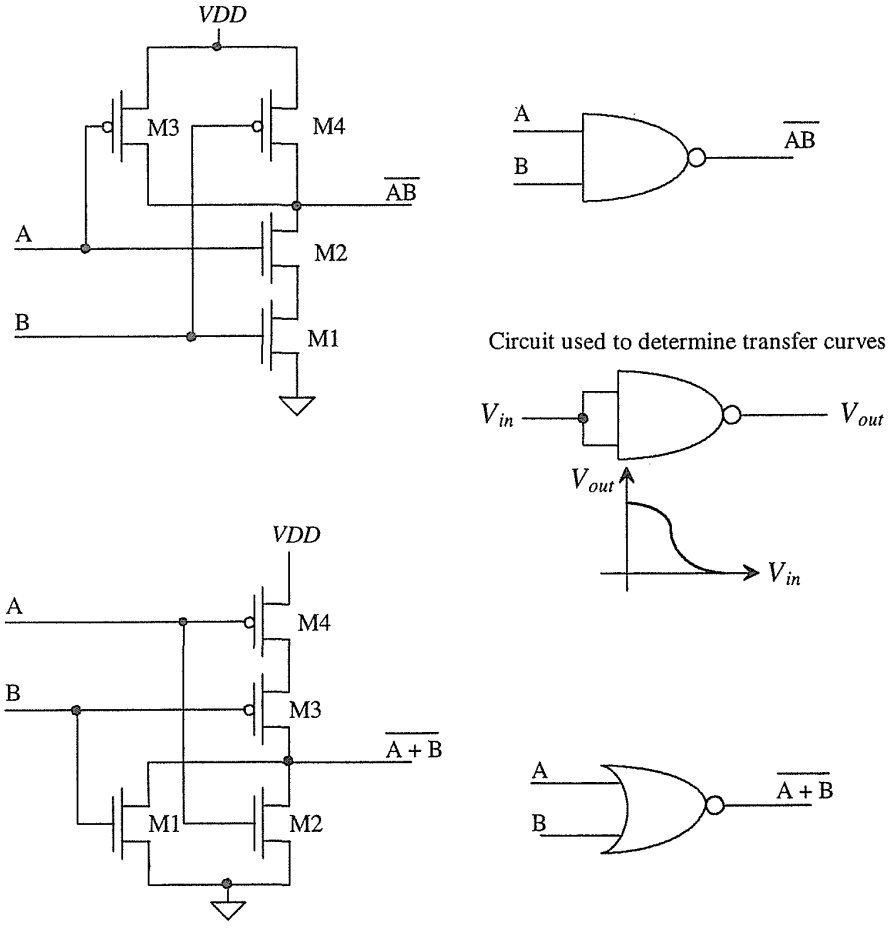



**Figure 12.1** NAND and NOR gate circuits and logic symbols.

If we neglect the body effect, then two MOSFETs in series (with their gates tied together) behave like a single MOSFET with a channel length equal to the sum of the individual MOSFET lengths. Referring to the figure above of the NAND, we can write for the n-channel MOSFETs

$$L_1 + L_2 = 2L_n \tag{12.3}$$

and the transconductance of the single MOSFET is given by

$$\beta_1 + \beta_2 = \frac{\beta_n}{2} \tag{12.4}$$

If we model the NAND gate with both inputs tied together as an inverter with an n-channel transistor having a width of $W_n$ and length $2L_n$ and a p-channel MOSFET with a width of $2W_p$ and length $L_p$, then we can write the transconductance ratio as

Microchip Ex. 1055, p. 256
Microchip v. Aptiv
IPR2024-00494

$$\text{Transconductance ratio of NAND gate} = \frac{\beta_n}{4\beta_p} \tag{12.5}$$

The switching point voltage, with the help of Eq. (11.4), of the two-input NAND gate is then given by

$$V_{SP} = \frac{\sqrt{\frac{\beta_n}{4\beta_p}} \cdot V_{THN} + (VDD - V_{THP})}{1 + \sqrt{\frac{\beta_n}{4\beta_p}}} \tag{12.6}$$

or in general for an n-input NAND gate (see Fig. 12.2), we get

$$V_{SP} = \frac{\sqrt{\frac{\beta_n}{N^2 \cdot \beta_p}} \cdot V_{THN} + (VDD - V_{THP})}{1 + \sqrt{\frac{\beta_n}{N^2 \cdot \beta_p}}} \tag{12.7}$$

Again, it should be remembered that we have neglected the body effect (an increase in the threshold voltage with increasing $V_{SB}$). Voltage transfer curves using one input, with the others tied to *VDD*, will give slightly different results because of this effect.

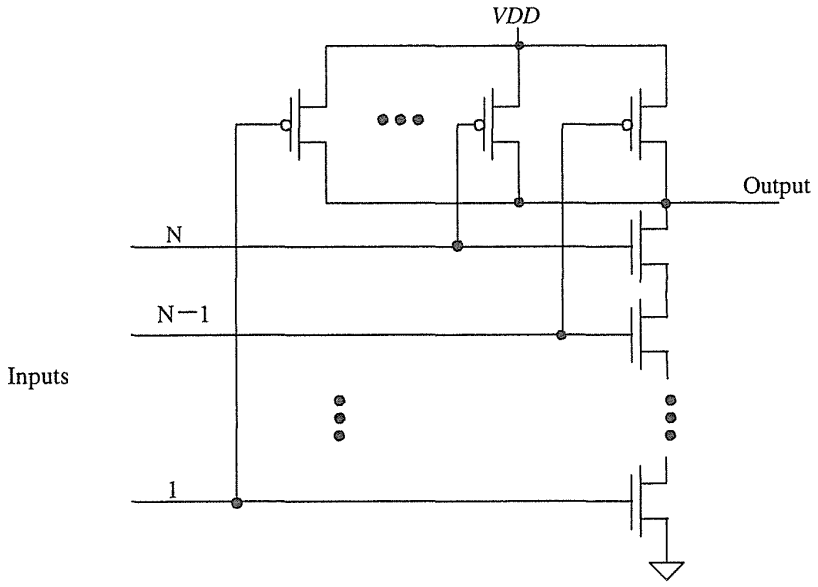



**Figure 12.2** Schematic of an n-input NAND gate.

**Example 12.1**

Determine $V_{SP}$ by hand calculations and compare to a SPICE simulation for a three-input NAND gate using minimum-size devices.

Microchip Ex. 1055, p. 257
Microchip v. Aptiv
IPR2024-00494

The switching point voltage is determined by calculating the transconductance ratio of the gate, or

$$\sqrt{\frac{\beta_n}{N^2\beta_p}} = \sqrt{\frac{\frac{50\mu A/V^2 \cdot 3\mu m}{2\mu m}}{9 \cdot \frac{17\mu A/V^2 \cdot 3\mu m}{2\mu m}}} = 0.572$$

and then using Eq. (12.7),

$$V_{SP} = \frac{0.572 \cdot (0.83) + (5 - 0.92)}{1.572} = 2.9 \text{ V}$$

The SPICE simulation results are shown in Fig. 12.3. The simulation gives a $V_{SP}$ of approximately 3.1 V. ■

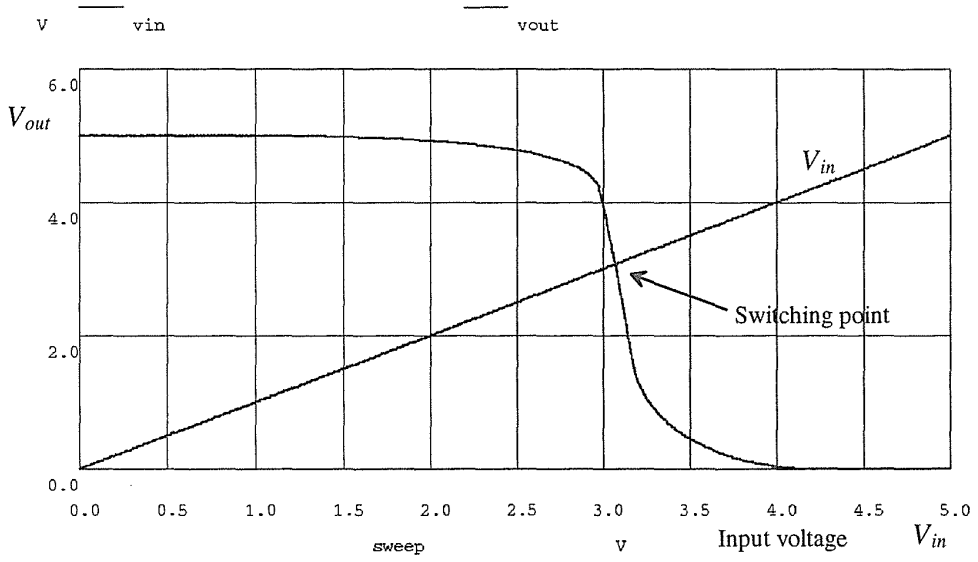



**Figure 12.3** Voltage transfer characteristics of the three-input minimum-size NAND gate.

### 12.1.2 DC Characteristics of the NOR gate

Following a similar analysis for the n-input NOR gate (see Fig. 12.4) gives a switching point voltage of

$$V_{SP} = \frac{\sqrt{\frac{N^2 \cdot \beta_n}{\beta_p}} \cdot V_{THN} + (VDD - V_{THP})}{1 + \sqrt{\frac{N^2 \cdot \beta_n}{\beta_p}}} \qquad (12.8)$$

Microchip Ex. 1055, p. 258
Microchip v. Aptiv
IPR2024-00494

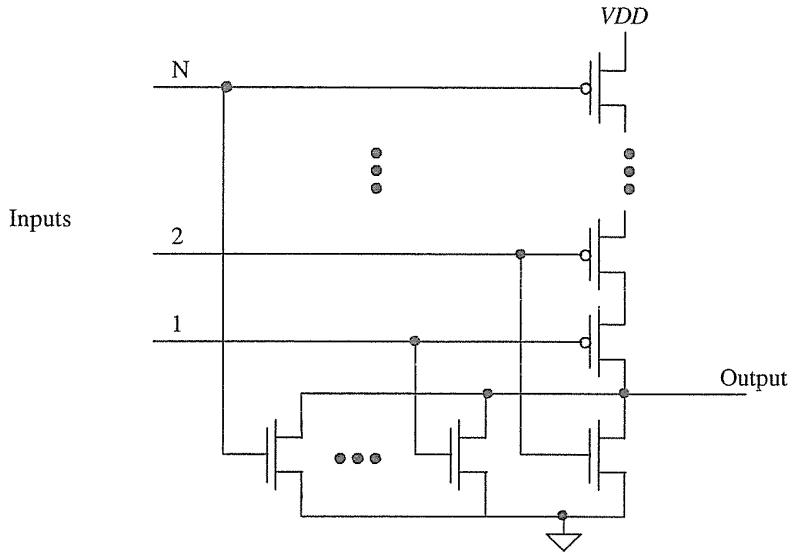



**Figure 12.4** Schematic of an n-input NOR gate.

**Example 12.2**

Compare the switching point voltage of a three-input NOR gate made from minimum-size MOSFETs to that of the three-input NAND gate of Ex. 12.1. Comment on which gate is closer to ideal, that is, $V_{SP} = VDD/2$.

The $V_{SP}$ of the minimum-size three-input NOR gate is 1.35 V, while the $V_{SP}$ of the minimum-size three-input NAND gate was calculated to be 2.9 V. For an ideal gate $V_{SP}$ = 2.5 V, so that the NAND gate is closer to ideal than the NOR gate. This arises because the transconductance (actually the mobility) of the n-channel is larger than that of the p-channel. In CMOS digital design, the NAND gate is used most often. This is due partly to the DC characteristics, better noise margins, and the dynamic characteristics. We will also see shortly that the NAND gate has better transient characteristics than the NOR gate. ■

## 12.2 Layout of the NOR and NAND Gates

Layout of the three-input minimum-size NOR and NAND gates is shown in Fig. 12.5 using the standard-cell frame. MOSFETs in series, for example, the n-channel MOSFETs in the NAND gate, are laid out using a single-drain and a single-source contact. The active area between the gate poly is shared between two devices. This has the effect of reducing the parasitic drain/source implant capacitances. MOSFETs in parallel, for example, the n-channel MOSFETs in the NOR gate, can share a drain area or a source area. The inputs and outputs of the gates are on poly.

Microchip Ex. 1055, p. 259
Microchip v. Aptiv
IPR2024-00494

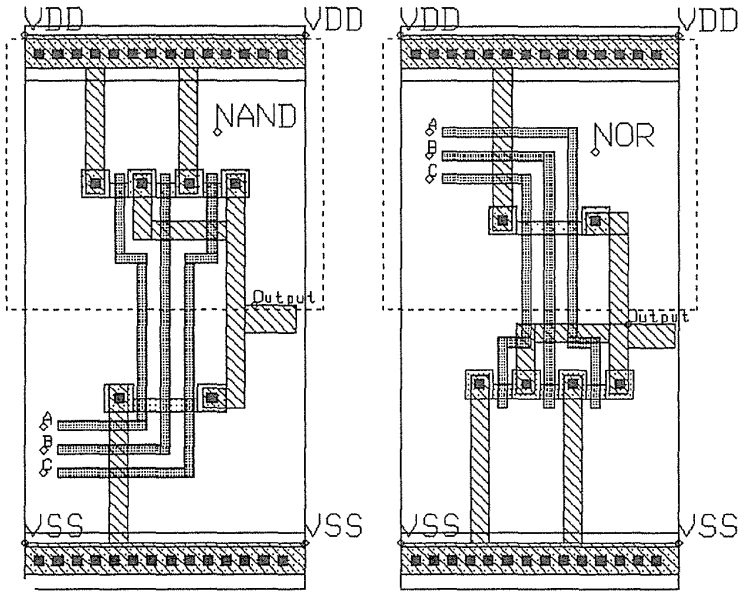



**Figure 12.5** Layout of the NAND and NOR gate.

## 12.3 Switching Characteristics

Consider the parallel connection of identical MOSFETs shown in Fig. 12.6 with their gates tied together. From the equivalent digital models, also shown, we can determine the intrinsic time constant of this chain of N MOSFETs by

$$t_{PLH} = \frac{R_p}{N} \cdot (N \cdot C_{outp}) = R_p C_{outp} \tag{12.9}$$

which for CN20 from Ex. 11.5 is 230 ps. With an external load capacitance, the low to high delay-time becomes

$$t_{PLH} = \frac{R_p}{N}(N \cdot C_{outp} + C_{load}) \tag{12.10}$$

This again assumes that the MOSFET's gates are tied together. For n-channel MOSFETs in parallel, a similar analysis yields

$$t_{PHL} = \frac{R_n}{N}(N \cdot C_{outn} + C_{load}) \tag{12.11}$$

The load capacitance, $C_{load}$, consists of all capacitances on the output node except the output capacitances of the MOSFETs in parallel.

Consider the series connection of identical n-channel MOSFETs shown in Fig. 12.7. We can estimate the intrinsic switching time of series-connected MOSFETs by

Microchip Ex. 1055, p. 260
Microchip v. Aptiv
IPR2024-00494

$$t_{PHL} = N \cdot R_n \left( \frac{C_{outn}}{N} \right) + 0.35 \cdot R_n C_{inn} (N-1)^2 \qquad (12.12)$$

The first term in this equation represents the intrinsic switching time of the series connection of MOSFETs, while the second term represents $RC$[1] delay caused by $R_n$ charging $C_{inn}$. For the case when $N = 1$, this simply reduces to $R_n C_{outn}$. With an external load capacitance, the high to low delay-time becomes

$$t_{PHL} = N \cdot R_n \cdot \left( \frac{C_{outn}}{N} + C_{load} \right) + 0.35 \cdot R_n C_{inn} (N-1)^2 \qquad (12.13)$$

For p-channel MOSFETs in series, a similar analysis yields

$$t_{PLH} = N \cdot R_p \cdot \left( \frac{C_{outp}}{N} + C_{load} \right) + 0.35 \cdot R_p C_{inp} (N-1)^2 \qquad (12.14)$$

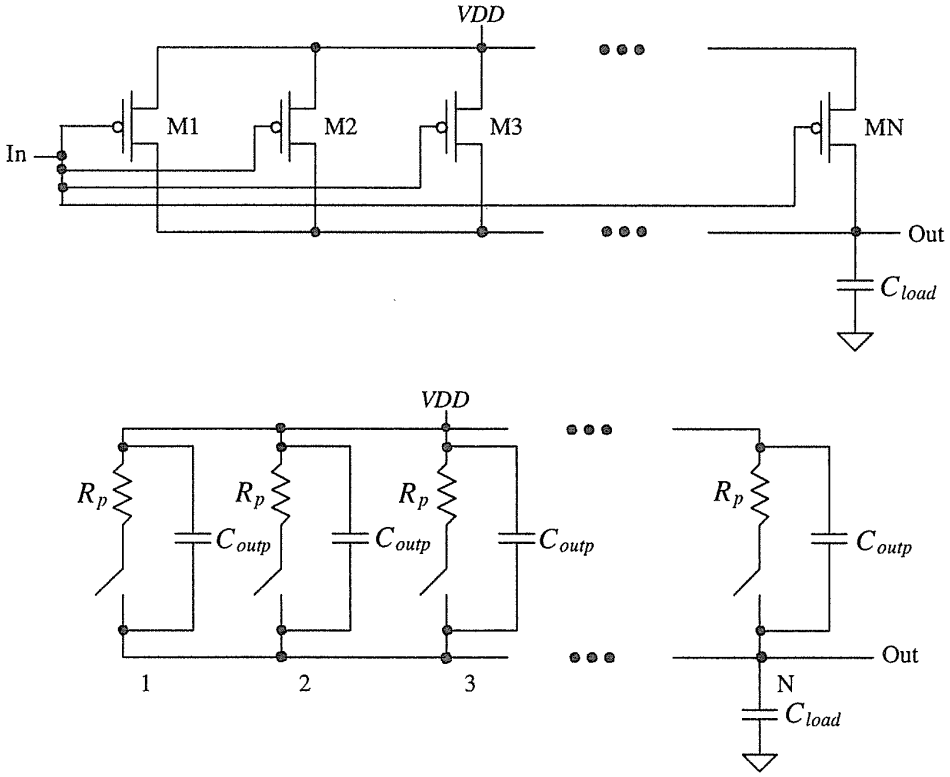



**Figure 12.6** Parallel connection of MOSFETs and equivalent digital model.

[1] This effect is similar to the delay through an RC transmission line. The $N$ - 1 term is due to the fact that the last MOSFET's source in the string is connected to ground and not to a load.

Microchip Ex. 1055, p. 261
Microchip v. Aptiv
IPR2024-00494

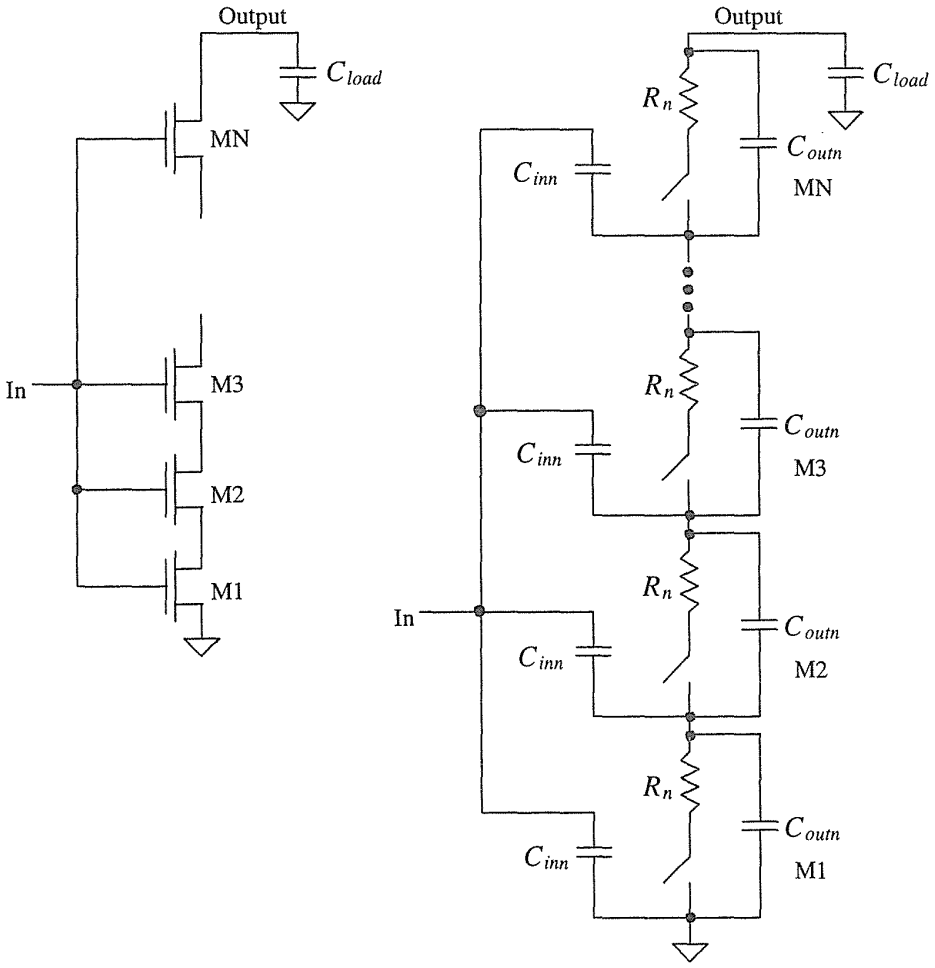



**Figure 12.7** Series connection of MOSFETs and equivalent digital model.

These equations are approximations for the propagation delays giving results usually to within a factor of two of the measurements.

### 12.3.1 NAND Gate

Consider the n-input NAND gate of Fig. 12.8 driving a capacitive load $C_{load}$. The low-to-high propagation time, using Eq. (12.10), is

$$t_{PLH} = \frac{R_p}{N}\left(N \cdot C_{outp} + \frac{C_{outn}}{N} + C_{load}\right) \tag{12.15}$$

where here $C_{load}$ represents the capacitance external to the gate, whereas in Eq. (12.10) $C_{load}$ represented the capacitance external to the parallel p-channel MOSFETs. If the

Microchip Ex. 1055, p. 262
Microchip v. Aptiv
IPR2024-00494

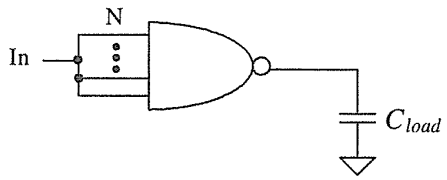



**Figure 12.8** An n-input NAND gate driving a load capacitance.

load capacitance is much greater than the output capacitance of the gate, the low to high propagation time can be estimated by

$$t_{PLH} \approx \frac{R_p}{N} \cdot C_{load} \tag{12.16}$$

The high to low propagation time, using Eq. (12.13), is given by

$$t_{PHL} = N \cdot R_n \left[ \frac{C_{outn}}{N} + N \cdot C_{outp} + C_{load} \right] + 0.35 \cdot R_n C_{inn} (N-1)^2 \tag{12.17}$$

or if $C_{load}$ is much larger than the output capacitance of the gate

$$t_{PHL} \approx N \cdot R_n \cdot C_{load} \tag{12.18}$$

**Example 12.3**

Estimate the intrinsic propagation delays, $t_{PHL} + t_{PLH}$, of a three-input NAND gate made using minimum-size transistors. Estimate and simulate the delay when the gate is driving a load capacitance of 100 fF. Assume that inputs are tied together.

For the three-input NAND gate, the three p-channel MOSFETs in parallel are used to pull the output high. Since they are minimum-size

$$R_p = 24 \text{ k}\Omega \text{ and } C_{outp} = 4.8 \text{ fF}$$

this gives a low-to-high propagation time, using Eq. (12.15), with $C_{load} = 0$, of

$$t_{PLH} = \frac{24\text{k}}{3} \left[ 3 \cdot 4.8 \text{ fF} + \frac{4.8\text{fF}}{3} \right] = 128 \text{ ps}$$

The intrinsic high-to-low propagation time of the three input NAND gate with minimum-size n-channel MOSFETs ($R_n$ = 8 kΩ and $C_{outn}$ = 4.8 fF) using Eq. (12.17), with $C_{load} = 0$,

$$t_{PHL} = 3 \cdot 8\text{k} \cdot \left[ \frac{4.8\text{fF}}{3} + 3 \cdot 4.8\text{fF} \right] + 0.35 \cdot 8\text{k} \cdot 7.2\text{fF}(3-1)^2 \approx 500\,\text{ps}$$

With a load of 100 fF, the propagation delays become, $t_{PLH}$ = 928 ps and $t_{PHL}$ = 2.9 ns. The SPICE simulation results are shown in Fig. 12.9 followed by the SPICE3 netlist. Helping with convergence, the .OPTIONS statement was used, the initially off n-channel MOSFETs were specified, and the input transition times were specified at 0.1 ns instead of the unrealistic conditions of 1 ps. ■

Microchip Ex. 1055, p. 263
Microchip v. Aptiv
IPR2024-00494

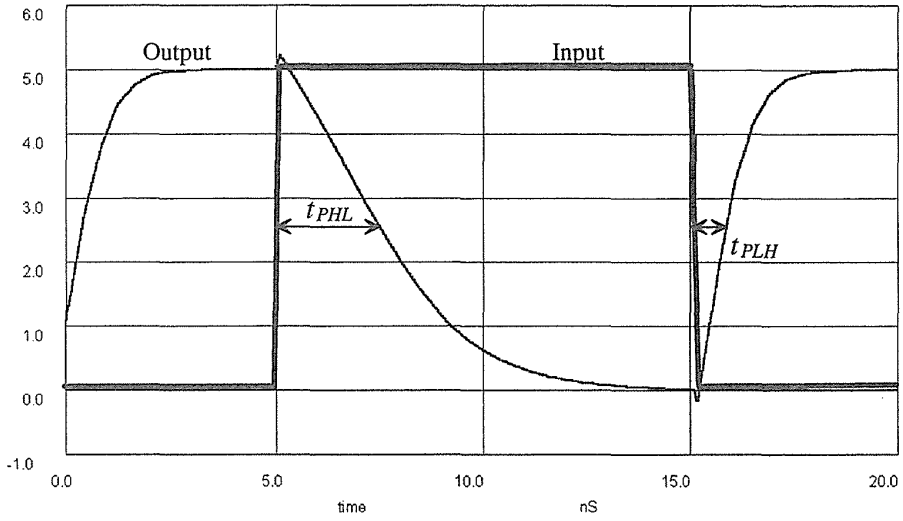



**Figure 12.9** Output of the minimum-size NAND gate driving a 100 fF capacitor.

```
*** Top Level Netlist for Example 12.3 ***.

C1      5 0 100f
M1      5 1 2 0 CMOSNB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u OFF
M2      2 1 4 0 CMOSNB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u OFF
M3      4 1 0 0 CMOSNB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u OFF
M4      5 1 Vdd Vdd CMOSPB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u
M5      5 1 Vdd Vdd CMOSPB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u
M6      5 1 Vdd Vdd CMOSPB  L=2u W=3u AD=36p AS=36p PD=24u PS=24u
V1      Vdd 0  DC 5
V2      1 0      DC 0 PULSE(0 5 5n .1n .1n 10n )

***** Spice models and macro models *****

.MODEL CMOSNB NMOS LEVEL=4
+VFB=-9.73820E-01, LVFB=3.67458E-01,WVFB=-4.72340E-02
See Appendix A for a complete listing.

.MODEL CMOSPB PMOS LEVEL=4
+ vfb=-2.65334E-01, lvfb=6.50066E-02, wvfb=1.48093E-01
See Appendix A for a complete listing.

.OPTION ABSTOL=1u RELTOL=0.01 VNTOL=1mv ITL4=100
.probe
.tran 100p 20n 0  uic
.plot tran all
.print tran all
.end
```

Microchip Ex. 1055, p. 264
Microchip v. Aptiv
IPR2024-00494

The delay equations derived in this section are useful in understanding the limitations on the number of MOSFETs used in a NAND gate for high-speed design. However a more useful, though not as precise, method of determining delays can be found by considering the fact that whenever the output changes from *VDD* to ground the discharge path is through $N$ resistors of value $R_n$. This is true if all or only one of the inputs to the NAND gate changes, causing the output to change. Under these circumstances, Eq. (12.18) predicts the high-to-low delay-time, or for *series* connection of $N$ n-channel MOSFETs,

$$t_{PHL} \approx N \cdot R_n \cdot C_{load} \tag{12.19}$$

The case when the output of the NAND gate changes from a low to a high is somewhat different then the high-to-low case. Referring to Fig. 12.6, we see that if one of the MOSFETs turns on, it can pull the output to *VDD* independent of the number of MOSFETs in parallel. Under these circumstances, Eq. (12.16) can be used with $N = 1$ to predict the low-to-high delay-time, or for a parallel connection of $N$ p-channel MOSFETs,

$$t_{PLH} \approx R_p \cdot C_{load} \tag{12.20}$$

We will try to use Eqs. (12.19) and (12.20) as much as possible because of their simplicity. The further simplified digital models of the n- and p-channel MOSFETs are shown in Fig. 12.10. (Input capacitance is not shown.)




**Figure 12.10** Further simplification of digital models not showing input capacitance.

**Example 12.4**

Estimate, using Eqs. (12.19) and (12.20), the propagation delays for the three-input minimum-size NAND gate, with only one input switching driving a 100 fF load capacitance. Compare your results to SPICE.

The propagation delay-times are given by

$$t_{PHL} = 3 \cdot 8\text{k} \cdot 100\text{fF} = 2.4\,\text{ns}$$

and

$$t_{PLH} = 24\text{k} \cdot 100\,\text{fF} = 2.4\ \text{ns}$$

The SPICE simulation results gave $t_{PLH} \approx t_{PHL} \approx 2.3$ ns with one input switching. If we had used Eqs. 12.19 and 12.20 in Ex. 12.3 where all inputs where changing at the same time, the calculated $t_{PHL}$ would have given an

Microchip Ex. 1055, p. 265
Microchip v. Aptiv
IPR2024-00494

underestimate (but not by much), while the calculated $t_{PLH}$ would have overestimated the delay. Also note that since the effective resistance of the p-channel is three times greater than that of the n-channel, the series connection of three NMOS devices gives approximately the same resistance as the single PMOS. The result is equal switching times and the reason the NAND gate is generally preferred over the NOR gate in CMOS circuit design. ■

### 12.3.2 Number of Inputs

As the number of inputs, $N$, to a static NAND (or NOR) gate increases, the scheme shown in Fig. 12.2 (Fig. 12.4) becomes difficult to realize. Consider a NOR gate with 100 inputs. This gate requires 100 p-channel MOSFETs in series and a total of 200 MOSFETs ($2N$ MOSFETs). The delay associated with the series p-channel MOSFETs charging of a load capacitance is too long for most practical situations.

Now consider the schematic of an $N$ input NOR gate shown in Fig. 12.11, which uses $N + 1$ MOSFETs. If any input to the NOR gate is high, the output is pulled low through the corresponding n-channel MOSFET to a voltage, when designed properly, of a few hundred millivolts. If all inputs are low, then all n-channel MOSFETs are off and the p-channel MOSFET pulls the output high (to *VDD*). A simple analysis of the output low voltage, $V_{OL}$, with *one* input at *VDD* yields

$$\frac{\beta_p}{2}(VDD - V_{THP})^2 = \beta_n\left[(VDD - V_{THN})V_{OL} - \frac{V_{OL}^2}{2}\right] \qquad (12.21)$$

Assuming that the maximum $V_{OL}$ allowed (the more inputs at *VDD* the lower $V_{OL}$) is 500 mV and that the n-channels have $W = 3$ μm and $L = 2$ μm results in a $W$ of 4 μm and an $L$ of 3 μm for the p-channel. In practice, the length of the p-channel can be increased beyond this size to lower $V_{OL}$ further. The static power dissipated by this gate when the output is high, neglecting leakage currents, is zero. When the output is low, a static power is dissipated due to both n- and p-channels conducting. The current that flows under this condition with the above sizes is 150 μA. Decreasing the W/L of the p-channel lowers power draw at the price of increased $t_{PLH}$.

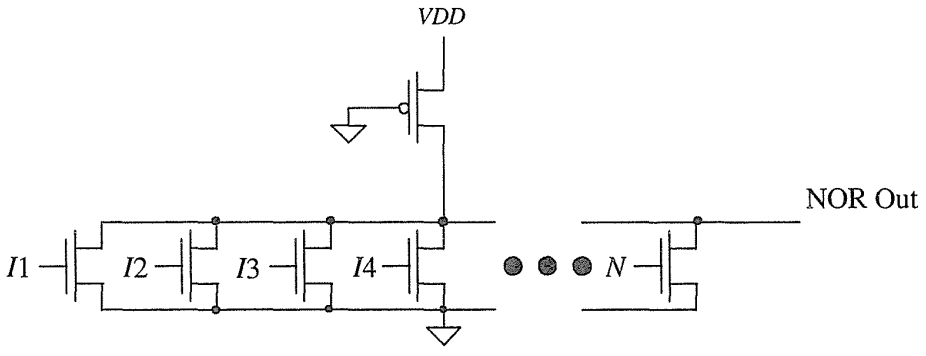



**Figure 12.11** NOR configuration used for a large number of inputs.

Microchip Ex. 1055, p. 266
Microchip v. Aptiv
IPR2024-00494

## 12.4 Complex CMOS Logic Gates

Implementation of complex logic functions in CMOS uses the basic building blocks shown in Fig. 12.12. We have already used the circuits to implement NAND and NOR gates. In general, any And-Or-Invert (AOI) logic function can be implemented using these techniques. A major benefit of AOI logic is that for a relatively complex logic function the delay can be significantly lower than a logic gate implementation. Consider the following example.

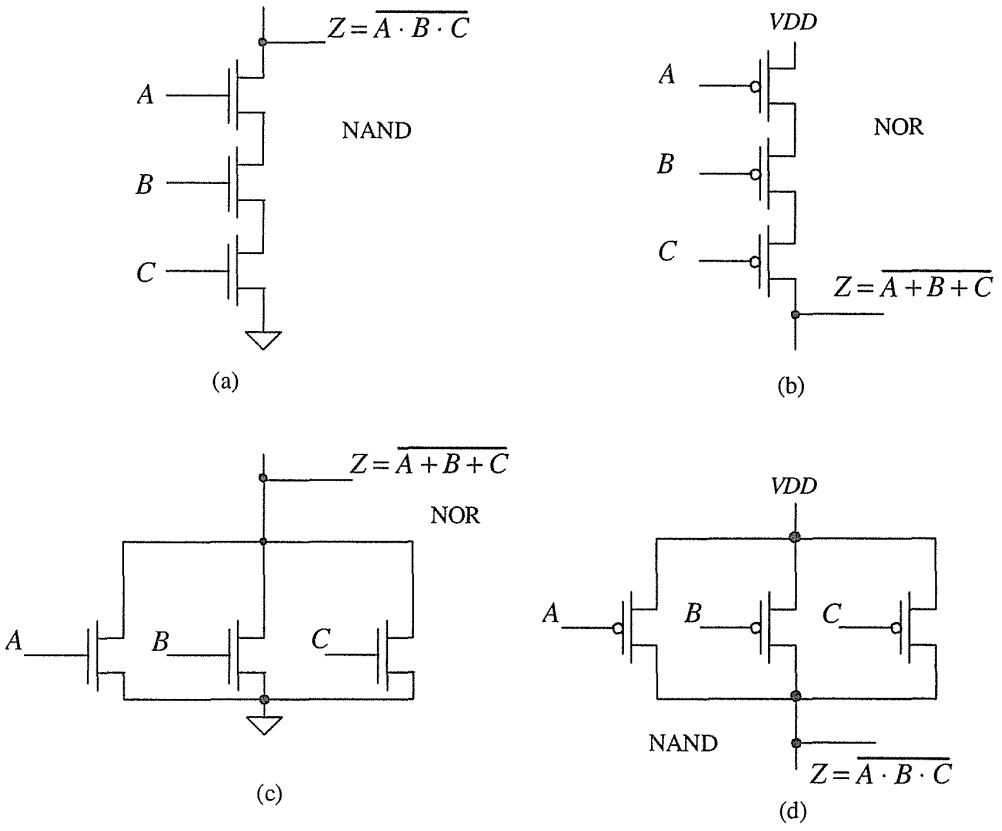



**Figure 12.12** Logic implementation in CMOS.

**Example 12.5**

Using AOI logic, implement the following logic functions:

$$Z=\overline{A}+BC \quad \text{and} \quad Z=A+\overline{B}C+CD$$

Microchip Ex. 1055, p. 267
Microchip v. Aptiv
IPR2024-00494

The implementation of the first function is shown in Fig. 12.13a. Notice that the p-channel configuration is the dual of the n-channel circuit. The function we obtain is the complement of the desired function, and therefore an inverter is used to obtain Z. Using an inverter is, in general, undesirable if both true and complements of the input variables are available. Applying Boolean algebra to the logic function, we obtain

$$Z=\overline{A}+BC \Rightarrow \overline{Z}=\overline{\overline{A}+BC}=A\cdot(\overline{B}+\overline{C}) \Rightarrow Z=\overline{A\cdot(\overline{B}+\overline{C})}$$

The AOI implementation of the result is shown in Fig. 12.13b. Logically, the circuits of Figs. 12.13a and b are equivalent. However, the circuit of Fig. 12.13b is simpler and thus more desirable. Note that to reduce the output capacitance

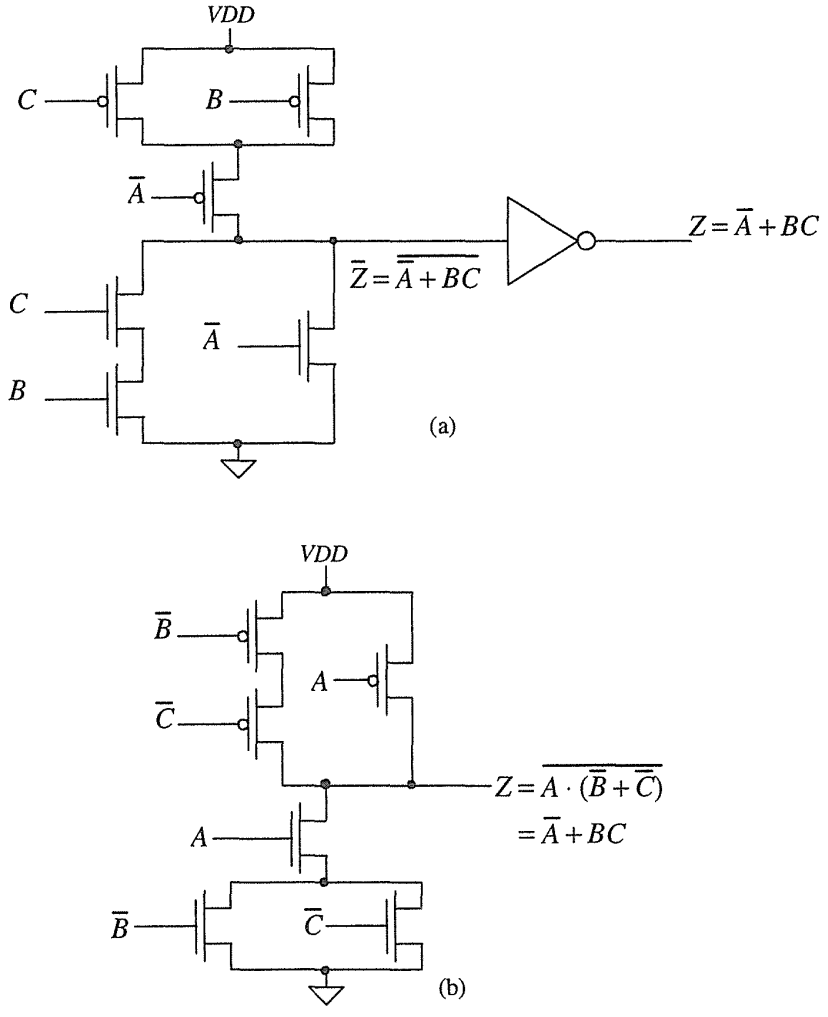



**Figure 12.13** First logic gate of Ex. 12.5.

Microchip Ex. 1055, p. 268
Microchip v. Aptiv
IPR2024-00494

and thus decrease the switching times, the parallel combination of n-channel MOSFETs is placed at the bottom of the logic block.

The second logic function is given by

$$Z = A + \overline{B}C + CD = A + C(\overline{B} + D) \Rightarrow \overline{Z} = \overline{A + C(\overline{B} + D)} = \overline{A} \cdot (\overline{C} + B\overline{D})$$

or

$$Z = \overline{\overline{A} \cdot (\overline{C} + B\overline{D})}$$

The logic implementation is given in Fig. 12.14. ■

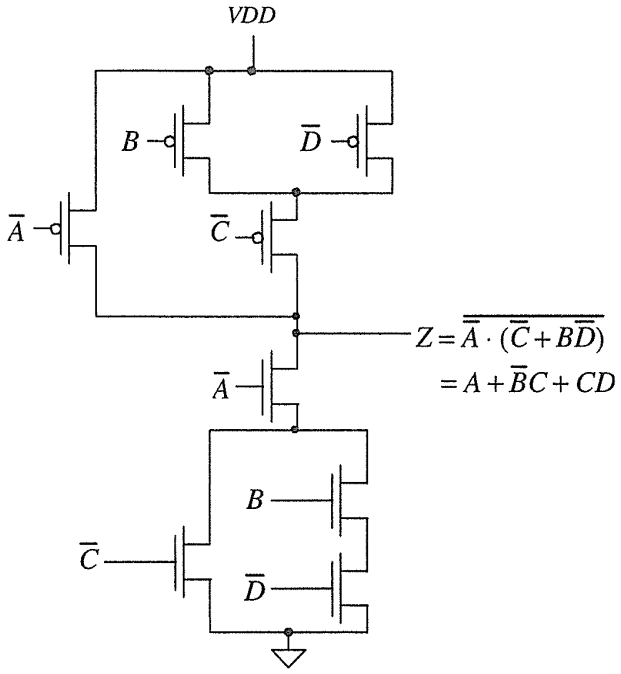



**Figure 12.14** Second logic gate of Ex. 12.5.

**Example 12.6**
Using AOI logic, implement an exclusive OR gate (XOR).

The logic symbol and truth table for an XOR gate are shown in Fig. 12.15. From the truth table, the logic function for the XOR gate is given by

$$Z = A \oplus B = (A + B) \cdot (\overline{A} + \overline{B}) \qquad (12.22)$$

or

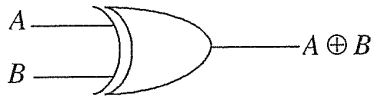
Microchip Ex. 1055, p. 269
Microchip v. Aptiv
IPR2024-00494

$$\overline{Z} = \overline{A \oplus B} = \overline{(A+B)\cdot(\overline{A}+\overline{B})} = \overline{A}\cdot\overline{B} + A\cdot B$$

and finally

$$Z = \overline{\overline{A}\cdot\overline{B} + A\cdot B} = A \oplus B \tag{12.23}$$

The CMOS AOI implementation of an XOR gate is shown in Fig. 12.16. ■




| A | B | A ⊕ B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**Figure 12.15** Exclusive OR gate.

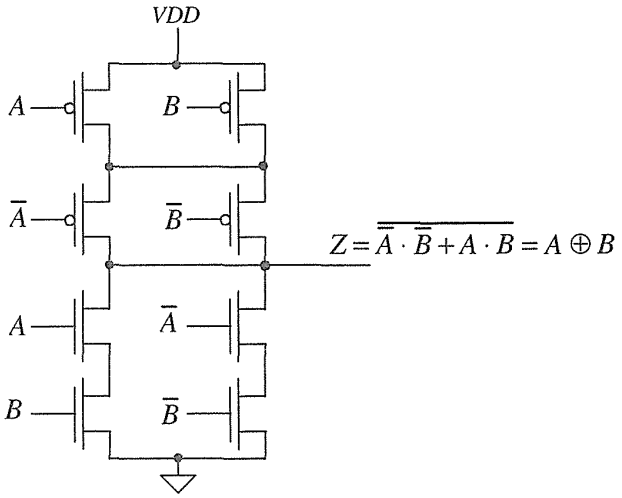



**Figure 12.16** CMOS AOI XOR gate.

**Example 12.7**

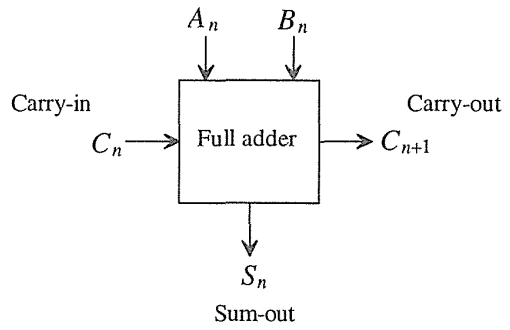
Design a CMOS full adder using CMOS AOI logic.

The logic symbol and truth table for a full adder circuit are shown in Fig. 12.17. The logic functions for the sum and carry outputs can be written as

$$S_n = A_n \oplus B_n \oplus C_n$$

$$C_{n+1} = A_n \cdot B_n + C_n(A_n + B_n)$$

Microchip Ex. 1055, p. 270
Microchip v. Aptiv
IPR2024-00494




| $A_n$ | $B_n$ | $C_n$ | $S_n$ | $C_{n+1}$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

**Figure 12.17** Full adder.

The logic expression for the sum can be rewritten as a sum of products

$$S_n = \overline{A}_n\overline{B}_nC_n + \overline{A}_nB_n\overline{C}_n + A_n\overline{B}_n\overline{C}_n + A_nB_nC_n$$

or since

$$\overline{C}_{n+1} = \left(\overline{A}_n + \overline{B}_n\right) \cdot \left(\overline{C}_n + \overline{A}_n \cdot \overline{B}\right.$$

the sum of products can be rewritten as

$$S_n = (A_n + B_n + C_n)\overline{C}_{n+1} + A_nB_nC_n$$

The AOI implementation of the full adder is shown in Fig. 12.18. ■

### 12.4.1 Cascode Voltage Switch Logic

Cascode voltage switch logic (CVSL) or differential cascode voltage switch logic (DVSL) is a differential output logic that uses positive feedback to speed up the switching times (in some cases). Figure 12.19 shows the basic idea. A gate cross-connected load is used instead of using p-channel switches, as in the AOI logic, to pull the output high. Consider the implementation of $Z = \overline{A} + BC$. (This logic function was implemented in AOI in Fig. 12.13.) N-channel MOSFETs are used to implement $Z$ and $\overline{Z}$ as shown in Fig. 12.20. Figure 12.21a shows the implementation of a two-input XOR/XNOR gate using CSVL, while Fig. 12.21b shows a CSVL three-input XOR/XNOR gate useful in adder design.

### 12.4.2 Differential Split-Level Logic

Differential split-level logic (DSL logic) is a scheme wherein the load is used to reduce output voltage swing and thus lower gate delays (at the cost of smaller noise margins). The basic idea is shown in Fig. 12.22. The reference voltage $V_{ref}$ is set to $VDD/2 + V_{THN}$. This has the effect of limiting the output voltage swing to a maximum of $VDD$ and a minimum of $VDD/2$. The main drawback of this logic implementation is the increased power dissipation resulting from the continuous power draw through the output leg at a voltage of $VDD/2$. The output leg at $VDD$ draws no DC power.

Microchip Ex. 1055, p. 271
Microchip v. Aptiv
IPR2024-00494

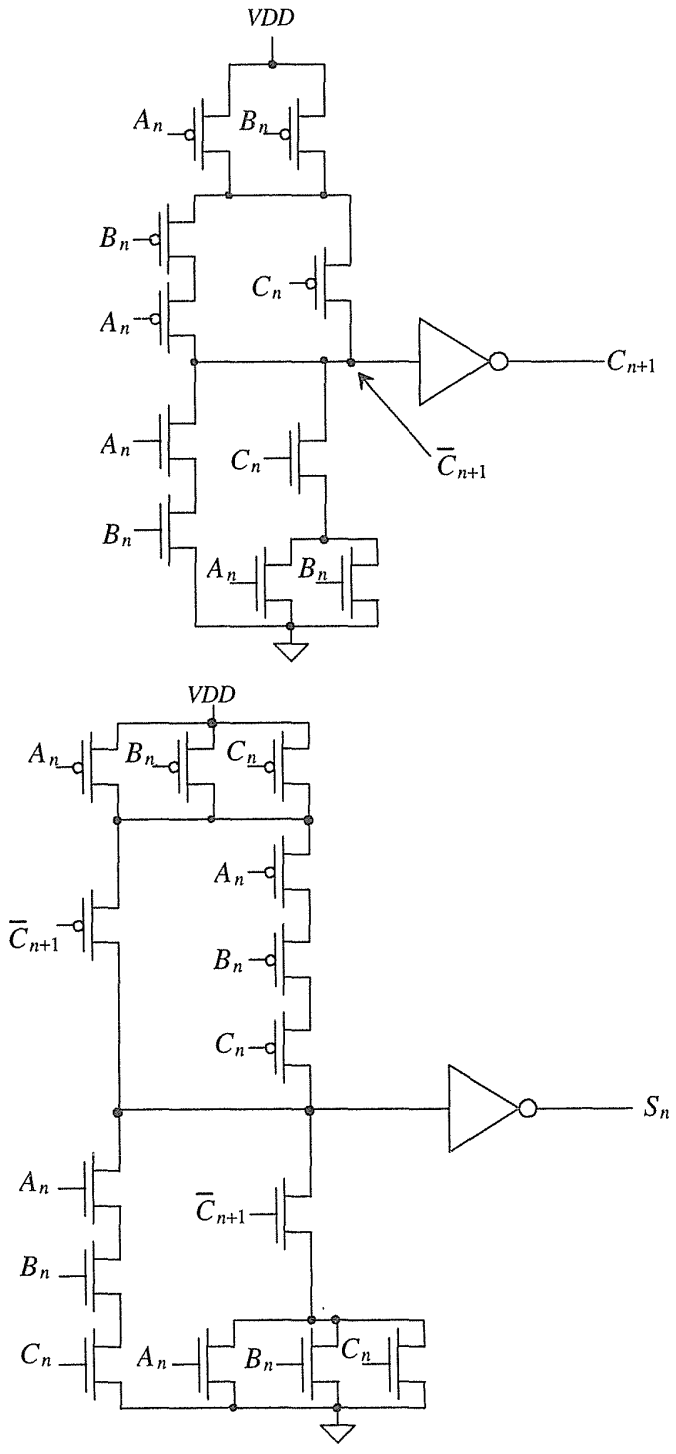



**Figure 12.18** AOI implementation of a full adder.

Microchip Ex. 1055, p. 272
Microchip v. Aptiv
IPR2024-00494

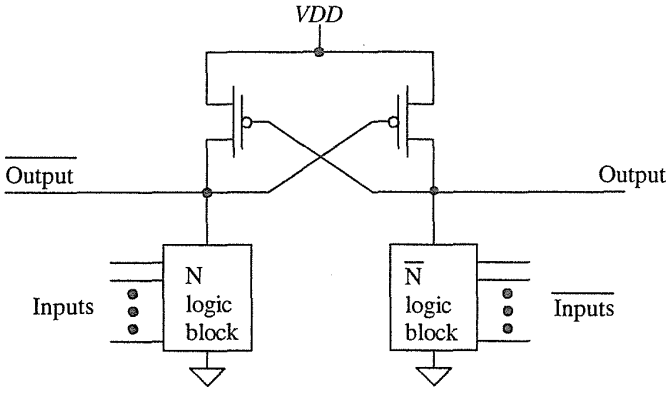



**Figure 12.19** CSVL block diagram.

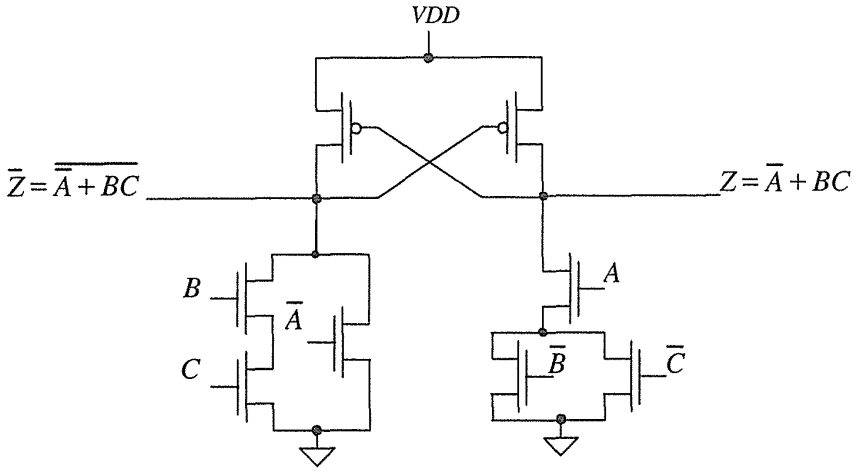



**Figure 12.20** CSVL logic gate.

Microchip Ex. 1055, p. 273
Microchip v. Aptiv
IPR2024-00494

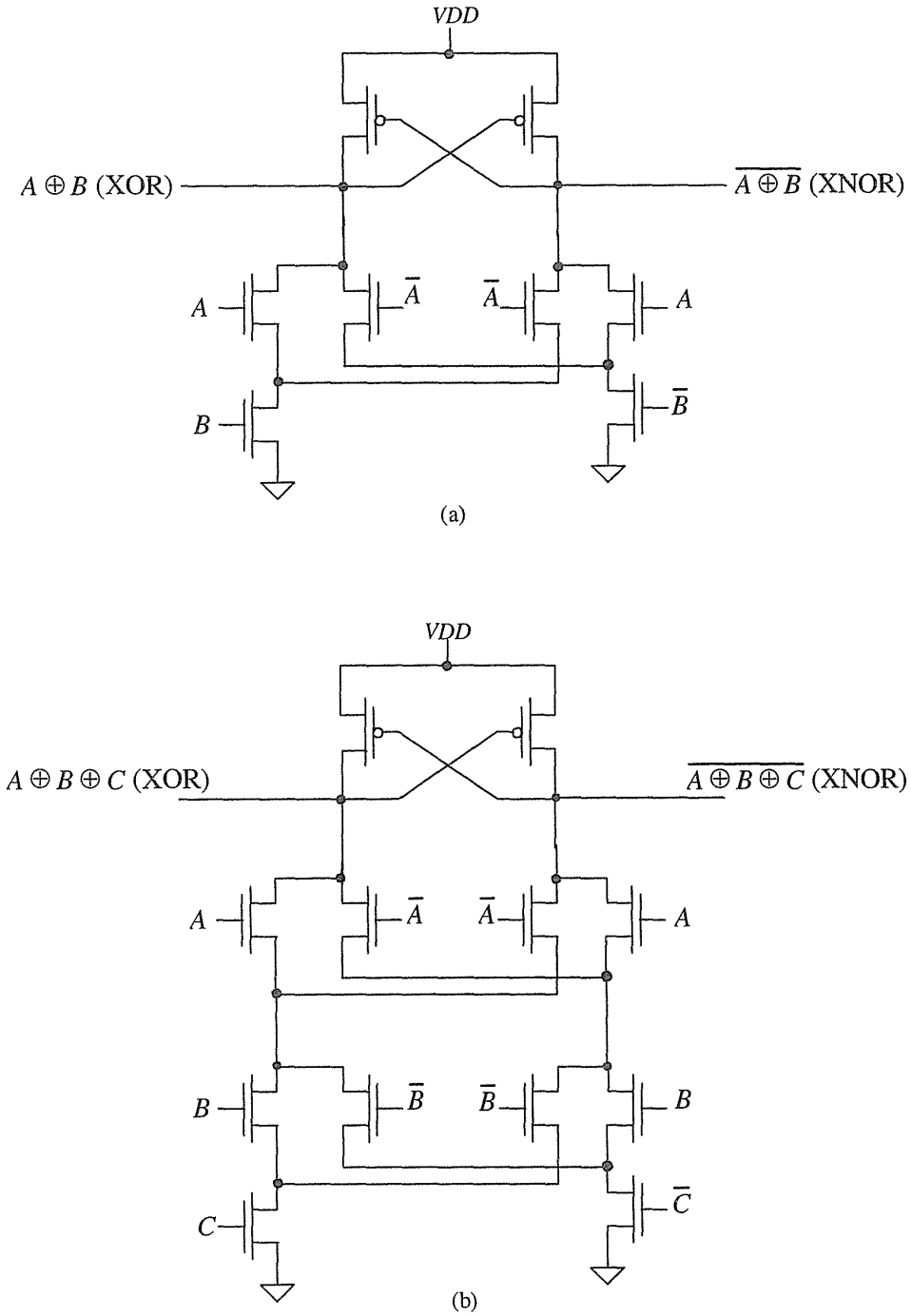



**Figure 12.21** (a) Two-input and (b) three-input XOR/XNOR gates.

Microchip Ex. 1055, p. 274
Microchip v. Aptiv
IPR2024-00494

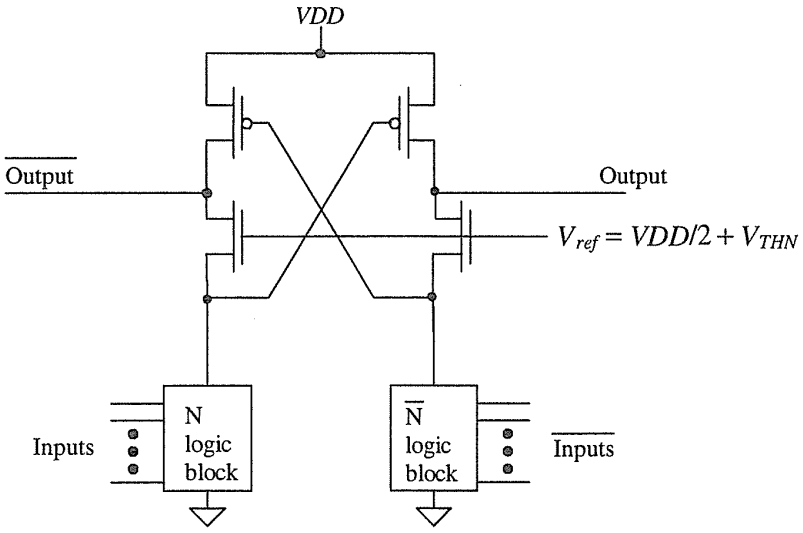



**Figure 12.22** DSL block diagram.

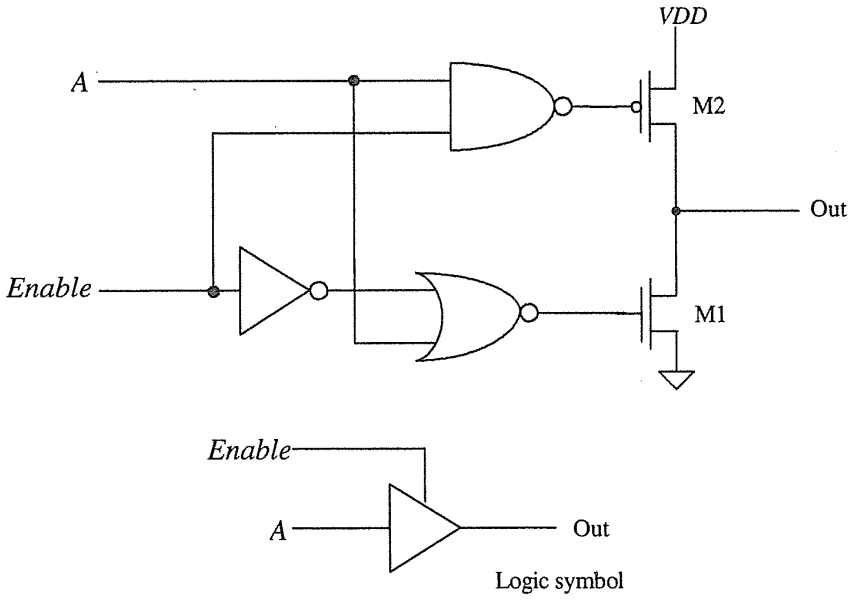



**Figure 12.23** Tri-state buffer.

Microchip Ex. 1055, p. 275
Microchip v. Aptiv
IPR2024-00494

### 12.4.3 Tri-State Outputs

A final example of a static logic gate, a tri-state buffer, is shown in Fig. 12.23. When the *Enable* input is high, the NAND and NOR gates invert and pass *A* (*VDD* or ground) to the gates of M1 and M2. Under these circumstances, M1 and M2 behave as an inverter. The combination of M1 and M2 with the inversion NAND/NOR gate causes the output to be the same polarity as *A*. When *Enable* is low, the gate of M1 is held at ground and the gate of M2 is held at *VDD*. This turns both M1 and M2 off. Under these circumstances, the output is said to be in the high-impedance or Hi-Z state. This circuit is preferable to the inverter circuits of Fig. 11.29 because only one switch is in series with the output to *VDD* or ground. An inverting buffer configuration is shown in Fig. 12.24.

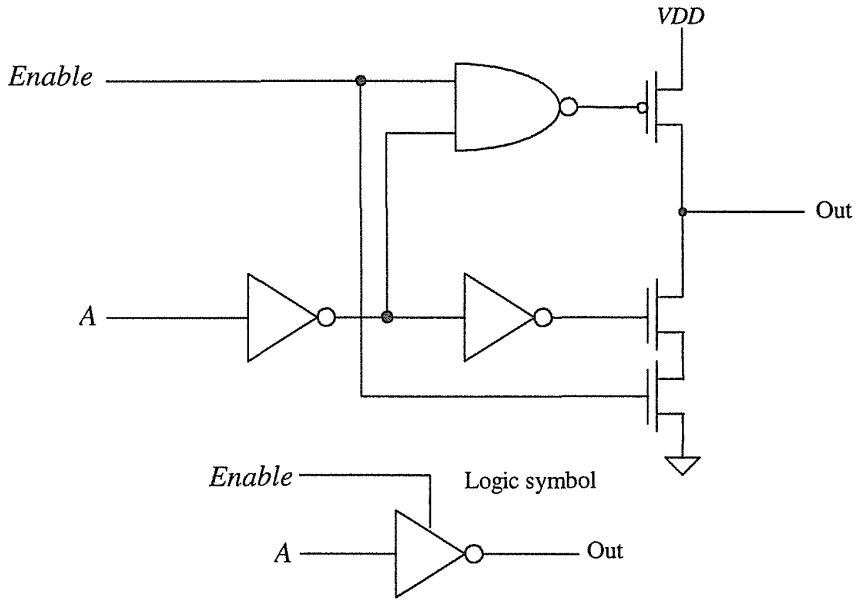



**Figure 12.24** Tri-state inverting buffer.

## REFERENCES

[1] M. I. Elmasry, *Digital MOS Integrated Circuits II,* IEEE Press, 1992. ISBN 0-87942-275-0, IEEE order number: PC0269-1.

[2] J. P. Uyemura, *Circuit Design for Digital CMOS VLSI,* Kluwer Academic Publishers, 1992.

[3] M. Shoji, *CMOS Digital Circuit Technology,* Prentice-Hall, 1988. ISBN 0-13-138850-9.

Microchip Ex. 1055, p. 276
Microchip v. Aptiv
IPR2024-00494

## PROBLEMS

Use the CN20 process unless otherwise specified.

**12.1** Design, lay out, and simulate the operation of a CMOS AND gate with a $V_{SP}$ of approximately 1.5 V. Use the standard-cell frame discussed in Ch. 4 for the layout.

**12.2** Design and simulate the operation of a CMOS AOI half adder circuit using static logic gates.

**12.3** Repeat Ex. 12.3 for a three-input NOR gate.

**12.4** Repeat Ex. 12.4 for a three-input NOR gate.

**12.5** Sketch the schematic of an OR gate with 20 inputs. Comment on your design.

**12.6** Sketch the schematic of a static logic gate that implements $\left(A + B \cdot \overline{C}\right) \cdot 1$. Estimate the worst-case delay through the gate when driving a 50 fF load capacitance.

**12.7** Design and simulate the operation of a CSVL OR gate made with minimum-size devices.

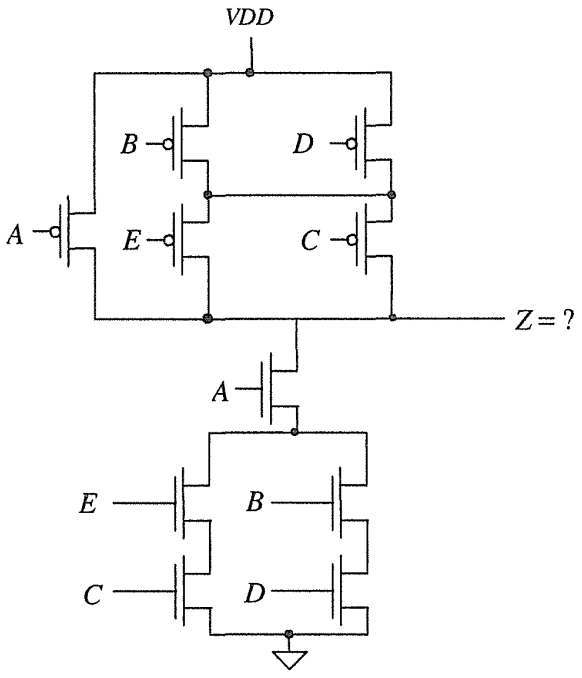



**Figure P12.10**

Microchip Ex. 1055, p. 277
Microchip v. Aptiv
IPR2024-00494

**12.8** Design a tri-state buffer that has propagation delays under 20 ns when driving a 1 pF load. Assume that the maximum input capacitance of the buffer is 100 fF.

**12.9** Sketch the schematic of a three-input XOR gate implemented in AOI logic.

**12.10** What logic function does the circuit of Fig. P12.10 implement?

**12.11** Calculate the switching point voltage of the gate shown in Fig. P12.11. What

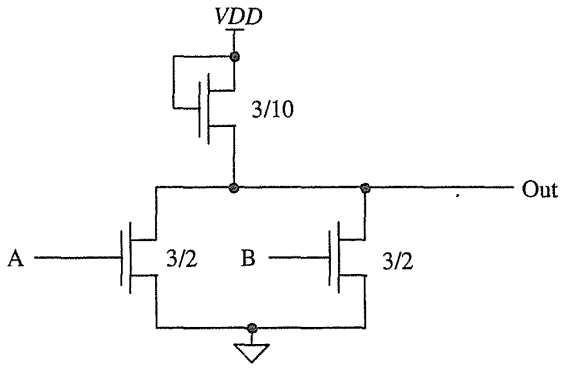



**Figure P12.11**

logic function does this circuit implement?

**12.12** Estimate the minimum and maximum output voltages for the gate of Fig. P12.11.

**12.13** The circuit shown in Fig. P12.13 is an edge-triggered one-shot that generates an output pulse, with width $t_d$, whenever the input makes a transition. Using inverters for delay elements, design and simulate the operation of a one-shot whose output pulse width is 10 ns. Comment on the resulting output if the width of the input pulse is less than $t_d$.

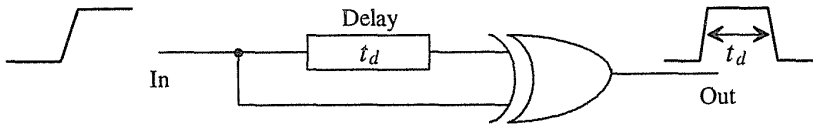



**Figure P12.13**

Microchip Ex. 1055, p. 278
Microchip v. Aptiv
IPR2024-00494

Chapter

13

# The TG and Flip-Flops

The transmission gate (TG) is used in digital CMOS circuit design to pass or not pass a signal. The schematic and logic symbol of the transmission gate (TG) are shown in Fig. 13.1. The gate is made up of the parallel connection of a p- and an n-channel MOSFET. Referring to the figure when $S$ (for select) is high we observe that the transmission gate passes the signal on the input to the output. The resistance between the input and the output can be estimated as $R_n \| R_p$. We begin this chapter with a description of the n- and p-channel pass transistor.

## 13.1 The Pass Transistor

Consider the single n-channel MOSFET shown in Fig. 13.2a. Assume that the voltage across the capacitor (the output of the pass transistor) is initially 5 V. When the gate (the select line) of the MOSFET is taken to *VDD*, the MOSFET turns on. In this situation, we can assume that the drain of the MOSFET is connected to the load capacitance and that the source (the input of the pass transistor) is connected to ground, keeping in mind that the drain and source are interchangeable. The delay-time of the capacitor discharging is simply

$$t_{PHL} = R_n C_{load} \tag{13.1}$$




**Figure 13.1** The transmission gate.

Microchip Ex. 1055, p. 279
Microchip v. Aptiv
IPR2024-00494

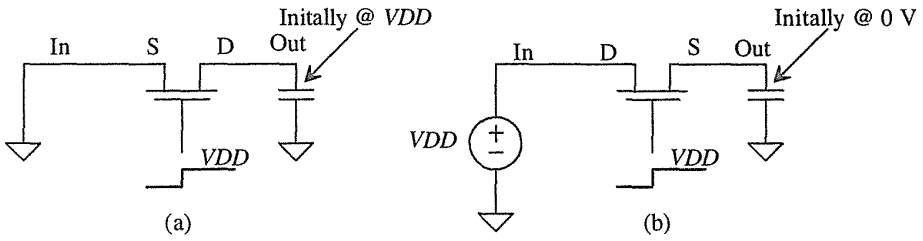



**Figure 13.2** An n-channel pass transistor showing transmission of 0 V and *VDD*.

Now consider Fig. 13.2b where the capacitor is initially at 0 V. In this case, the drain is connected to *VDD* and the source is connected to the load capacitance. Since the substrate, assumed at *VSS* = ground, is not at the same potential as the source, we have body effect present causing the threshold voltage to increase. When the gate of this MOSFET is raised to *VDD*, the load capacitor charges to $VDD - V_{THN}$ where $V_{THN}$, from Appendix A, is in the neighborhood of 1.5 V. Therefore, the low-to-high delay-time can be estimated by

$$t_{PLH} = R_n C_{load} \text{ for a high voltage of } VDD - V_{THN} \qquad (13.2)$$

In this derivation, we have neglected the parasitic capacitances of the MOSFET. The following example illustrates the switching behavior of the n-channel pass transistor.

**Example 13.1**

Estimate and simulate the delay through minimum-size n-channel pass transistors using the test setups of Fig. 13.2 driving a 100 fF load capacitance.

We know that for the minimum size ($W$ = 3 μm and $L$ = 2 μm) the n-channel effective resistance is 8 kΩ. Therefore, the propagation delays $t_{PHL} = t_{PLH}$ = 800 ps, remembering that the maximum high voltage is $VDD - V_{THN}$ or approximately 3.5 V. The simulation results are shown in Fig. 13.3. ■

A similar analysis of the p-channel MOSFET used as a pass transistor gives

$$t_{PHL} = R_p \cdot C_{load} \qquad (13.3)$$

and

$$t_{PLH} = R_p \cdot C_{load} \text{ for a low voltage of } V_{THP} \qquad (13.4)$$

The p-channel pass transistor can pass a logic high without signal loss, while passing a low results in a minimum low voltage of $V_{THP}$ (with body effect). The n-channel transistor can pass a logic low without signal loss, while passing a high results in a maximum high voltage of $VDD - V_{THN}$. One advantage of the p-channel pass transistor is that it can be laid out, in an n-well process, with the well tied to the source, eliminating body effect.

Microchip Ex. 1055, p. 280
Microchip v. Aptiv
IPR2024-00494

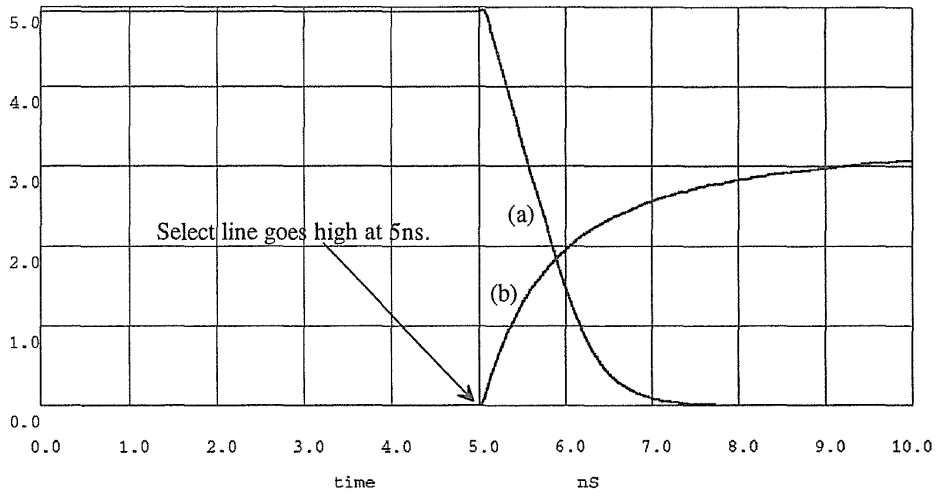



**Figure 13.3** Simulation results of an n-channel MOSFET driving a 100 fF load capacitance, (a) with output initially at 5 V passing 0 V, (b) with output initially at 0 V passing 5V.

The intrinsic propagation delays of the n- and p-channel pass transistors (no load capacitance) can be approximated by

$$t_{PHL}, t_{PLH} = R_n C'_{ox} WL = \tau_n \tag{13.5}$$

and

$$t_{PHL}, t_{PLH} = R_p C'_{ox} WL = \tau_p \tag{13.6}$$

The pass transistor turning on must discharge the charge stored on the output capacitance of the MOSFET through its own effective resistance.

## 13.2 The CMOS TG

Since the n-channel passes logic lows well and the p-channel passes logic highs well, putting the two complementary MOSFETs in parallel, as was shown in Fig. 13.1, results in a TG that passes both logic levels well. The CMOS TG requires two control signals, $S$ and $\overline{S}$ (see Fig. 13.4). The propagation delay-times of the CMOS TG are

$$t_{PHL} = t_{PLH} = (R_n || R_p) \cdot C_{loaa} \tag{13.7}$$

The capacitance on the $S$ input of the TG is the input capacitance of the n-channel MOSFET, or $C_{inn}$ (= $1.5C_{oxn}$). The capacitance on the $\overline{S}$ input of the TG is the input capacitance of the p-channel MOSFET, or $C_{inp}$. Making the widths of the MOSFETs used in the TG large reduces the propagation delay-times from the input to the output of the TG when driving a specific load capacitance. However, the delay-times in turning the TG on, the select lines going high, increase because of the increase in input capacitance. This should be remembered when simulating. Using a voltage source in

Microchip Ex. 1055, p. 281
Microchip v. Aptiv
IPR2024-00494

SPICE for the select lines, which can supply infinite current to charge the input capacitance of the TG, gives the designer a false sense that the delay through the TG is limited by $R_n$ and $R_p$. Often, when simulating logic of any kind, the SPICE-generated control signals are sent through a chain of inverters so that the control signals more closely match what will actually control the logic on die.

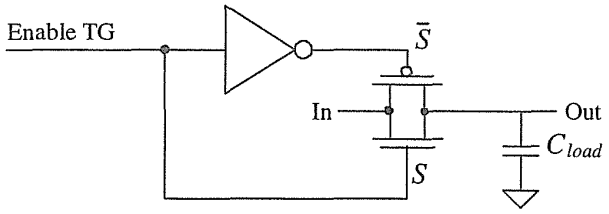



**Figure 13.4** The transmission gate with control signals shown.

**Example 13.2**

Estimate and simulate the delays through the TG shown in Fig. Ex13.2 when minimum-size MOSFETs are used and the load capacitance is 150 fF.

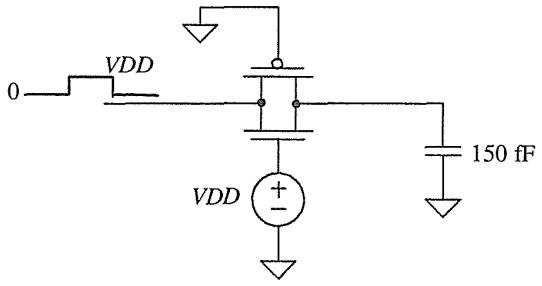



**Figure Ex13.2**

For minimum-size MOSFETs $R_n \| R_p$ = 6 kΩ so that $t_{PLH} = t_{PHL}$ = 900 ps. The SPICE simulation results are shown in Fig. 13.5. In this example we have applied the propagation delays defined by Eq. (13.7) somewhat differently. We are estimating how long it takes a change on the input of the TG to reach the output, at the 50 percent points, with the TG enabled. ■

### 13.2.1 Layout of the CMOS TG

Figure 13.6 shows the layout of the minimum-size CMOS transmission gate. Often, the inverter used to generate the complementary select signal (see Fig. 13.4) is added to the layout of the cell. The inverter allows the use of a single select signal which can be desirable in systems where both true and complement signals are not available. The n- and p-channel pass transistors can be laid out in a similar manner to the CMOS TG.

Microchip Ex. 1055, p. 282
Microchip v. Aptiv
IPR2024-00494

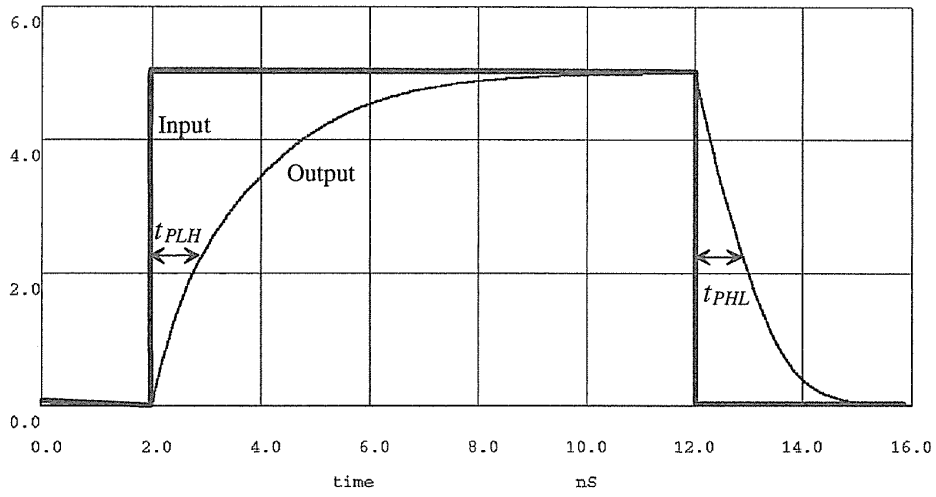



**Figure 13.5** Simulation results of CMOS minimum-size TG driving a 150 fF load.

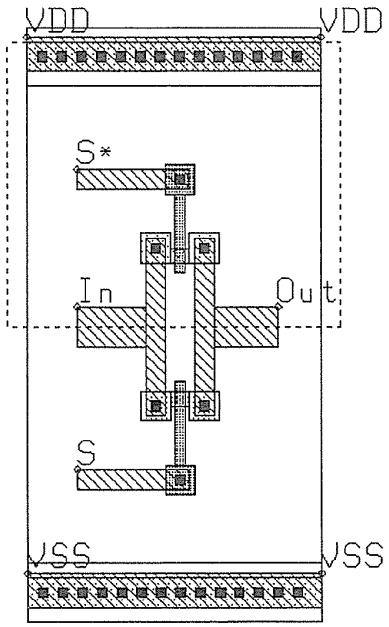



**Figure 13.6** Layout of the CMOS transmission gate.

Microchip Ex. 1055, p. 283
Microchip v. Aptiv
IPR2024-00494

### 13.2.2 Series Connection of Transmission Gates

Consider the series connection of CMOS transmission gates shown in Fig. 13.7. The equivalent digital model is also depicted in this figure. The output capacitance of the individual MOSFETs is not shown in this figure and will be neglected in the following analysis. The delay through the series connection can be estimated by

$$t_{PHL} = t_{PLH} = N \cdot (R_n||R_p)(C_{load}) + 0.35 \cdot (R_n||R_p)(C_{inn} + C_{inp})(N)^2 \quad (13.8)$$

The first term in this equation is simply the sum of the TG effective resistances, while the second term in the equation describes the RC transmission line effects.

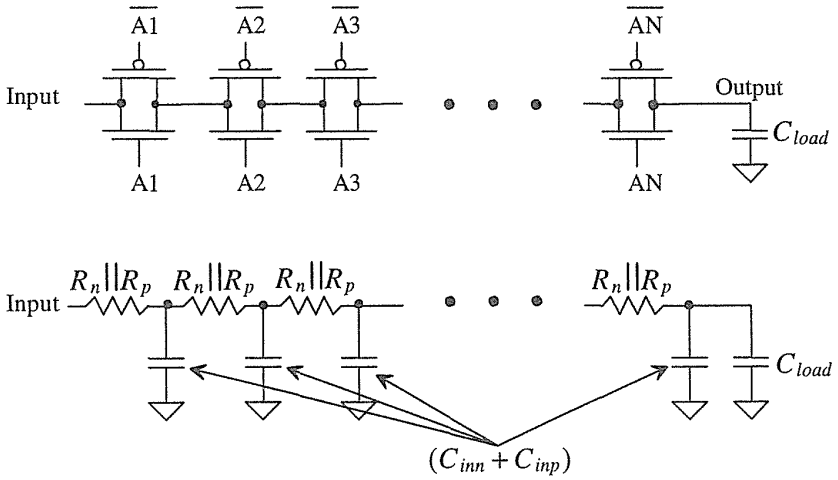



**Figure 13.7** Series connection of transmission gates with digital model.

## 13.3 Applications of the Transmission Gate

In this section, we present some of the applications of the TG [1, 2].

*Path Selector*

The circuit shown in Fig. 13.8 is a two-input path selector. Logically, the output of the circuit can be written as

$$Z = AS + B\bar{S} \quad (13.9)$$

When the selector signal $S$ is high, $A$ is passed to the output while a low on $S$ passes $B$ to the output.

This same idea can be used to implement multiplexers/demultiplexers (MUX/DEMUX). Consider the block diagrams of a MUX and DEMUX shown in Fig. 13.9. The number of control lines is related to the number of input lines by

Microchip Ex. 1055, p. 284
Microchip v. Aptiv
IPR2024-00494

$$2^m = n \tag{13.10}$$

where $n$ is the number of inputs (outputs) to the MUX (DEMUX) and $m$ is the number of control lines. A 4 to 1 MUX/DEMUX is shown in Fig. 13.10. Note that the MUX is bidirectional; that is, it can be used as a MUX or a DEMUX. The logic equation describing the operation of the MUX is given by

$$Z = A(S1 \cdot S2) + B(S1 \cdot \overline{S2}) + C(\overline{S1} \cdot S2) + D(\overline{S1} \cdot \overline{S2}) \tag{13.11}$$

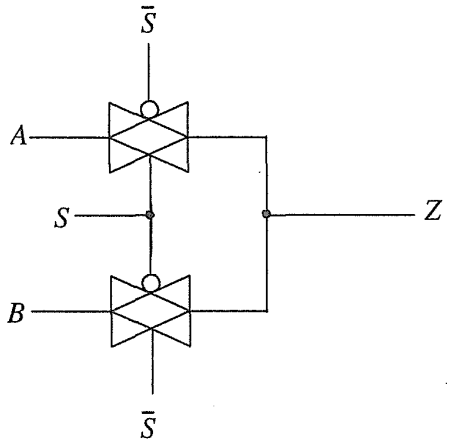



**Figure 13.8** Path selector.

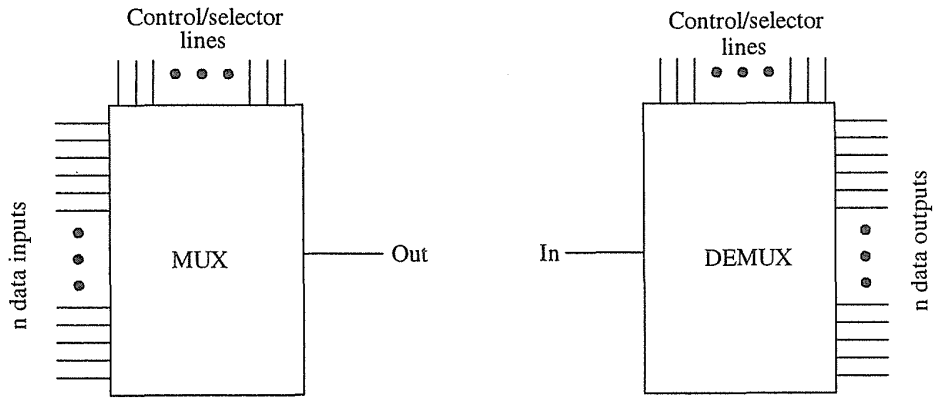
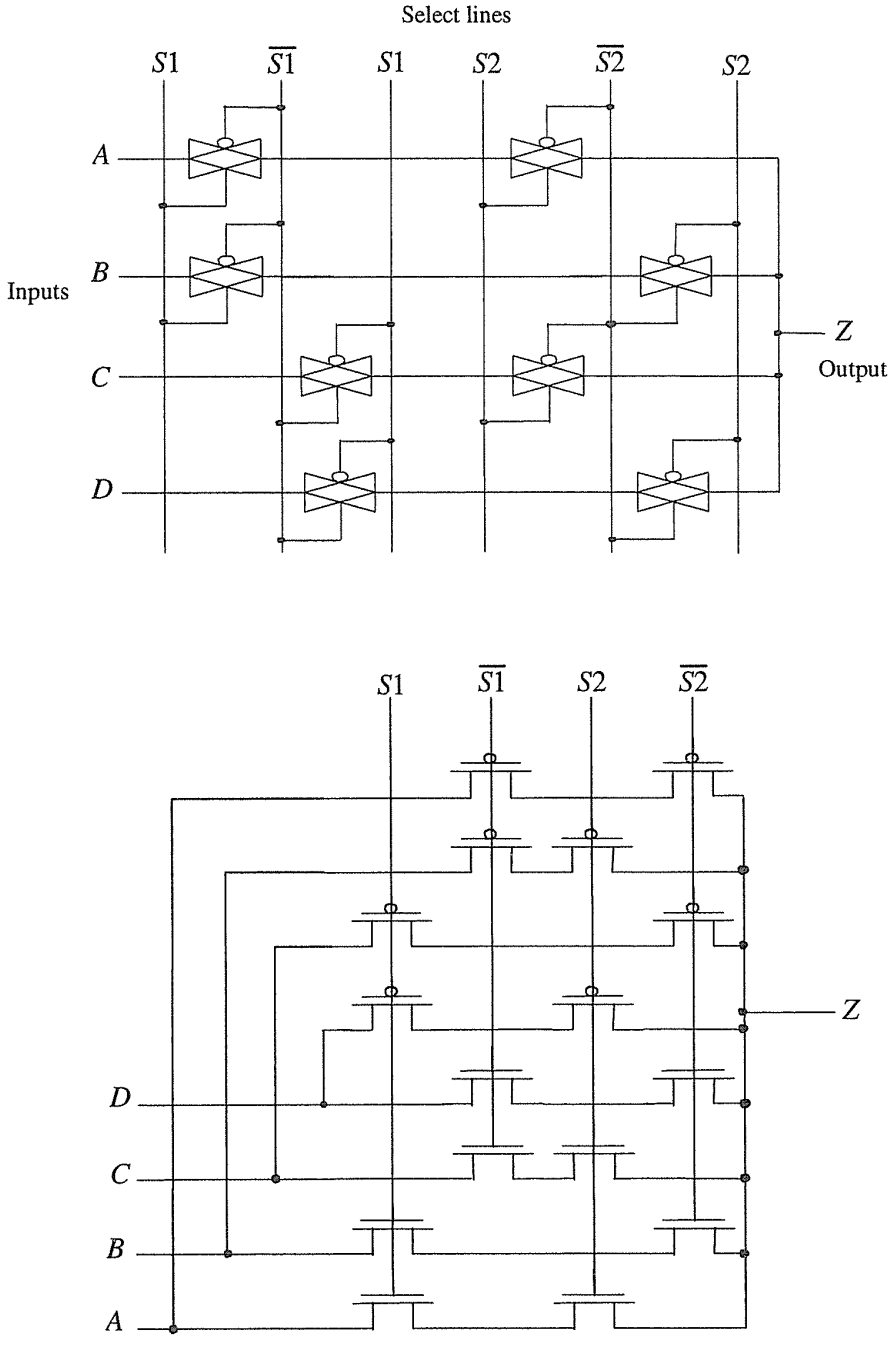



**Figure 13.9** Block diagram of MUX/DEMUX.

Microchip Ex. 1055, p. 285
Microchip v. Aptiv
IPR2024-00494




**Figuer 13.10** Circuit implemetations of a 4 to 1 MUX/DEMUX.

Microchip Ex. 1055, p. 286
Microchip v. Aptiv
IPR2024-00494

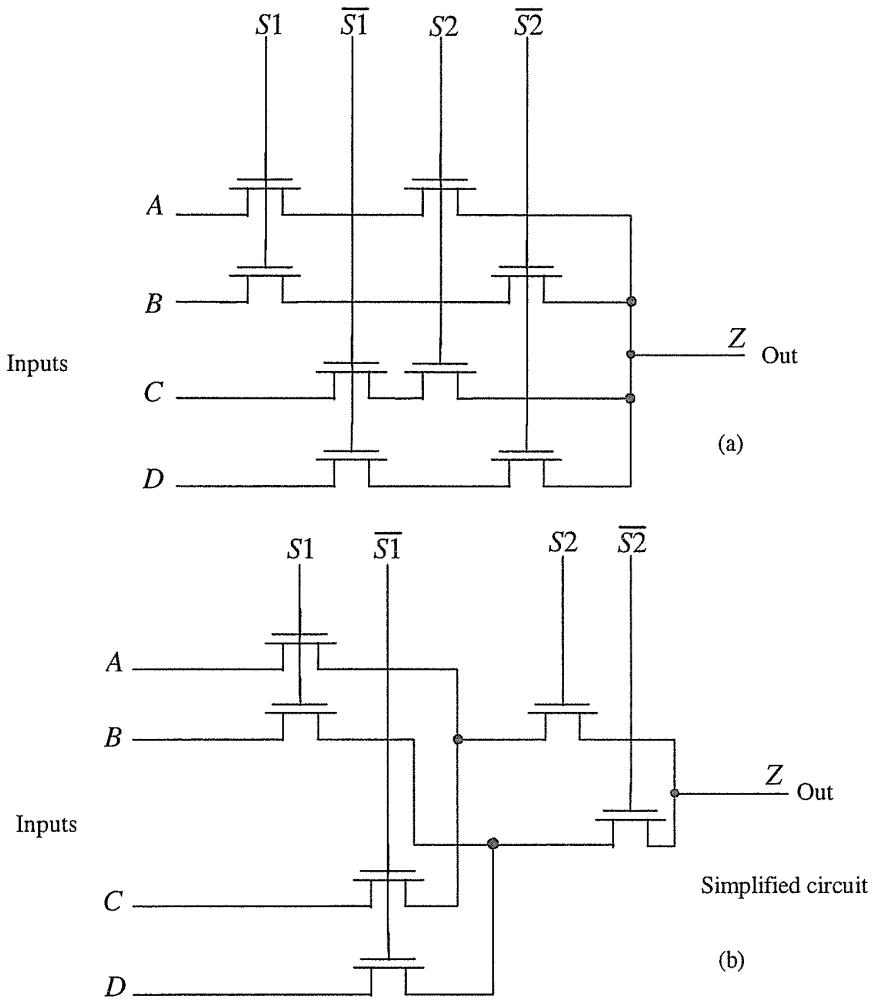



**Figure 13.11** MUX/DEMUX using pass transistors.

Figure 13.11a shows a pass transistor implementation of the 4 to 1 MUX. The pass transistor implementation is simpler, using fewer transistors, at the price of a threshold voltage drop from input to output when the input is a high (*VDD*). A simplified version of the circuit of Fig. 13.11a is shown in Fig. 13.11b. Here the MOSFETs connected to *S2* and $\overline{S2}$ are combined to reduce the total number of MOSFETs used. The reduction of the total number of MOSFETs used can be extended to an n-input (output) MUX (DEMUX). Again, it should be remembered that a DEMUX can be formed using the circuits of Figs. 13.10 or 13.11 by switching the inputs with the outputs.

Microchip Ex. 1055, p. 287
Microchip v. Aptiv
IPR2024-00494

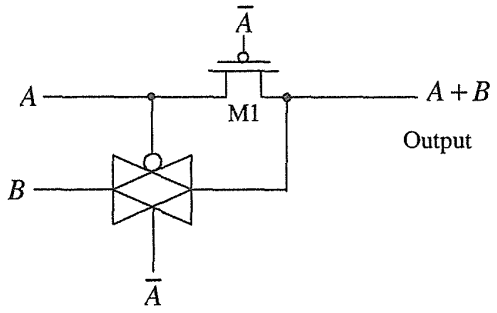



**Figure 13.12** TG-based OR gate.

*Static Gates*

The TG can be used to form static logic gates. Consider the OR gate shown in Fig. 13.12. To understand the operation of the gate, consider the case when both $A$ and $B$ are low. Under these circumstances the pass transistor, M1, and the TG are on. The inputs $A$ and $B$, both low, are passed to the output. If $A$ is high, M1 is on and $A$ is passed to the output. If $B$ is high and $A$ is low, $B$ is passed to the output through the TG. If both $A$ and $B$ are high, the TG is off and M1 is on passing $A$, a high, to the output.

Figure 13.13 shows an XOR and an XNOR gate made using TGs. Consider the XOR gate with both $A$ and $B$ low. Under these circumstances, the top TG is on and its output is connected to $A$, a low. If both inputs are high, the bottom TG connects the output to $\overline{A}$, again a low. If $A$ is high and $B$ is low, the top TG is on and the output is connected to $A$, a high. Similarly, if $A$ is low and $B$ is high, the bottom TG is on and connects the output to $\overline{A}$, a high.

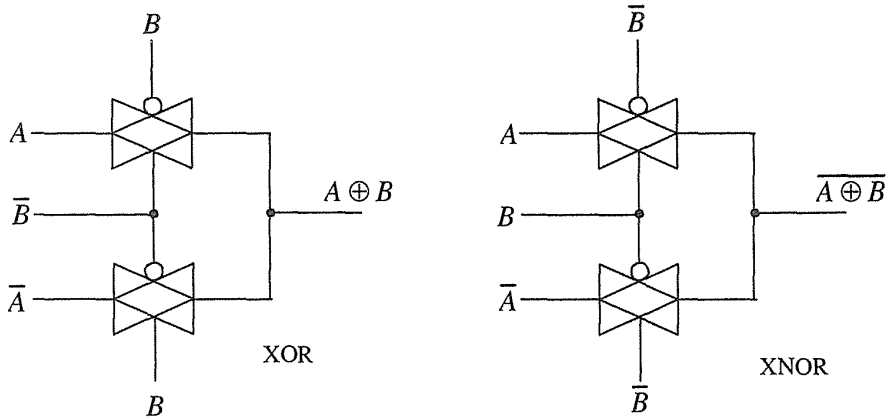
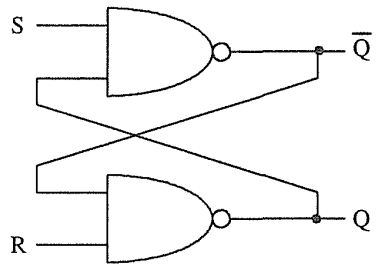



**Figure 13.13** TG implementation of XOR/XNOR gate.

Microchip Ex. 1055, p. 288
Microchip v. Aptiv
IPR2024-00494

## 13.4 The Flip-Flop

Consider the set-reset flip-flop (SR FF) shown in Fig. 13.14 made using NAND gates. The logic symbol and truth table are also shown in this figure. Consider the case when $S$ is high and $R$ is low. Forcing $R$ low causes $Q$ to go high. Since $S$ is high and $Q$ is high, the $\overline{Q}$ output is low. Now consider the case when both $S$ and $R$ are low. Under these circumstances, the FF outputs are both high. This FF can easily be designed and laid out with the techniques of Ch. 12.

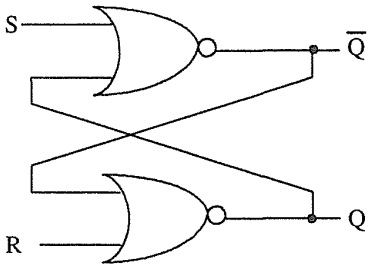



Truth table

| $S$ | $R$ | $Q$ | $\overline{Q}$ |
|---|---|---|---|
| 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 1 | $Q$ | $\overline{Q}$ |

**Figure 13.14** Set-reset flip-flop made using NAND gates.

An alternative implementation of the SR flip-flop is shown in Fig. 13.15 using NOR gates. Consider the case when $S$ is high and $R$ is low. For the NOR gate, a high input forces the output of the gate low. Therefore, the $\overline{Q}$ output is low whenever the $S$ input is high. Similarly, whenever the $R$ input is high, the $Q$ output must be low. The case of both inputs being high causes both $Q$ and $\overline{Q}$ to go low, or in other words the outputs of the FF are no longer complements. Figure 13.16 shows the logic symbol of the SR flip-flop. Note that the true and complement locations on the logic symbol are switched with the location of Figs. 13.14 and 13.15.




Truth table

| $S$ | $R$ | $Q$ | $\overline{Q}$ |
|---|---|---|---|
| 0 | 0 | $Q$ | $\overline{Q}$ |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 |

**Figure 13.15** Set-reset flip-flop made using NOR gates.

Microchip Ex. 1055, p. 289
Microchip v. Aptiv
IPR2024-00494

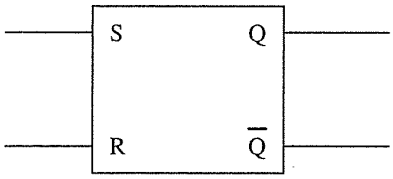



**Figure 13.16** Logic symbol of the SR flip-flop.

### 13.4.1 Clocked Flip-Flops

Clocked flip-flops can be divided into three categories. The first category consists of those in which the clock signal pulse width must be short compared to the propagation delay through the FF. In other words, the clock input should go high and then low before the output of the flip-flop changes state. The second category of clocked FFs consists of those in which the output changes while the clock signal is high. This type of FF is sometimes called a level-sensitive FF. The final category of FF is the edge-triggered type. The output of the FF changes state on the rising or falling edge of the FF.

*Short Clock Pulse Widths*

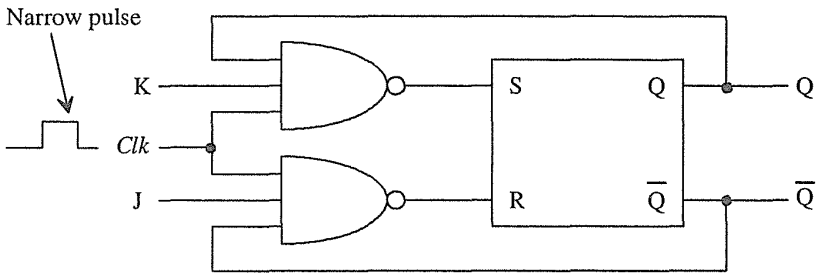
The clocked JK FF is shown in Fig. 13.17. The JK FF is constructed using the NAND SR FF and two NAND gates. However, unlike the SR FF, both J and K can be high at the same time without the outputs becoming equal. The operation of the JK FF depends on the previous state of the FF. With the clock signal held low, the inputs and outputs of the SR FF do not change. Holding the clock signal high causes the output of the FF to oscillate between a logic 0 and 1. With application of a short clock pulse and J = K = 0, the outputs of the FF do not change. If K = 1 and J = 0, the output, Q, is 0 after application of the clock pulse, while if, J = 1 and K = 0 the output is set high. If both J and K are high, the output becomes the complement of the previous state.

A toggle or T flip-flop can be constructed using the JK FF by setting J = K = 1, or simply by replacing the three-input NAND gates of Fig. 13.17 with two-input NAND gates. The clock input of the JK FF is used as the T input of the T FF. Application of a short pulse to T input of the T FF causes the output, Q, to toggle states. If the output is a high and the T input goes high, the output changes to a low. Pulsing the T input high again causes the output to change back to a high. The T FF can be used to divide a clock signal in half, keeping in mind the pulse width limit (when using the FF of Fig. 13.17).

*Level-Sensitive Flip-Flops*

Flip-flops that are clocked with a signal that enables the output to change with the input have no specific pulse width requirements for the clock. To illustrate a level-sensitive

Microchip Ex. 1055, p. 290
Microchip v. Aptiv
IPR2024-00494




Truth table

| $J$ | $K$ | $Q_{n+}$ |
|---|---|---|
| 0 | 0 | $Q_n$ |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q}_n$ |

**Figure 13.17** Clocked JK FF. Clock pulse width should be short compared to FF delay.

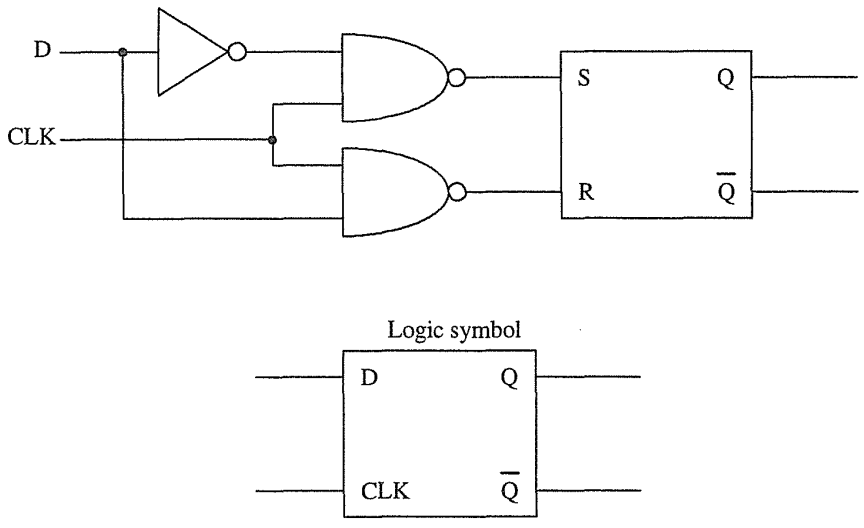



**Figure 13.18** Level-sensitive D flip-flop.

Microchip Ex. 1055, p. 291
Microchip v. Aptiv
IPR2024-00494

FF, consider the data or D FF shown in Fig. 13.18 with associated logic symbol. When the clock signal is high, the D input can pass directly to the SR FF. If D is a 1 while CLK is high, the output, Q, is a 1, while if D is low the output is a low. If D changes at any time while the CLK input is high, the output will follow. When the CLK signal goes low, the current logic level of D is latched into the SR FF. Note that this FF is not an edge-sensitive FF because the output changes at other times than the edge transition time.

*Edge-Triggered Flip-Flops*

The JK master-slave FF, shown in Fig. 13.19, is an example of an edge-triggered FF. When the CLK signal goes high, the master JK FF is enabled. Since the slave FF cannot change states when CLK is high, the clock pulse width does not have to be less than the propagation delay of the FF. When CLK goes low, the master data are transferred to the slave. If both J and K are low, the output of the master remains unchanged, and therefore so does the output of the slave. If J = 1 and K= 0 when the CLK pulse goes low, the master output, Q, goes high. When the CLK goes low, the high output of the master is transferred to the slave. The master-slave JK FF behaves just like the JK FF of the previous section except for the fact that the data are not available until CLK goes low and there is no restriction on the pulse width of the clock (i.e., the FF is falling edge triggered). Adding reset or set capability to the FF can be accomplished by adding logic gates between the NAND and the SR FF of Fig. 13.19. The logic gates simply ensure that the SR FF are placed into a certain state upon application of a reset or set signal.

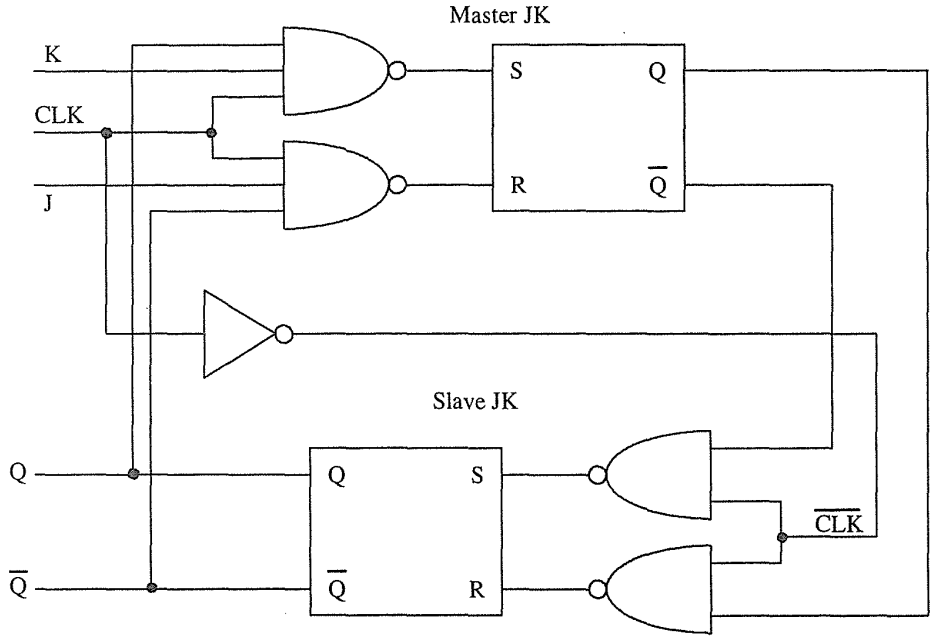



**Figure 13.19** Edge-triggered JK master-slave flip-flop.

Microchip Ex. 1055, p. 292
Microchip v. Aptiv
IPR2024-00494

An implementation of the positive edge-triggered D FF is shown in Fig. 13.20a. The SR FF is made using NAND gates. When the CLK input is low, the outputs of the NAND gates are both high, keeping the SR FF in the "no change mode." When CLK goes high, the logic value on the D input of the FF is transferred to the S input and the complement is transferred to the R input of the SR FF. The $\overline{CLK}$ input of the NAND gates goes low three inverter delays after CLK goes high. This forces both R and S high, putting the flip-flop in the no change mode. The only time that CLK and $\overline{CLK}$ can be high, the condition required to transfer D to the input of the FF, is the time between CLK going high and $\overline{CLK}$ going low. This time is determined by the propagation delay of the inverters. In practice a single inverter, in place of the three, will not provide a sufficient delay to allow the inputs of the SR FF to fully charge to D and $\overline{D}$. Also, there are minimum rise- and falltimes requirements for the clock.

Another implementation of the positive edge-triggered D FF using transmission gates is shown in Fig. 13.20b. When the CLK input is low, the logic value at D is setting at node A and $\overline{D}$ is on node B. Transmission gates T2 and T3 are off. The datum on node C is available on the output of the FF and is the result of the previous leading edge transition of the CLK input pulse. When CLK goes high, T1 and T4 turn off, while T2 and T3 turn on and the datum on node is C is transferred, with the appropriate inversion to the outputs. A D FF with set and clear inputs is shown in Fig. 13.20c.

*Flip-Flop Timing*

The data must be set up or present on the D input of the FF (see Fig. 13.20c) a certain time before we apply the clock signal. This time is defined as the setup time of the FF. To understand the origin of this time, consider the time it takes the signal at D to propagate through T1 and the NAND gate to point B. Before the clock pulse can be applied, the logic level D must be settled on point B. Consider the waveforms of Fig. 13.21. The time between D going high (or low) and the clock rising edge is termed the setup time of the FF and is labeled, $t_s$.

The wanted D input must be applied $t_s$ before the clock pulse is applied. Now the question becomes "How long does the wanted D input have to remain on the input of the FF after the clock pulse is applied?" This time called the hold time, $t_h$, is illustrated in Fig. 13.22. Shown in this figure, $t_h$ is a positive number. However, inspection of Fig. 13.20 shows that if the D input is removed slightly before the clock pulse is applied, the point B will remain unchanged because of the propagation delay from D to B. Analysis of this FF would yield a negative hold time. In other words, for the point B to charge to D, a time labeled $t_s$ is needed. Once point B is charged, the D input can be changed as long as the clock signal occurs within $t_h$.

One final important comment regarding the clock input of a FF is in order. If the clock input risetime is slow, the FF will not function properly. There will not be an abrupt transition between the sets of transmission gates turning on and off. The result will be logic levels at indeterminate states. What is usually done to eliminate this

Microchip Ex. 1055, p. 293
Microchip v. Aptiv
IPR2024-00494

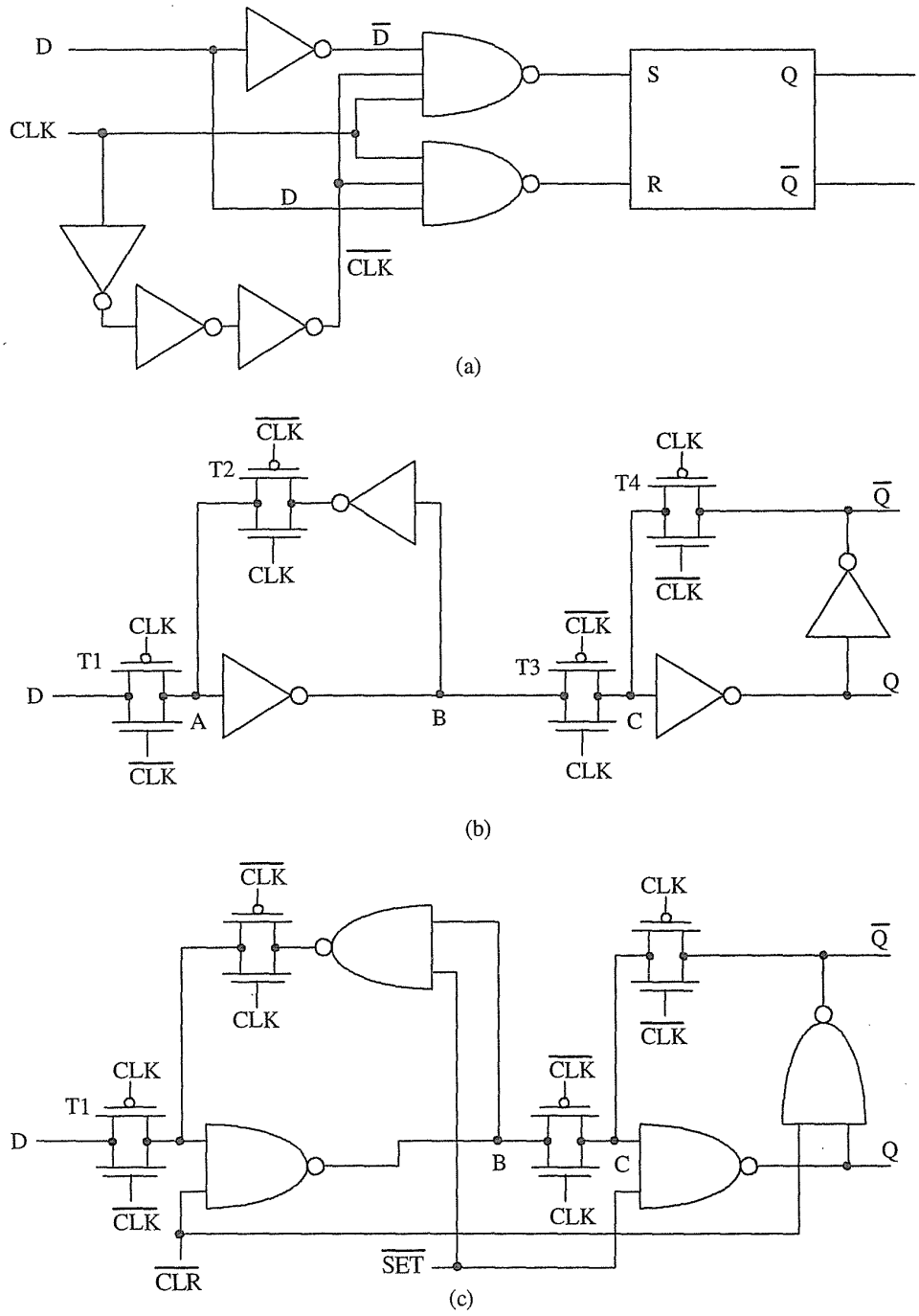



**Figure 13.20** Edge-triggered D flip-flops, (a) Gate implementation, (b) TG implementation, and (c) TG implementation with set and clear.

Microchip Ex. 1055, p. 294
Microchip v. Aptiv
IPR2024-00494

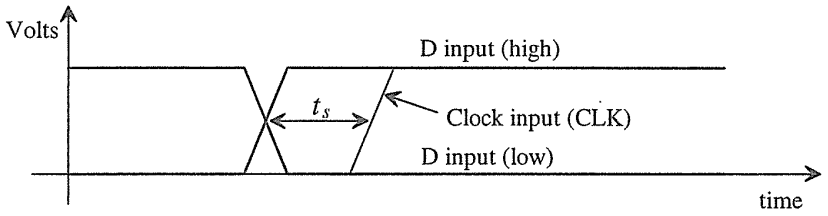



**Figure 13.21** Illustrating D FF setup time.

problem is to buffer the clock input through several inverters. This has the effect of speeding up the leading and trailing edges of a slow input pulse and presenting a lower input capacitance on the clock input to whatever is driving the FF. The main disadvantage is the increase in delay times, $t_{PHL}$ and $t_{PLH}$ (defined by clock to output), of the FF. In general, the FFs of Fig 13.20b and c should not be laid out without buffering the clock inputs.

The minimum pulse width of the clock, set, or clear inputs is labeled $t_w$. The minimum width is determined by the delay through (referring to Fig. 13.20) two NAND gates and a TG. The last timing definition we will consider here is the recovery time, that is, the time between removing the set or clear inputs and a valid clock input. This variable is labeled $t_{rec}$.

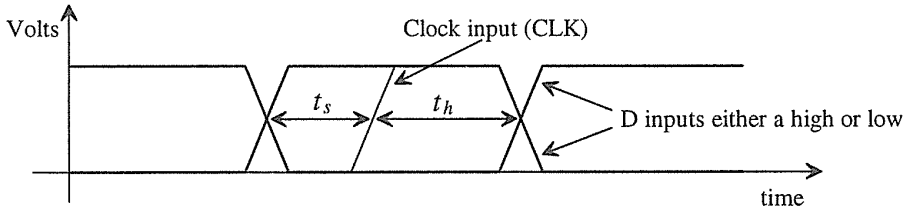



**Figure 13.22** Illustrating D FF hold time.

*Simple D Flip-Flop*

A simple D flip-flop using inverters and TGs is shown in Fig. 13.23. The cross-coupled connection of inverters is sometimes referred to as a latch and is the basis for the static RAM storage cell discussed further in Ch. 17. To understand the operation of this circuit, consider the case when CLK is low. The TGs are off, and the outputs do not change from their previous state. When CLK is high, provided the inverters are sized correctly (more on this shortly), the $D$ input is connected to $Q$ and the $\overline{D}$ input is connected to $\overline{Q}$. Thus, when CLK goes back low, the value of $D$ is remembered and latched. A couple of points should be made regarding this flip-flop: (1) the outputs change with the inputs whenever CLK is high, that is, it is not an edge-triggered flip-flop, and (2) the inputs must supply a current during switching.

Microchip Ex. 1055, p. 295
Microchip v. Aptiv
IPR2024-00494

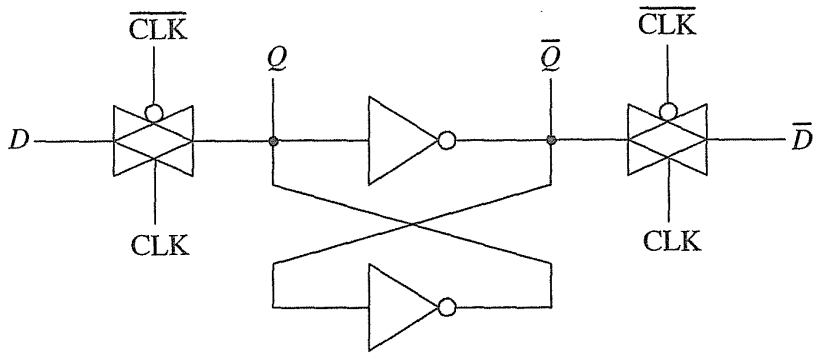



**Figure 13.23** Clocked D flip-flop using the basic latch and TGs.

The input DC current comes from the fact that the output of an inverter is connected to each TG. In order to change the voltage at $Q$ and $\overline{Q}$, with the inputs, the effective digital resistances of the inverters should be large compared to the sum of the TG resistance and the driver resistance. (The driver resistance is the effective resistance of whatever gate is driving the TG.) In other words, the $R_n$ and $R_p$ of the inverter should be large. The length of the devices used in the inverters can be longer than the minimum length to reduce the input current.

## REFERENCES

[1] J. P. Uyemura, *Circuit Design for Digital CMOS VLSI,* Kluwer Academic Publishers, 1992.

[2] M. I. Elmasry, *Digital MOS Integrated Circuits II,* IEEE Press, 1992. ISBN 0-87942-275-0, IEEE order number: PC0269-1.

[3] M. Shoji, *CMOS Digital Circuit Technology,* Prentice-Hall, 1988. ISBN 0-13-138850-9.

## PROBLEMS

Unless otherwise stated, use the CN20 process.

**13.1** Verify the simulation results shown in Fig. 13.3. If we increase the width of the n-channel pass transistor, what happens to the delay-times? The gate of the pass transistor is driven from some other logic on the chip. What happens to the capacitance seen by this logic when we increase the width of the pass transistor?

**13.2** Design and simulate the operation of a half adder circuit using TGs.

**13.3** Estimate and simulate the delay through 10 TGs (assume minimum-size) connected to a 100 fF load capacitance.

**13.4** Sketch the schematic of an 8 to 1 DEMUX using n-channel pass transistors. Estimate the delay through the DEMUX when the output is connected to a 50 fF load capacitance.

Microchip Ex. 1055, p. 296
Microchip v. Aptiv
IPR2024-00494

**13.5** Verify, using SPICE, that the circuit of Fig. 13.13 operates as an XOR gate.

**13.6** Simulate the operation of an SR FF made with NAND gates using minimum-size MOSFETs. Show all four logic transitions possible for the FF.

**13.7** Simulate the operation of the clocked D FF of Fig. 13.23b using minimum-size MOSFETs. Comment on any glitches you encounter. Show the FF clocking in a logic 1 and 0. What are the setup and hold times for your design?

**13.8** Design and simulate the operation of the FF shown in Fig. P13.8.

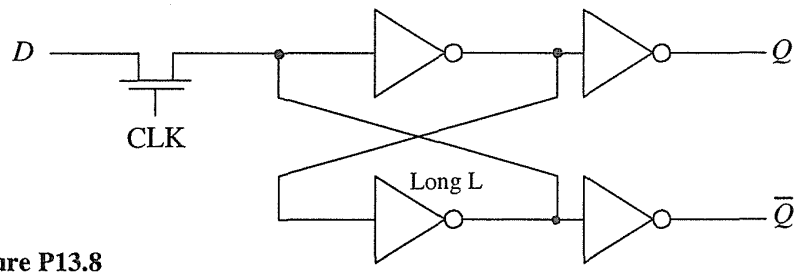



**Figure P13.8**

**13.9** The FF shown in Fig. P13.8 has several practical problems, including not presenting a purely capacitive load at the D-input and large layout size. The FF of Fig. P13.9 is a different implementation of an inverter-based latch which does present a purely capacitive load to the D-input and a (possibly) smaller layout size. Simulate the operation of this FF using the device sizes shown.

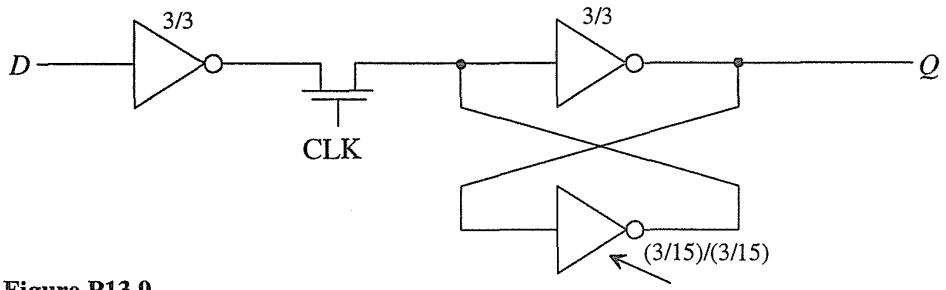



**Figure P13.9**

Long L (both n- and p-channel) inverter.

**13.10** Repeat Ex. 13.1 using minimum-size (0.9/0.6) MOSFETs using the CMOS14TB process.

**13.11** Repeat Ex. 13.2 using the CMOS14TB process.

Microchip Ex. 1055, p. 297
Microchip v. Aptiv
IPR2024-00494

**13.12** Estimate and simulate the delay through 10 TGs (assume minimum-size) connected to a 100 fF load capacitance using the CMOS14TB process.

**13.13** Using a SPICE DC sweep, plot the output voltage against the input voltage for the circuit of Fig. P13.13 with the input varying from 0 to 5 V and then from 5 V to 0. Comment on the difference in the plots.

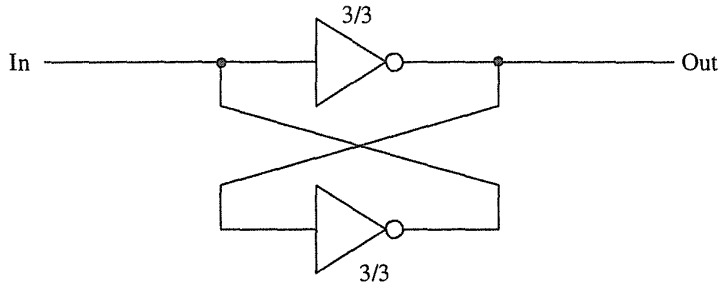



**Figure P13.13**

Microchip Ex. 1055, p. 298
Microchip v. Aptiv
IPR2024-00494

Chapter

14

# Dynamic Logic Gates

Dynamic or clocked logic gates are used to decrease complexity, increase speed, and lower power dissipation. The basic idea behind dynamic logic is to use the capacitive input of the MOSFET to store a charge and thus remember a logic level for use later. Before we start looking into the design of dynamic logic gates, let's discuss leakage current and the design of clock circuits.

## 14.1 Fundamentals of Dynamic Logic

Consider the n-channel pass transistor shown in Fig. 14.1 driving an inverter. If we clock the gate of the pass transistor high, the logic level on the input, point A, will be passed to the input of the inverter, point B. If this logic level is a "0," the input of the inverter will be forced to ground while a logic "1" will force the input of the inverter to $VDD - V_{THN}$. When the clock signal goes low, the pass transistor shuts off and the input to the inverter "remembers" the logic level. In other words, when the pass transistor turns on, the input capacitance of the inverter is charged to $VDD - V_{THN}$, or ground, through the pass transistor. As long as this charge is present, the logic value is remembered. What we are concerned with at this point is the leakage mechanisms present which can leak the stored charge off the node. A node, such as the one labeled B in Fig. 14.1, is called a dynamic node or a storage node. Note that this node is a high-impedance node and is easily susceptible to noise (see Ex. 3.4).

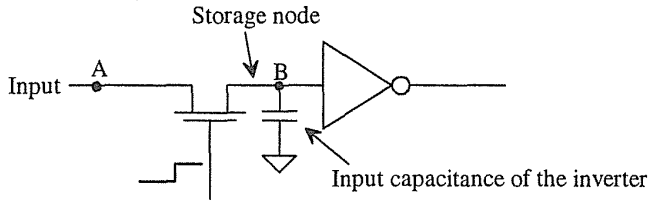



**Figure 14.1** Example of a dynamic circuit and associated storage capacitance.

Microchip Ex. 1055, p. 299
Microchip v. Aptiv
IPR2024-00494

### 14.1.1 Charge Leakage

Consider the expanded view of the charge storage node shown in Fig. 14.2. Practically, the only leakage path on this node is through the MOSFET's drain (or source since the drain and source are interchangeable) n+ /p-substrate diode. If we consider this node the drain of the MOSFET, the current is given by

$$I_D = I_{leakage} = I_S(e^{-V_B/nV_T} - 1) \tag{14.1}$$

where $V_B$ is the voltage on the storage node to ground, assuming the substrate is at ground potential. From the BSIM model parameters, the scale current is given by

$$I_S = AD \cdot JS \tag{14.2}$$

In order to simplify hand calculations we will assume that the leakage current is equal to the scale current, or

$$I_{leakage} = I_S = AD \cdot JS \tag{14.3}$$

The rate at which the storage node discharges is given by

$$\frac{dV}{dt} = \frac{I_{leakage}}{C_{node}} = \frac{AD \cdot JS}{C_{node}} \tag{14.4}$$

The node capacitance is the sum of the input capacitance of the inverter, the capacitance to ground of the metal or poly line connecting the inverter to the pass transistor, and the capacitance of the drain implant to substrate (the depletion capacitance). For practical applications, we assume that

$$C_{node} \approx C_{in} \text{ of the inverter} \tag{14.5}$$

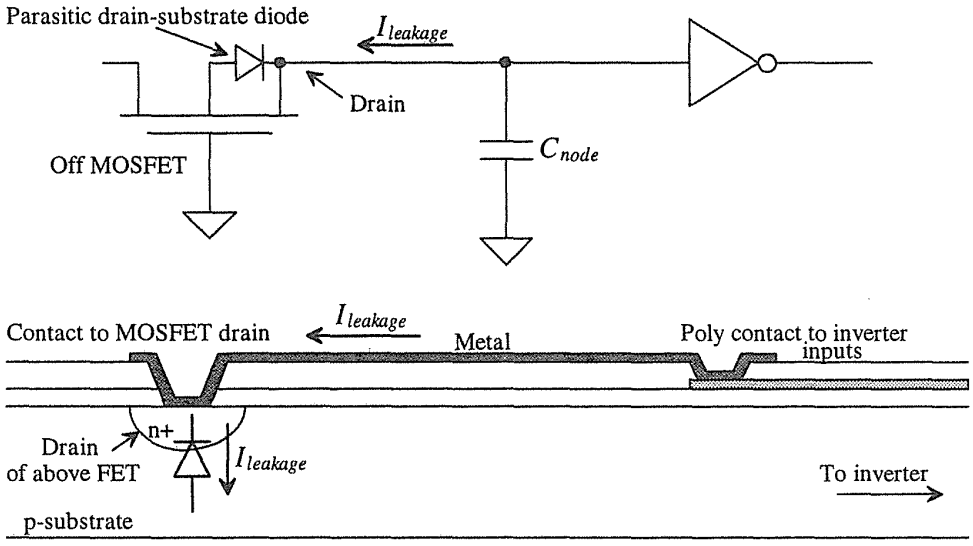



**Figure 14.2** Leakage from a storage node through the drain-substrate diode.

Microchip Ex. 1055, p. 300
Microchip v. Aptiv
IPR2024-00494

**Example 14.1**
Estimate the discharge rate of the 50 fF capacitor shown below. Assume that the MOSFET drain and source areas measure 6 μm by 6 μm.

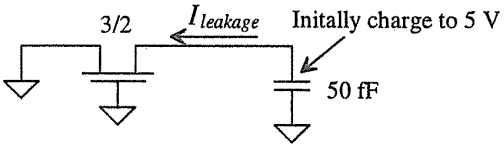



**Figure Ex14.1**

From the BSIM model parameters, $JS = 10^{-8}$ A/m$^2$; therefore, the leakage current can be estimated by

$$I_{leakage} = AD \cdot JS = 36\text{p} \cdot 10^{-8} = 360 \times 10^{-21}\text{A}$$

and the discharge rate is estimated by

$$\frac{dV}{dt} = \frac{360 \times 10^{-21}}{50\text{fF}} = 7.2\ \mu\text{V/s}$$

This is a very slow discharge rate. In practice, the MOSFET can have a nonzero gate-source voltage causing a subthreshold current to flow, increasing the discharge rate. Also, the value of current density given in the BSIM model in Appendix A and used above, that is, $JS = 10^{-8}$ A/m$^2$, is the SPICE default value. This indicates that the leakage current was not measured when generating the SPICE model, indicating another possible source of error. ■

### 14.1.2 Simulating Dynamic Circuits

Because of the extremely small leakage currents involved, simulating dynamic circuits can be difficult. First, when SPICE simulates any circuit, it puts a resistor with a conductance value given by the parameter GMIN across every pn junction and MOSFET drain to source. The default value of GMIN is $10^{-12}$ mhos or a 1 TΩ resistor. A charge storage node at a potential of 5 V has a leakage current, due to GMIN, of 5 pA. Of course, as the node voltage starts to decrease, the leakage current decreases as well. The leakage current calculated in Ex. 14.1 was $360 \times 10^{-21}$ A, or over a million times smaller than the 5 pA flowing through the default value of GMIN. The value of GMIN can be set using the .OPTIONS command, at the cost of a longer or more difficult convergence time, to a smaller value, say $10^{-15}$.

The ABSTOL (current accuracy), RELTOL (relative accuracy), or VNTOL (voltage tolerance) simulation parameters can limit the accuracy of the simulation and give false results. The default value of the current accuracy, ABSTOL, is 1 pA. Since the leakage from the drain-substrate diode can be significantly less than the 1 pA, ABSTOL must be reduced. If we set ABSTOL = 1E-21, the simulation accuracy is helped. However, when simulating, SPICE uses the larger of ABSTOL or the product of RELTOL and the simulation current to determine if convergence has been reached for a given current. Therefore, RELTOL would need to be reduced as well to get SPICE

Microchip Ex. 1055, p. 301
Microchip v. Aptiv
IPR2024-00494

results closer to hand calculations. In addition, CHGTOL, the capacitance charge tolerance that defaults at 10 fC (1 V on a 10 fF capacitor), would need reduction.

In practice, we use the default values of SPICE, which give a pessimistic estimate for the discharge time of storage nodes in dynamic circuits. The leakage current, for $VDD = 5$ V, is given by

$$I_{leakage} \approx 5\ pA = VDD \cdot GMIN \tag{14.6}$$

and

$$\frac{dV}{dt} = \frac{5\ pA}{C_{node}} = \frac{VDD \cdot GMIN}{C_{node}} \tag{14.7}$$

For $C_{node} = 50$ fF, it takes approximately 10 ms for the voltage on the charge storage node to fall 1 V. If 1 V is the most we will allow the node to fall before we apply another clock signal, then the minimum clock frequency is 100 Hz. The following example illustrates the dominance of GMIN in the simulation of a dynamic circuit.

**Example 14.2**

Simulate the circuit of Ex. 14.1. Estimate the discharge rate of the capacitor due to the default value of GMIN.

The discharge rate from Eq. (14.7) is 1 V per 10 ms for a GMIN of $10^{-12}$ mhos. The SPICE simulation results are shown in Fig. 14.3. Notice how the leakage drain current is jagged. This is the result of the numerical iteration scheme used by SPICE. The simulation currents will vary by an amount less than ABSTOL, or 1 pA. In most simulations, we do not see the small current variations. ■

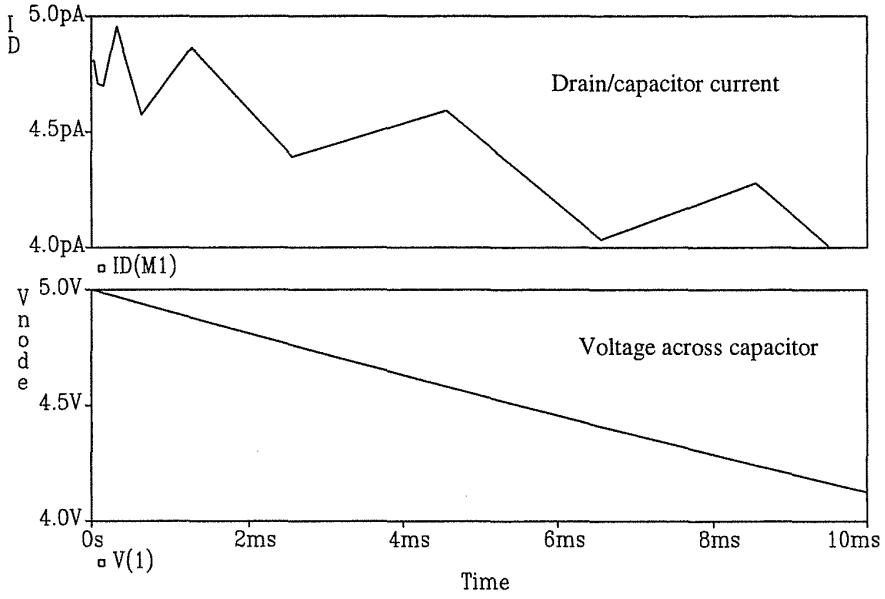



**Figure 14.3** Simulation results showing discharge of a capacitor.

Microchip Ex. 1055, p. 302
Microchip v. Aptiv
IPR2024-00494

### 14.1.3 Nonoverlapping Clock Generation

Consider the string of pass transistors/inverters shown in Fig. 14.4. This circuit is called a dynamic shift register. When $\phi_1$ goes high, the first and third stages of the register are enabled. Data are passed from the input to point A0 and from point A1 to A2. If $\phi_2$ is low while $\phi_1$ is high, the data cannot pass from A0 to A1 and from A2 to A3. If $\phi_1$ goes low and $\phi_2$ goes high, data are passed from A0 to A1 and from A2 to A3. If both $\phi_1$ and $\phi_2$ are high at the same time, the input of the shift register and the output are connected together, which is not desirable in a shift register application. The purpose of the inverter between pass transistors is to restore logic levels, since the n-channel pass transistor passes a high with a threshold voltage drop. Two inverters would be used to eliminate the logic inversion between stages. The clocks used in this dynamic circuit must be nonoverlapping, or logically

$$\phi_1 \cdot \phi_2 = 0 \tag{14.8}$$

There should be a period of dead time between transitions of the clock signals, labeled $\Delta$ in Fig. 14.4. The rise- and falltimes of the clock signals should not occur at the same time.

Since the design and layout of the dynamic shift register is straightforward let's concentrate on the generation of clock signals, $\phi_1$ and $\phi_2$. Note that a simple logic inversion will not generate nonoverlapping clock signals.

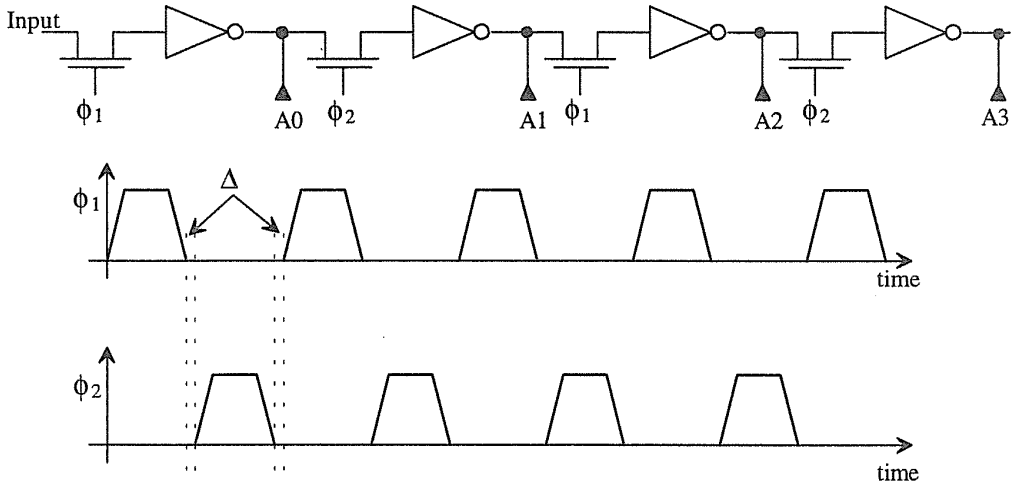



**Figure 14.4** Dynamic shift register with associated nonoverlapping clock signals.

Microchip Ex. 1055, p. 303
Microchip v. Aptiv
IPR2024-00494

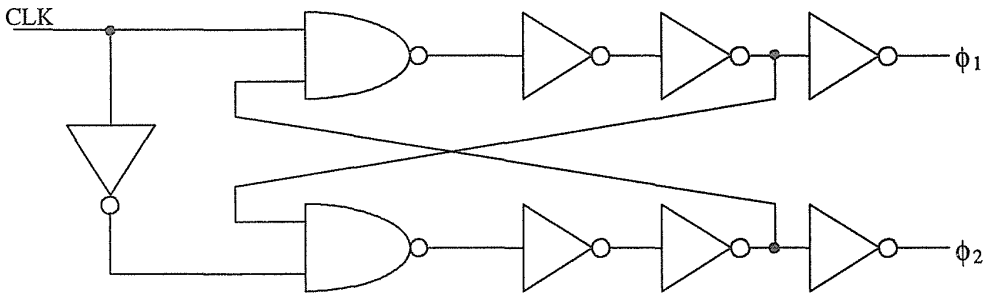



**Figure 14.5** Nonoverlapping clock generation circuit.

Consider the schematic of the nonoverlapping clock generator shown in Fig. 14.5. This circuit takes a clock signal and generates a two-phase nonoverlapping clock. The amount of overlap is set by the delay through the NAND gate and the two inverters on the NAND gate output. Consider the input clock going high. This forces $\phi_1$ high and $\phi_2$ low. When the input clock goes low, $\phi_1$ goes low. After $\phi_1$ goes low, $\phi_2$ can go high. When driving long transmission lines such as poly, where the risetime of the signals can be significant, a large number of inverters may need to be used. Line drivers, a string of inverters used to drive a large capacitance, can be used as part of the delay in the nonoverlapping clock generation circuit.

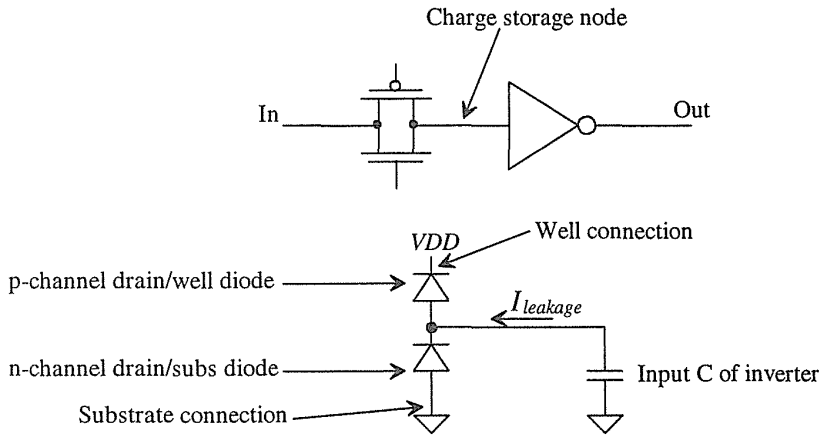



**Figure 14.6** CMOS TG used in dynamic logic.

Microchip Ex. 1055, p. 304
Microchip v. Aptiv
IPR2024-00494

### 14.1.4 CMOS TG in Dynamic Circuits

The CMOS TG used as a switch to charge or discharge the node capacitance of the charge storage node is shown in Fig. 14.6. Since understanding the charging and discharging of the input capacitance of the inverter follows many of the same analysis and discussions of Ch. 13, we will concentrate here on the charge leakage from the TG.

The leakage of charge off of or onto the input capacitance of the inverter in Fig. 14.6 can be attributed to the drain-well diode of the p-channel MOSFET and the drain-substrate diode of the n-channel MOSFET used in the TG. If these leakage currents were equal, then the leakage of charge off of the storage node would be zero. In general, we use the same hand analysis that was used for the n-channel MOSFET alone; namely, the leakage causes the voltage to change 1 volt in 10 ms. Notice that unlike the n-channel MOSFET, the charge storage node can leak to *VDD* or *VSS* (ground), depending on the size of the drain areas and the leakage currents.

## 14.2 Clocked CMOS Logic

Clocked CMOS, $C^2MOS$, logic is used to reduce power dissipation and layout size and to increase speed. The standard CMOS static gate requires 2N MOSFETs for an n-input gate. In general, an n-input $C^2MOS$ gate requires N + 2 MOSFETs where two MOSFETs are used in the clocking scheme. Additional MOSFETs can be used for buffering or for helping the gate appear more static in operation.

*Clocked CMOS Latch*

Consider the circuit shown in Fig. 14.7. This circuit performs the dynamic latch operation similar to the circuit of Fig. 14.6. When the clock input $\phi_1$ is high, the input is inverted and available on the output of the gate. For low-input clock signals, the

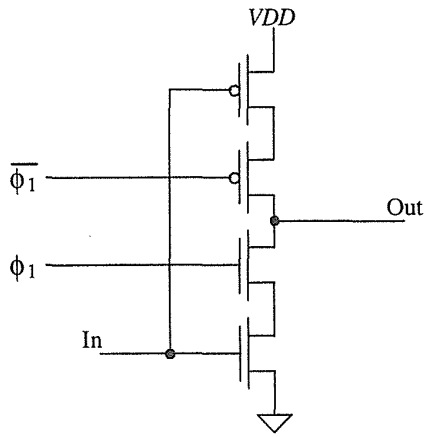



**Figure 14.7** A clocked CMOS latch. The clock signals can be generated with an RS FF so that the edges occur essentially at the same moment in time.

Microchip Ex. 1055, p. 305
Microchip v. Aptiv
IPR2024-00494

output is in the Hi-Z state, or in other words a high-impedance node very susceptible to signal feedthrough. The layout of the C²MOS gate is thus more critical than the static gate. Because of this node, running signal lines above this node in the layout is a definite problem. The output of the gate is not static. When the latch is enabled and $\phi_1$ is high, the capacitance on the output node is charged. The same leakage mechanisms present in the CMOS TG latch are present here. This limits the minimum clock frequency to about 100 Hz. Implementing a shift register requires nonoverlapping clocks for adjacent stages. The total number of clock signals needed for a C²MOS shift register is four. The nonoverlapping clocks $\phi_1$ and $\phi_2$ and their complements.

*PE Logic*

This section discusses precharge-evaluate logic, or PE logic. Consider the three-input NAND gate shown in Fig. 14.8. The operation of this gate relies on a single clock input. When $\phi_1$ is low, the output node capacitance is charged to *VDD* through M5. During the evaluate phase, $\phi_1$ is high, M1 is on, and if A0, A1, and A2 are high, the output is pulled low. The logic output is available only when $\phi_1$ is high. The output is a logic one when $\phi_1$ is low. One disadvantage of PE logic is that the gate logic output is available part of the time and not all of the time as in the static gates.

Several important characteristics of the PE gate should be pointed out. The input capacitance of the PE gate is less than that of the static gate. Each input is connected to a single MOSFET where the static gate inputs are tied to two MOSFETs. Potentially the PE gate is then faster and dissipates less power.

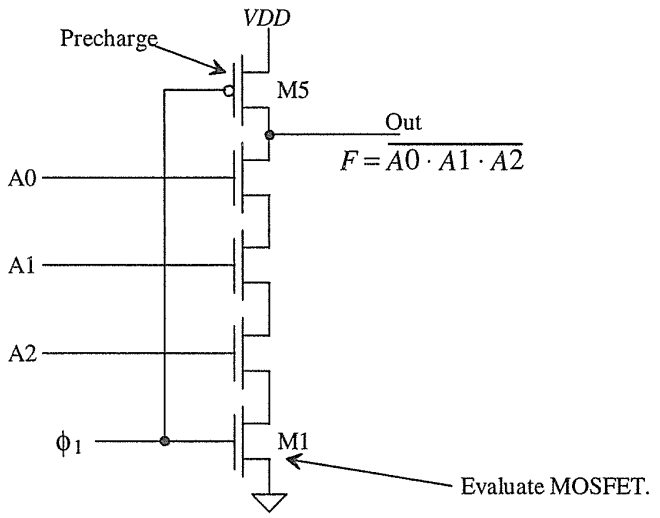



**Figure 14.8** Precharge-evaluate three-input NAND gate.

Microchip Ex. 1055, p. 306
Microchip v. Aptiv
IPR2024-00494

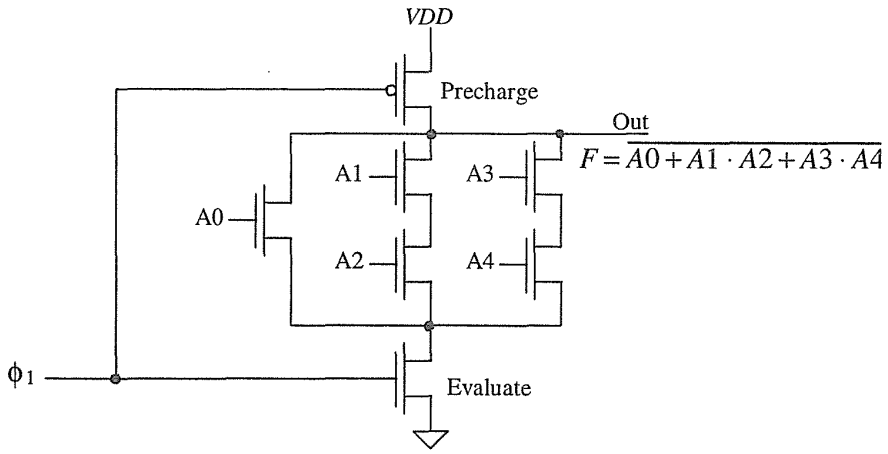



**Figure 14.9** A complex PE gate.

The size of the MOSFETs used in a PE gate does not need ratioing for symmetrical switching point voltage. The absence of complementary devices and the fact that the output is pulled high during each half cycle makes the gate $V_{SP}$ meaningless. However, we may need to size the devices to attain a certain speed for a given load capacitance. If the sizes of all NMOS transistors used in Fig. 14.8 are equal, then the $t_{PHL}$ is approximately $3R_nC_{node}$ and the $t_{PLH}$ is $R_pC_{node}$ where $C_{node}$ is the total capacitance on the output node. This may include the interconnecting capacitance and the input capacitance of the next stage. Here we have neglected both the transmission line effects through a series connection of MOSFETs and the intrinsic switching speeds. A more complex logic function, $F = \overline{A0 + A1 \cdot A2 + A3 \cdot A4}$, implemented in PE logic is shown in Fig. 14.9.

*Domino Logic*

Consider the cascade of PE gates shown in Fig. 14.10. During the precharge phase of the clock, the output of each PE gate is a logic high. This high-level output is connected to the input of the next PE gate. Suppose the logic out of the first PE gate during the evaluate phase is a low. This output will turn off any MOSFETs in the second PE gate. However, during the precharge phase, those same MOSFETs in the second PE gate will be turned on. The delay between the clock pulse going high and the valid output of the first gate will cause the second gates output to glitch or show an invalid logic output. If we can hold the output voltage of the PE gate low, instead of high we can eliminate this race condition. Upon adding an inverter to the PE gate (Fig. 14.11) the condition for glitch-free operation is met. The PE gate with the addition of an inverter is called Domino logic. The name *Domino* comes from the fact that a gate in a series of Domino logic gates cannot change output states until the previous gate changes states. The change in output of the gates occurs similar to a series of falling dominoes. The inverter used in the Domino gate has the added advantage that it can be sized to drive large capacitive loads.

Microchip Ex. 1055, p. 307
Microchip v. Aptiv
IPR2024-00494

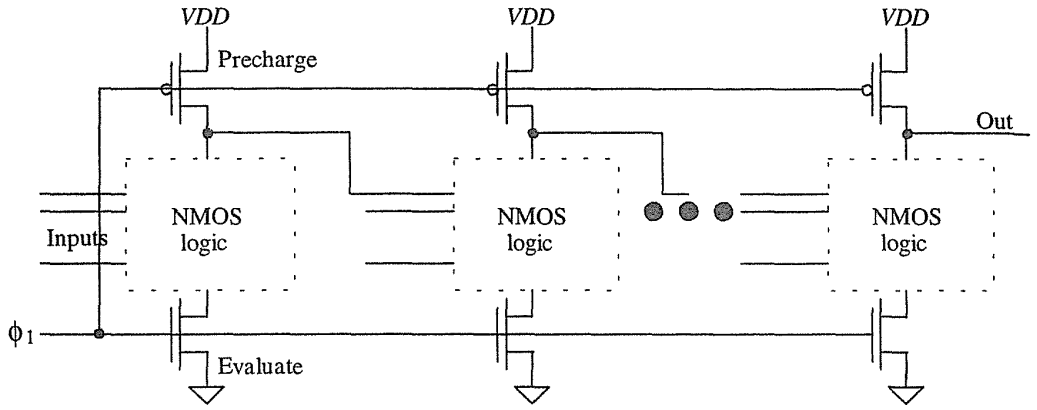



**Figure 14.10** Problems with a cascade of PE gates.

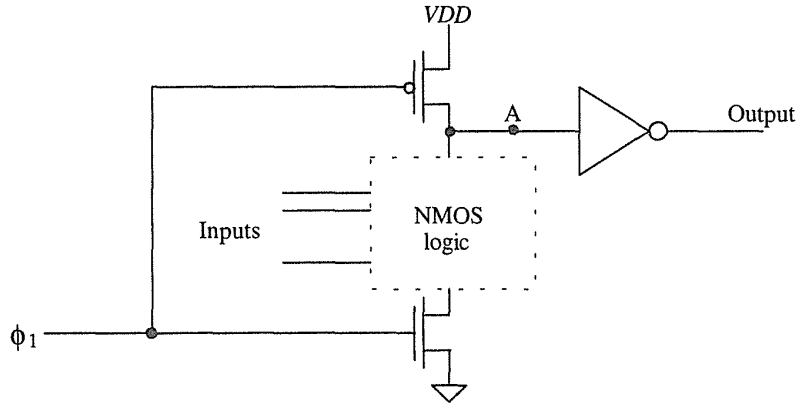



**Figure 14.11** Domino logic gate.

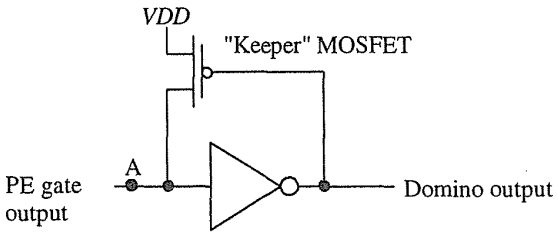



**Figure 14.12** Keeper MOSFET used to hold node A in Fig. 14.11 at VDD when PE gate output is high.

Microchip Ex. 1055, p. 308
Microchip v. Aptiv
IPR2024-00494

One problem does exist with this scheme, however, referring to Fig. 14.11, note that during the precharge phase, node A is charged to *VDD*. If the NMOS logic results in a logic high on node A during the evaluate phase, then that node is at a high impedance with no direct path to *VDD* or ground. The result is charge leakage off of node A when the PE output is a logic high. The circuit of Fig. 14.12 eliminates this problem. A "keeper" p-channel MOSFET is added to help keep node A at *VDD* when the NMOS logic is off. The *W/L* of this MOSFET is small, so that it provides enough current to compensate for the leakage but not so much that the NMOS logic can't drive node A down to ground.

*NP Logic (Zipper Logic)*

The idea behind implementing a logic function using NP logic is shown in Fig. 14.13. Staggering NMOS and PMOS stages eliminates the need for and delay associated with the inverter used in Domino logic, making higher speed operation possible. A circuit that can easily be implemented in NP logic is the full adder circuit of Fig. 12.18. The NMOS section of the carry circuit is implemented in the first section of the NP logic, while the PMOS section of the sum circuit is implemented in the PMOS section of the NP logic gate.

*Pipelining*

The NP logic adder just described performs one two-bit addition with carry during each clock cycle. Adding two-four bit words requires the use of pipelining [4]; see Fig. 14.14. The bits of the word are delayed, both on the input and output of the adder, so that all bits of the sum reach the output of the adder at the same time. Note, however, that two new four-bit words can be input to the adder at the beginning of each clock cycle and that it takes four clock cycles to finish the addition of the two words. If this circuit were dedicated to continually performing the addition of two words, we could input the words at a very fast rate, around 30 Mwords/s for the CN20 process. However, since performing a single addition requires four clock cycles, applications of pipelining where two numbers are not added continuously can result in longer delay-times.

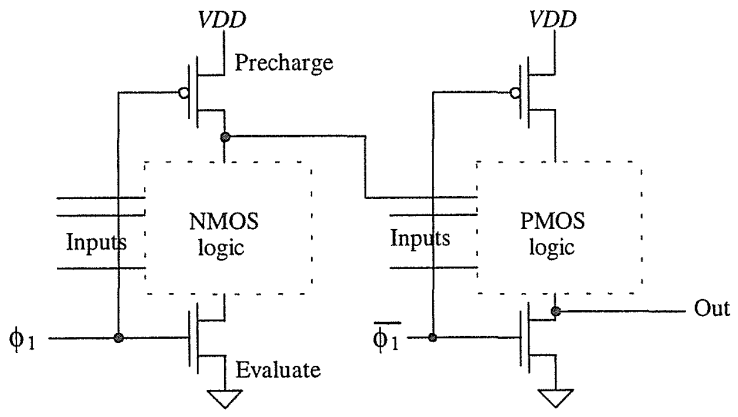



**Figure 14.13** NP logic.

Microchip Ex. 1055, p. 309
Microchip v. Aptiv
IPR2024-00494

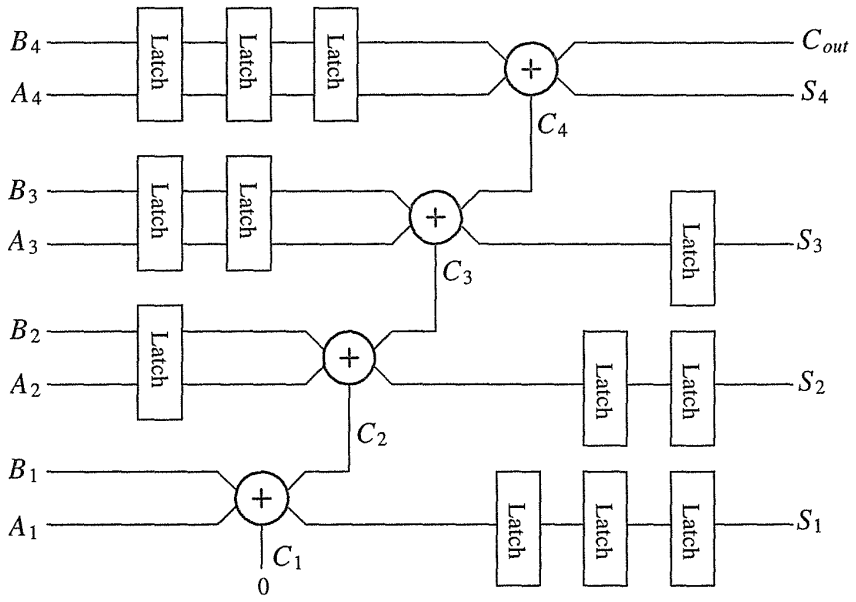



**Figure 14.14** A pipelined adder. The latches (clocked) behave as delay elements.

## REFERENCES

[1] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI-Design Techniques for Analog and Digital Circuits,* McGraw-Hill Publishing Co., 1990. ISBN 0-07-023253-9.

[2] M. I. Elmasry, *Digital MOS Integrated Circuits II,* IEEE Press, 1992. ISBN 0-87942-275-0, IEEE order number: PC0269-1.

[3] J. P. Uyemura, *Circuit Design for Digital CMOS VLSI,* Kluwer Academic Publishers, 1992.

[4] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design,* Addison-Wesley, 2nd ed., 1993. ISBN 0-201-53376-6.

[5] J. Yuen and C. Svensson, "New Single-Clock CMOS Latches and Flipflops with Improved Speed and Power Savings," *IEEE Journal of Solid-State Circuits,* Vol. 32, No. 1, pp. 62-69, 1997.

## PROBLEMS

Unless otherwise stated, use the CN20 process.

**14.1** Repeat Ex. 14.2 with a GMIN of $10^{-9}$ mhos. Use the .OPTIONS to set the value of GMIN.

Microchip Ex. 1055, p. 310
Microchip v. Aptiv
IPR2024-00494

**14.2** Simulate the operation of the nonoverlapping clock generator circuit made using minimum size MOSFETs in Fig. 14.5. Assume that the input clock signal is running at 50 MHz. Show how both $\phi_1$ and $\phi_2$ are nonoverlapping.

**14.3** Design and simulate the operation of a PE gate that will implement the logical function $F = \overline{ABCD + E}$.

**14.4** Simulate the operation of the clocked CMOS latch shown in Fig. 14.7. Use minimum-size MOSFETs.

**14.5** If the PE gate shown in Fig. 14.9 drives a 50 fF capacitor, estimate the worst-case $t_{PHL}$.

**14.6** Implement an XOR gate using Domino logic. Simulate the operation of the resulting implementation.

**14.7** The circuit shown in Fig. P14.7 is the implementation of a high-speed adder cell (1-bit). What type of logic was used to implement this circuit? Using timing diagrams, describe the operation of the circuit.

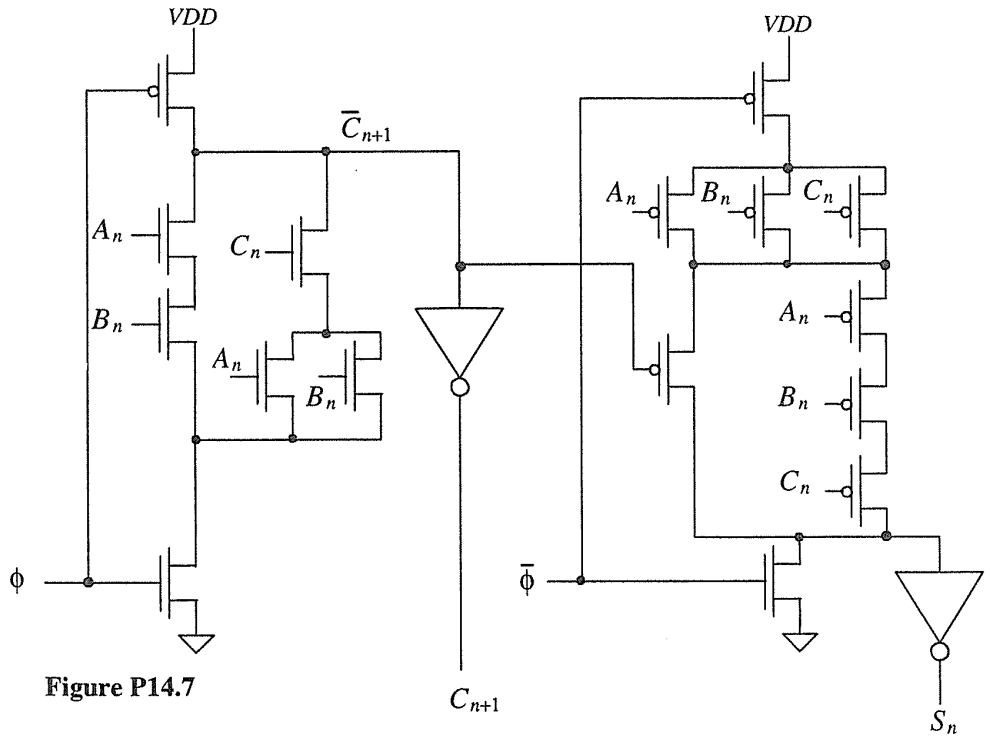



**Figure P14.7**

**14.8** Discuss the design of a two-bit adder using the adder cell of Fig. P14.7. If a clock, running at 20 MHz, is used with the two-bit adder, how long will it take to add two words? How long will it take if the word size is increased to 32 bits?

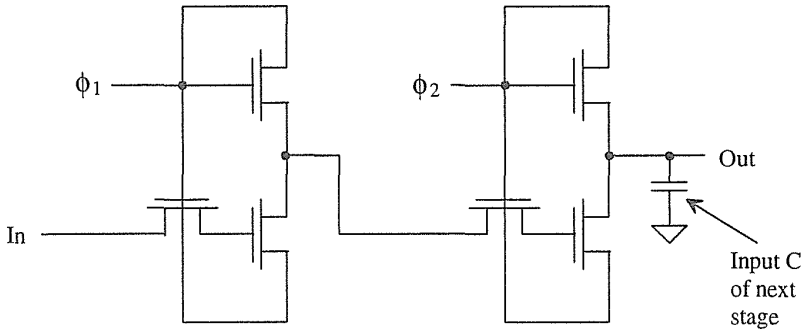
Microchip Ex. 1055, p. 311
Microchip v. Aptiv
IPR2024-00494

**14.9** Sketch the implementation of an NP logic half adder cell.

**14.10** Design (sketch the schematic of ) a full adder circuit using PE logic.

**14.11** Simulate the operation of the circuit designed in Problem 14.10.

**14.12** Figure P14.12 shows one bit of a shift register implemented in the so-called ratioless NMOS logic. The term *ratioless* results from the fact that the MOSFET sizes do not affect the switching point voltages. Also, this gate can be laid out in a very small area and the outputs can swing down to ground. Discuss and simulate the operation of this circuit. Keep in mind that $\phi_1$ and $\phi_2$ are nonoverlapping clock signals. What is the maximum output voltage of this circuit?




All MOSFETs are minimum size.

**Figure P14.12**

**14.13** Show that the dynamic circuit shown in Fig. P14.13 is an edge-triggered flip-flop [5]. Note that a single-phase clock signal is used.

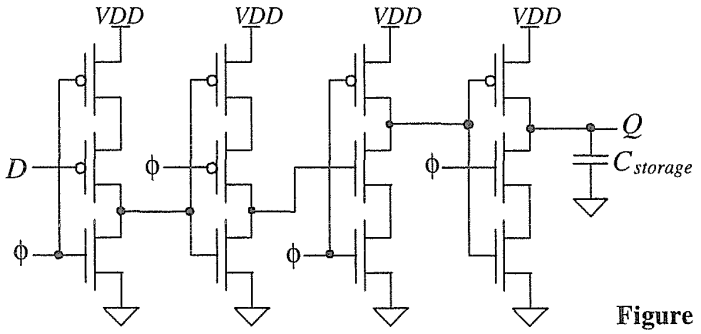



**Figure P14.13**

Microchip Ex. 1055, p. 312
Microchip v. Aptiv
IPR2024-00494

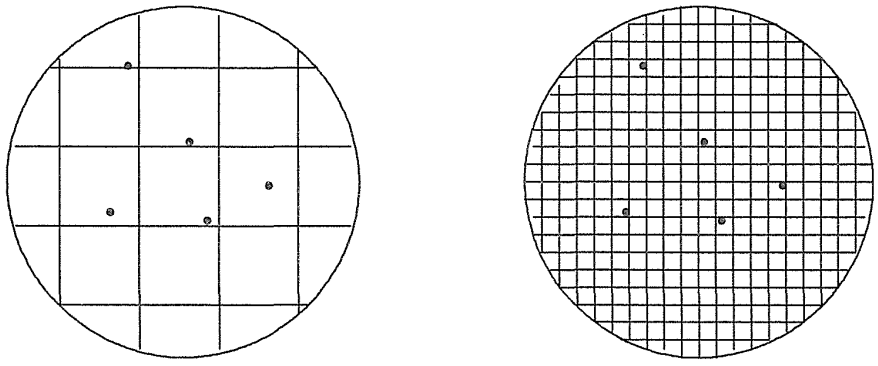
Chapter

15

# VLSI Layout

The past chapters have concentrated on basic logic-gate design and layout. In this chapter we discuss the implementation of logic functions on a chip where the size and organization of the layouts are of importance. The number of MOSFETs on a chip, depending on the application, can range from tens (an op-amp) to hundreds of millions (a 256 MEG DRAM). Designs where thousands of MOSFETs or more are integrated on a single die are termed *very-large-scale-integration* (VLSI) designs.

To help us understand why chip size is important, examine Fig. 15.1. The dark dots indicate a defect that will lead to a chip which doesn't function properly. Figure 15.1a shows a wafer with nine full die. The partial die around the edge of the wafer are wasted. Five of the nine die do not contain a defect and thus can be packaged and sold. Next consider a reduction in the die size (Fig. 15.1b). We are assuming each die,



(a) (b)

**Figure 15.1** Defect density effects on yield.

Microchip Ex. 1055, p. 313
Microchip v. Aptiv
IPR2024-00494

whether discussing the die of Fig. 15.1a or b, performs the same function. This reduction can be the result of having better layout (resulting in a smaller layout area) or fabricating the chips in a process with smaller device dimensions (e.g., going from a 2 μm process to a 0.5 μm process). The total number of die lost (see Fig. 15.1b), due to defects is five; however, the number of good die is significantly larger than the five good die of Fig. 15.1a. The yield (number of good die/total number of die on the wafer) is increased with smaller die size. The result is more die/wafer available for sale. Another benefit of reducing die size comes from the realization that processing costs per wafer are constant and increasing the number of die on a wafer decreases the cost per die.

## 15.1 Chip Layout

VLSI designs can be implemented using many different techniques including gate-arrays, standard-cells, and full-custom design[1]. Since designs based on gate-arrays are, in general, used where low volume and fast turnaround time are required and the chip designer need know little to nothing[1] about the actual implementation of the CMOS circuits, we will concentrate on full-custom design and design using standard cells.

*Regularity*

An important consideration when implementing a VLSI chip design is regularity. The layout should be an orderly arrangement of cells. Toward this goal the first step in designing a chip is drawing up a chip (or section of the chip) floor plan. Figure 15.2 shows a simple floor plan for an adder data-path. This floor plan can be added to the floor plan of an overall chip, which includes output buffers, control logic, and memory.

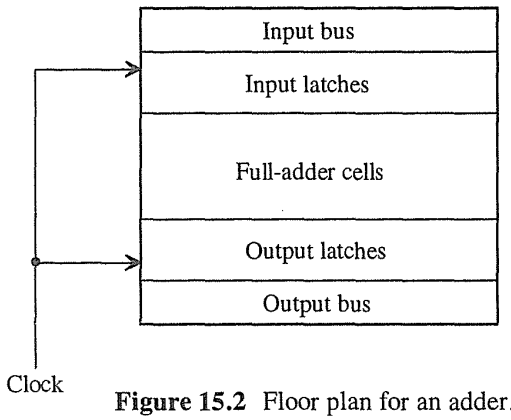



**Figure 15.2** Floor plan for an adder.

[1] At many universities, programmable-logic-arrays (PLAs) are discussed in the first course on Digital Logic Design, while design using a hardware description language (HDL) with field-programmable-gate arrays (FPGAs) is discussed in the first course on Digital Systems Design.

Microchip Ex. 1055, p. 314
Microchip v. Aptiv
IPR2024-00494

At this point, we may ask the question, "How do we determine the size of the blocks in Fig. 15.2?" The answer to this question leads us into the design and layout of the cells used to implement each of the logic blocks in Fig. 15.2.

*Standard-Cell Examples*

Standard cells are layouts of logic elements including gates, flip-flops, and ALU functions that are available in a cell library for use in the design of a chip. *Custom design* refers to the design of cells or standard cells using MOSFETs at the lowest level. *Standard-cell design* refers to design using standard cells; that is, the designer connects wires between standard cells to create a circuit or system. The difference between the two types of design can be illustrated using a printed circuit board-level analogy. A standard-cell design is analogous to designing with packaged parts. The design is accomplished by connecting wires between the pins of the packaged parts. Custom design is analogous to designing the "insides" of the packaged parts themselves.

Figure 15.3 shows an example of an inverter [2]. In addition to keeping the layout size as small as possible, an important consideration, when laying out a standard cell, is the routing of signals. Keeping this in mind, we can state the following general guidelines for standard-cell design:

1. Cell inputs and outputs should be available, at the same relative horizontal distance, on the top and bottom of the cell.
2. Horizontal runs of metal are used to supply power and ground to the cell, a.k.a. power and ground busses. Also, well and substrate tie downs should be under these busses.
3. The height of the cells should be a constant, so that when the standard cells are placed end to end the power and ground busses line up. The width of the cell should be as narrow as the layout will allow. However, the absolute width is not important and can be increased as needed.
4. The layout should be labeled to indicate power, ground, input, and output connections. Also, an outline of the cell, useful in alignment, should be added to the cell layout.

Figure 15.4 illustrates the connection of standard cells to a bus. Note that poly, which runs vertically, can cross the metal1 lines, which run horizontally without making contact. This fact is used to route signals and interconnect standard cells in a VLSI design. Also, in this figure, note how the two inverter standard cells are placed end to end. The result is that power and ground are automatically routed to each cell.

Other examples of static standard cells are shown in Fig. 15.5. A double inverter standard cell is shown in Fig. 15.5a while NAND, NOR, and transmission gate standard cells are shown in Figs. 15.5b, c, and d.

Figure 15.6 shows the layout of a NAND-based SR flip-flop. This layout differs from the others we have discussed. All layouts discussed so far have metal1 and

Microchip Ex. 1055, p. 315
Microchip v. Aptiv
IPR2024-00494

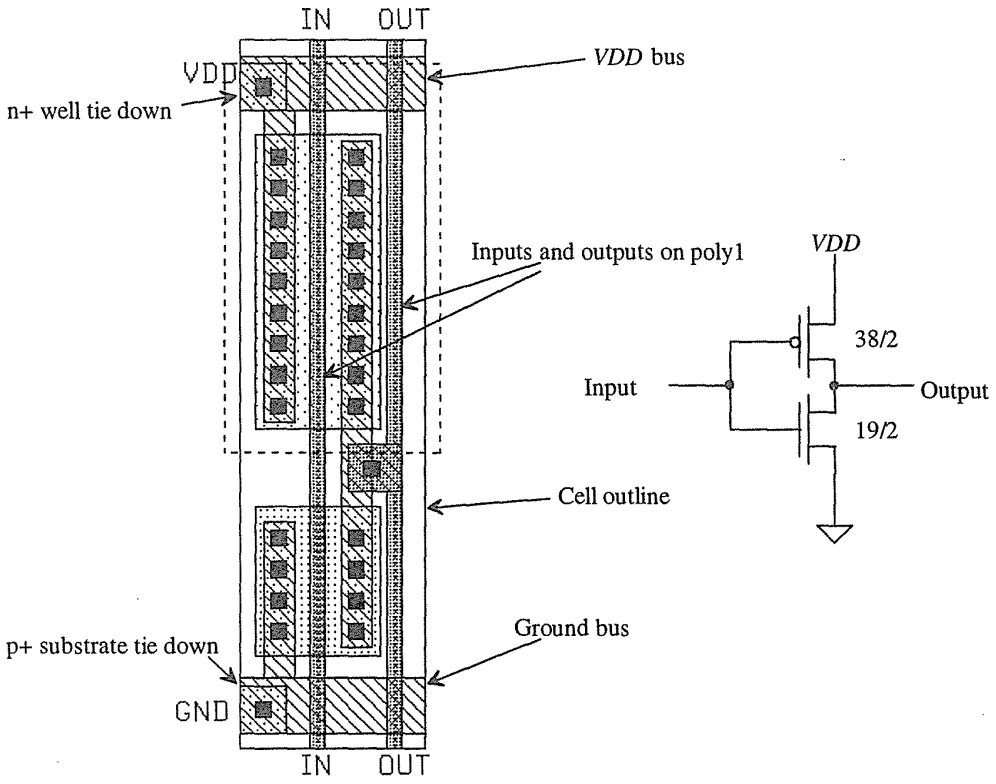



**Figure 15.3** Standard-cell layout of an inverter.

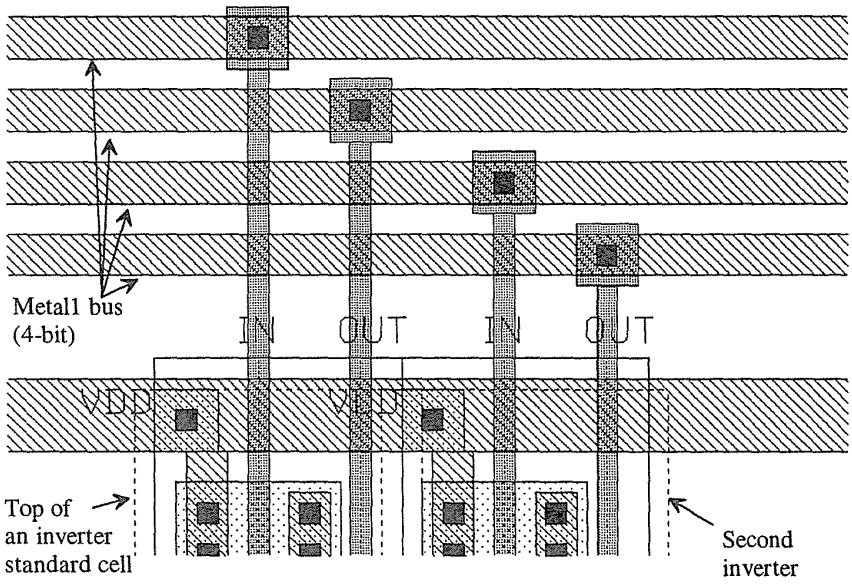



**Figure 15.4** Connection of two inverter standard cells to a bus.

Microchip Ex. 1055, p. 316
Microchip v. Aptiv
IPR2024-00494

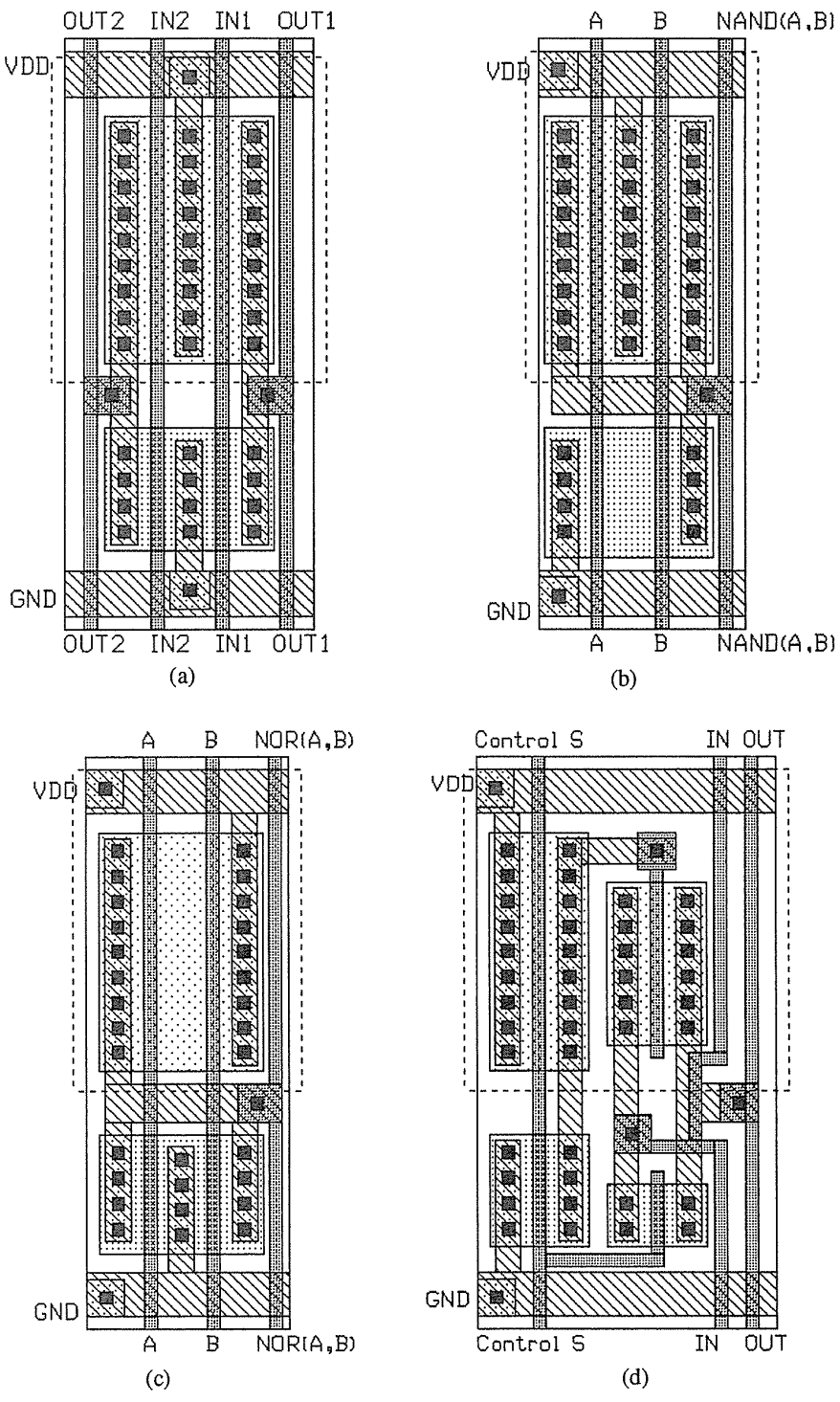



**Figure 15.5** (a)Double inverter, (b) two-input NAND, (c) two-input NOR, and (d) transmission gate.

Microchip Ex. 1055, p. 317
Microchip v. Aptiv
IPR2024-00494

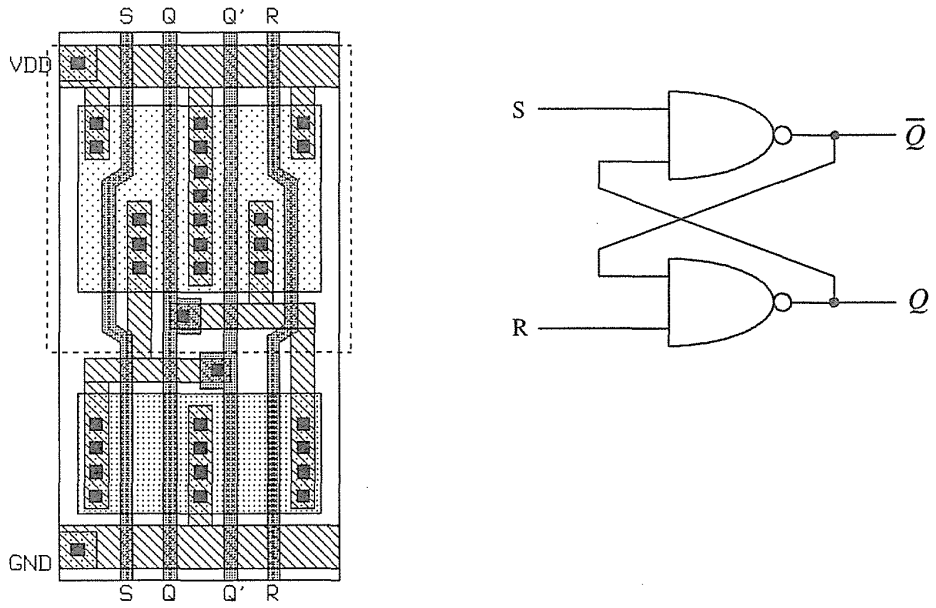



**Figure 15.6** SR flip-flop using NAND gates.

contacts adjacent to the gate poly. Also, the gate poly has been laid down without bends. The expanded view of a p-channel MOSFET used in the SR flip-flop is shown in Fig. 15.7. Keeping in mind that whenever poly crosses active (n+ or p+) a MOSFET is formed, we see that the source of the MOSFET is connected to metal through two contacts, while the p+ implant forms a resistive connection to metal1 along the remainder of the device. The layout size, in this case the width of the standard cell, can be reduced using this technique. Because of the bend in the gate, the width of this

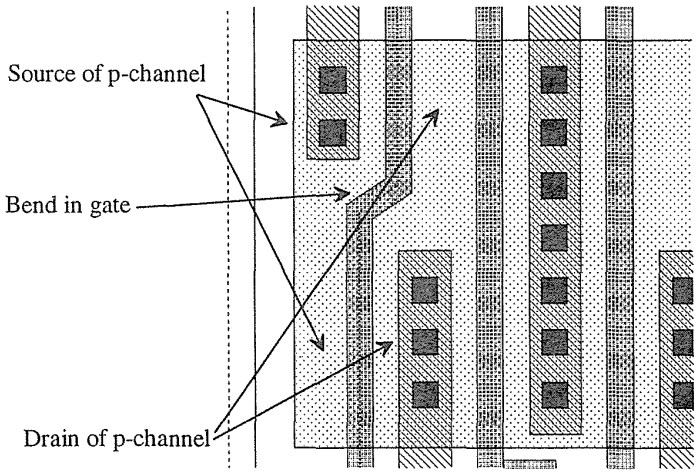



**Figure 15.7** Section of the layout shown in Fig. 15.6.

Microchip Ex. 1055, p. 318
Microchip v. Aptiv
IPR2024-00494

MOSFET is longer than the adjacent MOSFET. This additional width is of little importance and has little effect on the DC and transient properties of the gate. Figure 15.8 shows the NOR implementation of an SR flip-flop.

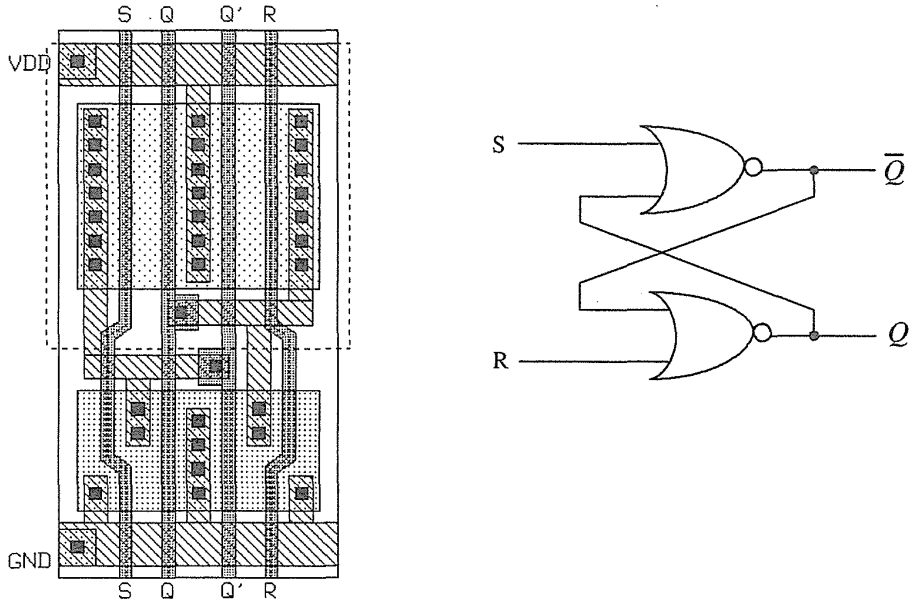



**Figure 15.8** SR flip-flop using NOR gates.

*Power and Ground Connections*

Many of the problems encountered when designing a chip can be related to distribution of power and ground. When power and ground are not distributed properly, noise can be coupled from one circuit onto the power and ground conductors and injected into some other circuit.

Consider the placement of standard cells in a padframe shown in Fig. 15.9a without connections to power and ground shown. Approximately 600 standard cells are shown in this figure. The space between the rows of standard cells is used for the routing of signals. A line drawing of a possible power and ground bussing architecture is shown in Fig. 15.9b. Consider the section of bus shown in Fig. 15.9c. Wire A is used to connect the standard cells in the top row to *VDD*, while wire B is used for connection to ground. Ideally, the current supplied on A (*VDD*) is returned on B (ground). In practice, there exists coupling between conductors B and C, which gives rise to an unwanted signal (noise) on either conductor. This coupling can be reduced by increasing the space between B and C. This reduces the inductive and capacitive coupling between the conductors. Another solution is to increase the capacitance between A and B. A standard-cell decoupling capacitor (Fig. 15.10) can be used toward

Microchip Ex. 1055, p. 319
Microchip v. Aptiv
IPR2024-00494

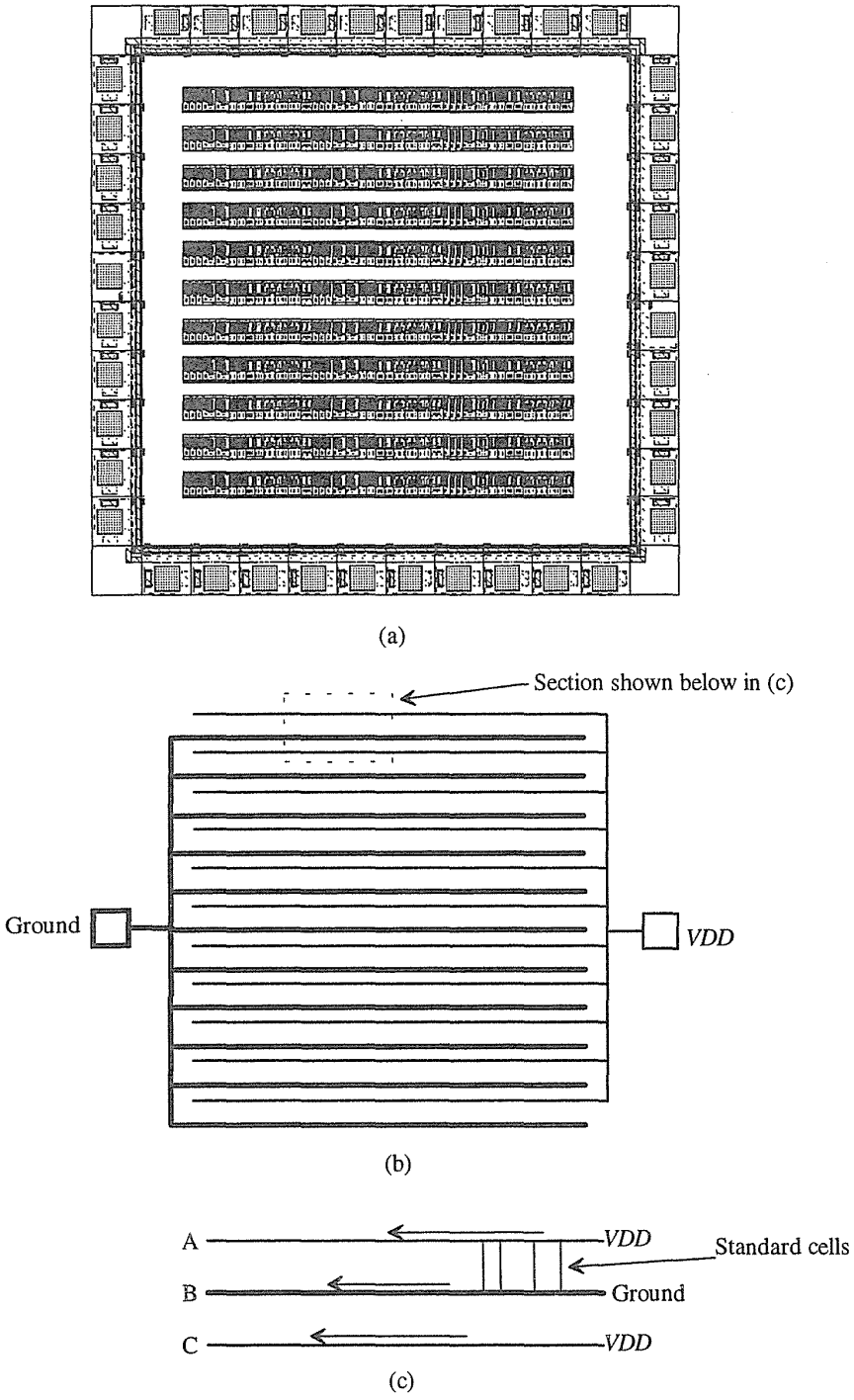



**Figure 15.9** Connection of power and ground to standard cells.

Microchip Ex. 1055, p. 320
Microchip v. Aptiv
IPR2024-00494

this goal. The capacitor is placed in the middle of a standard-cell row. Also, the AC resistive drop effects discussed in Ch. 3 are greatly reduced by inclusion of this capacitor.

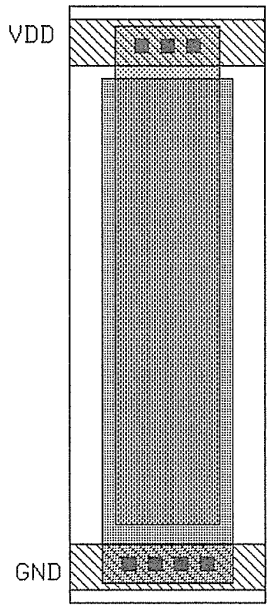



**Figure 15.10** Decoupling capacitor.

Coupling is a problem on signal busses as well. Figure 15.11 shows a simple scheme used to reduce coupling. The length of a section, where two wires are adjacent, is reduced by routing the wire to other locations at varying distances along the bus. The inductive or capacitive coupling between two conductors is directly related to the length of the wire runs.

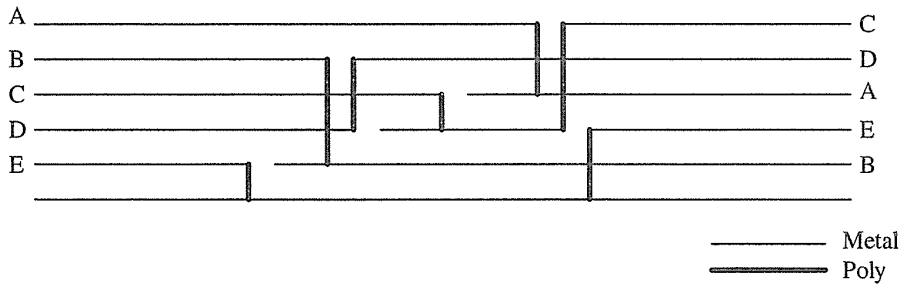
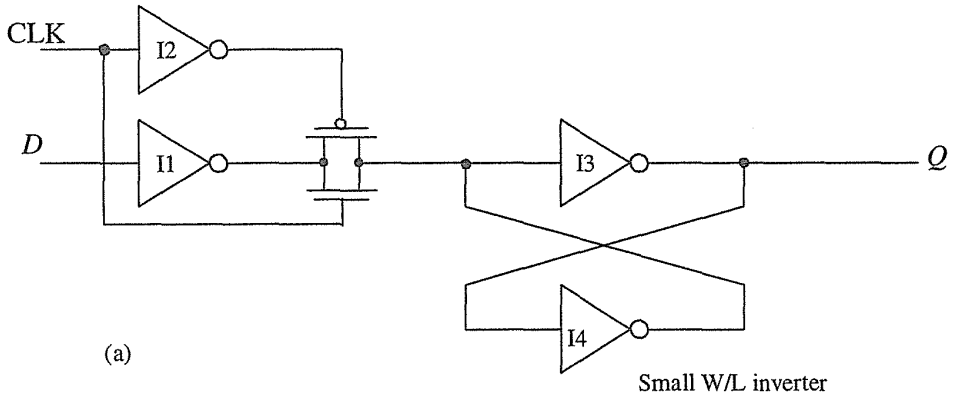



**Figure 15.11** Bussing structure used to decrease signal coupling.

Microchip Ex. 1055, p. 321
Microchip v. Aptiv
IPR2024-00494

*An Adder Example*

As another example, let's consider the implementation of a four-bit adder. (The floorplan for this adder was shown in Fig. 15.2.) The first components that must be designed are the input and output latches. Figure 15.12a shows the schematic of the D flip-flop used in the latches. This FF is the level-triggered type discussed in the last chapter. When CLK is high, the output, *Q*, changes states with the input, *D*. The inverter, I4, is used to provide positive feedback and is sized with a small *W/L* ratio so that I1 does not need to supply a large amount of DC current to force the latch to change



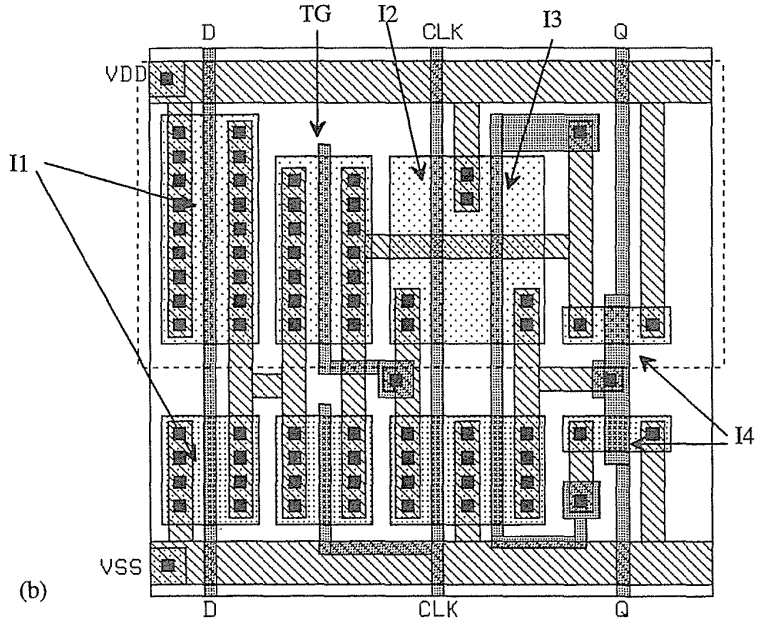




**Figure 15.12** Schematic and layout of a D FF.

Microchip Ex. 1055, p. 322
Microchip v. Aptiv
IPR2024-00494

states. The layout of the FF is shown in Fig. 15.12b. The layout size and the size of the MOSFETs used in these examples are larger than what would be used in practice to make understanding and viewing the layouts easier.

The layout of the static adder is shown in Fig. 15.13. This is the implementation, using near minimum-size MOSFETs, of the AOI static adder of Fig. 12.18. Both the carry-out and sum-out logic functions are implemented in this cell.

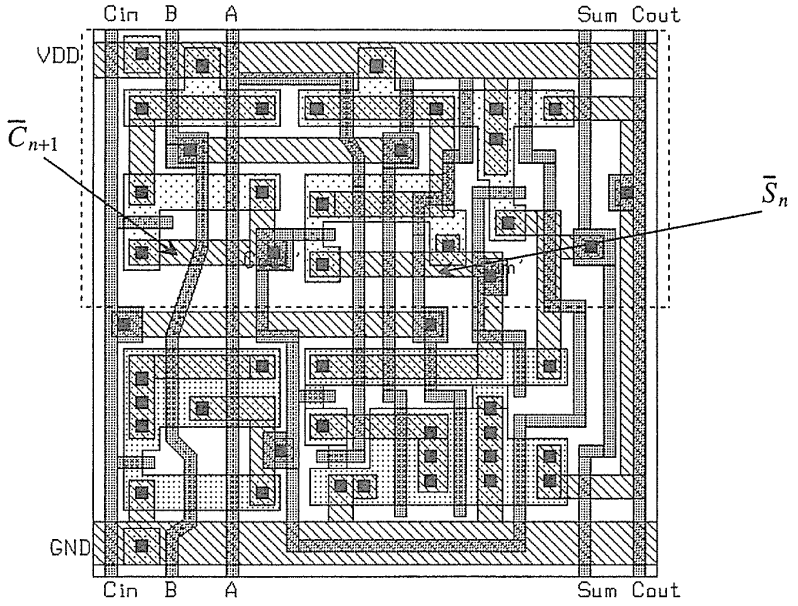



**Figure 15.13** Layout of the static adder of Fig. 12.18.

The complete layout of the adder is shown in Fig. 15.14. The two four-bit words, Word-A and Word-B, are input to the adder on the input bus. These data are clocked into the input latch when CLK is high, while the results of the addition are clocked into the output latch when CLK is low. The inverter standard cell of Fig. 15.3 is placed at the end of the output latches and is used to generate $\overline{\text{CLK}}$ for use in the output latches. The inputs and outputs of the adder cells are run on poly because of the short distances involved. The carry-in of the adders is connected to ground, as shown in the figure.

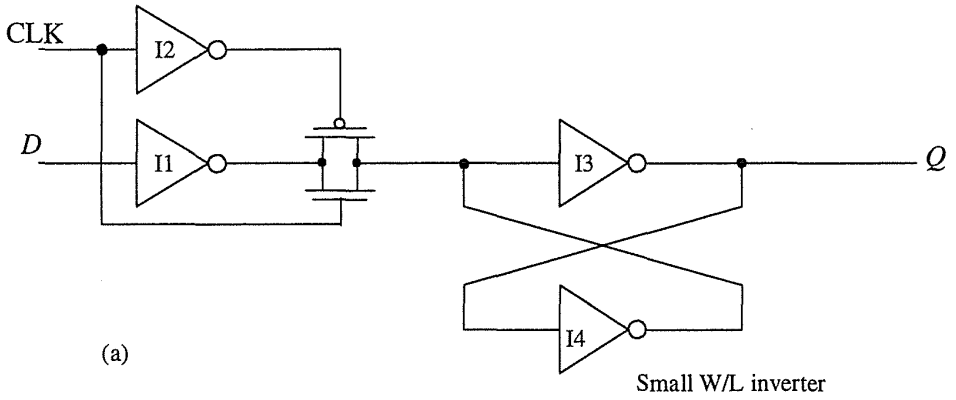
*A 4 to 1 MUX/DEMUX*

The layout of a 4 to 1 MUX/DEMUX is shown in Fig. 15.15 (based on the circuit schematic of Fig. 13.11). This layout is different from the layouts discussed so far since the circuit does not require power and ground connections and the input/output signals are connected on n+. The select signals are supplied to the circuit on metal1 at the top of the layout. For A to be connected to the output, the signals S1 and S2 should be high. For a large MUX, the propagation delay through the n+ should be considered.

Microchip Ex. 1055, p. 323
Microchip v. Aptiv
IPR2024-00494

*An Adder Example*

As another example, let's consider the implementation of a four-bit adder. (The floorplan for this adder was shown in Fig. 15.2.) The first components that must be designed are the input and output latches. Figure 15.12a shows the schematic of the D flip-flop used in the latches. This FF is the level-triggered type discussed in the last chapter. When CLK is high, the output, *Q*, changes states with the input, *D*. The inverter, I4, is used to provide positive feedback and is sized with a small *W/L* ratio so that I1 does not need to supply a large amount of DC current to force the latch to change



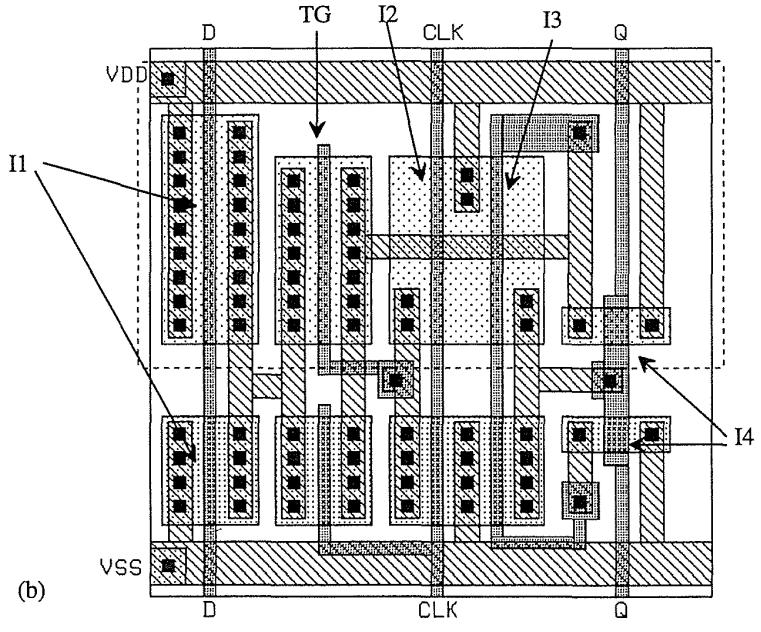




**Figure 15.12** Schematic and layout of a D FF.

Microchip Ex. 1055, p. 324
Microchip v. Aptiv
IPR2024-00494

states. The layout of the FF is shown in Fig. 15.12b. The layout size and the size of the MOSFETs used in these examples are larger than what would be used in practice to make understanding and viewing the layouts easier.

The layout of the static adder is shown in Fig. 15.13. This is the implementation, using near minimum-size MOSFETs, of the AOI static adder of Fig. 12.18. Both the carry-out and sum-out logic functions are implemented in this cell.

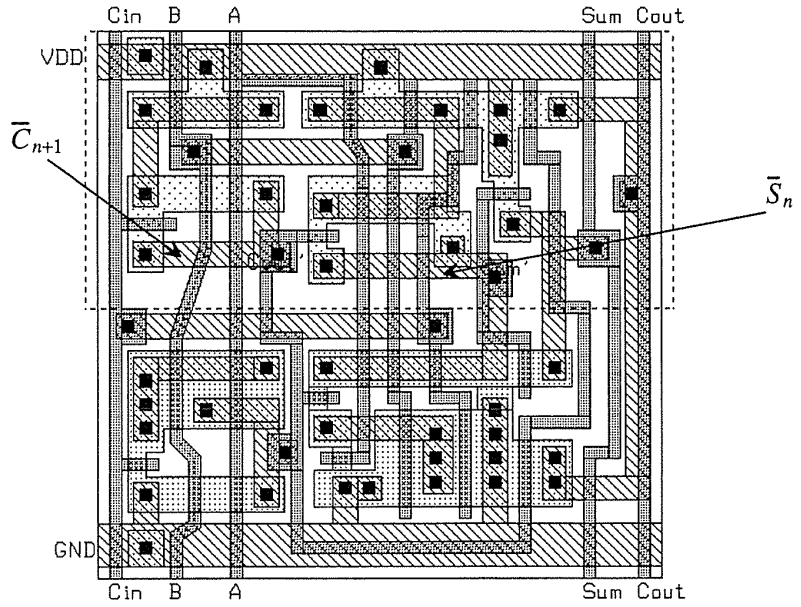



**Figure 15.13** Layout of the static adder of Fig. 12.18.

The complete layout of the adder is shown in Fig. 15.14. The two four-bit words, Word-A and Word-B, are input to the adder on the input bus. These data are clocked into the input latch when CLK is high, while the results of the addition are clocked into the output latch when CLK is low. The inverter standard cell of Fig. 15.3 is placed at the end of the output latches and is used to generate $\overline{CLK}$ for use in the output latches. The inputs and outputs of the adder cells are run on poly because of the short distances involved. The carry-in of the adders is connected to ground, as shown in the figure.

*A 4 to 1 MUX/DEMUX*

The layout of a 4 to 1 MUX/DEMUX is shown in Fig. 15.15 (based on the circuit schematic of Fig. 13.11). This layout is different from the layouts discussed so far since the circuit does not require power and ground connections and the input/output signals are connected on n+. The select signals are supplied to the circuit on metal1 at the top of the layout. For A to be connected to the output, the signals S1 and S2 should be high. For a large MUX, the propagation delay through the n+ should be considered.

Microchip Ex. 1055, p. 325
Microchip v. Aptiv
IPR2024-00494

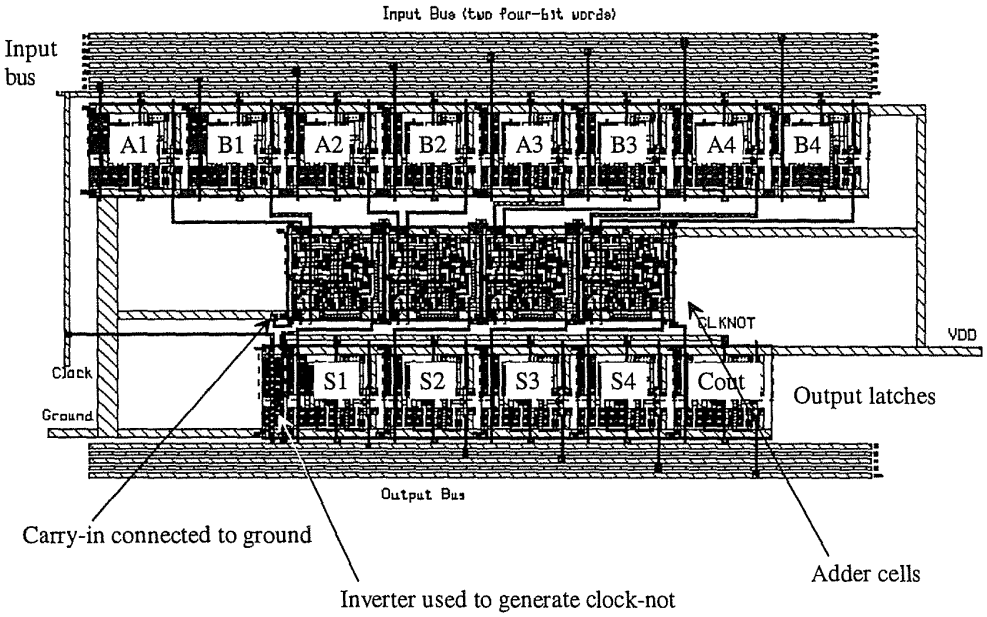



**Figure 15.14** Layout of the complete adder.

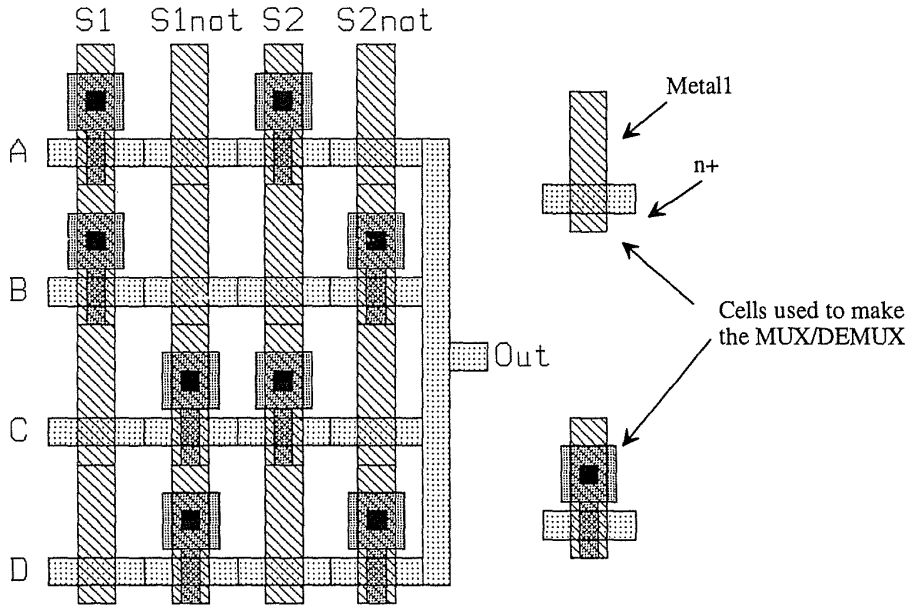



**Figure 15.15** Layout of a 4 to 1 MUX/DEMUX.

Microchip Ex. 1055, p. 326
Microchip v. Aptiv
IPR2024-00494

## 15.2 Layout Steps *by Dean Moriarty, Crystal Semiconductor*

The steps involved in rendering a schematic diagram into its physical layout are: plan, place, connect, polish, and verify. Let's illustrate each of these steps in some detail through the use of examples.

*Planning and Stick Diagrams*

The planning steps start with paper and pencil. Colored pencils are useful for distinguishing one object from another. You can use gridded paper to help achieve a sense of proportion in the cell plan but don't get too bogged down in the details of design rules or line widths at this point; we just want to come up with a general plan. A "stick diagram" is a paper and pencil tool that you can use to plan the layout of a cell. The stick diagram resembles the actual layout but uses "sticks" or lines to represent the devices and conductors. When used thoughtfully, it can reveal any special hook-up problems early in the layout, and you can then resolve them without wasting any time.

Figure 15.16a shows the schematic of an inverter. In order to realize the layout of this circuit, it is first necessary to define the direction and metalization of the power supply, ground, input, and output. Since the standard-cell template "sframe" in the CN20 setups does this for us, we'll use it. Power and ground run horizontally in metal1 and are each 7 microns wide. The input and output are accessible from the top or bottom of the cell and will be in metal2 running vertically. Figure 15.16b shows the completed stick diagram. Note the use of "X" and "O" to denote contacts and vias, respectively. The stick diagram should be compared to the resulting layout of Fig. 15.17.

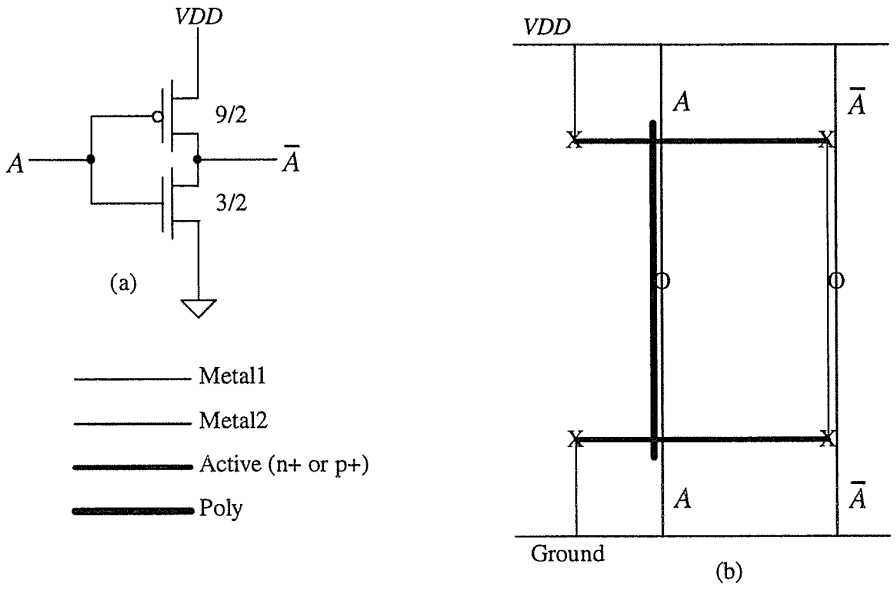



**Figure 15.16** (a) Inverter and (b) stick diagram used for layout.

Microchip Ex. 1055, p. 327
Microchip v. Aptiv
IPR2024-00494

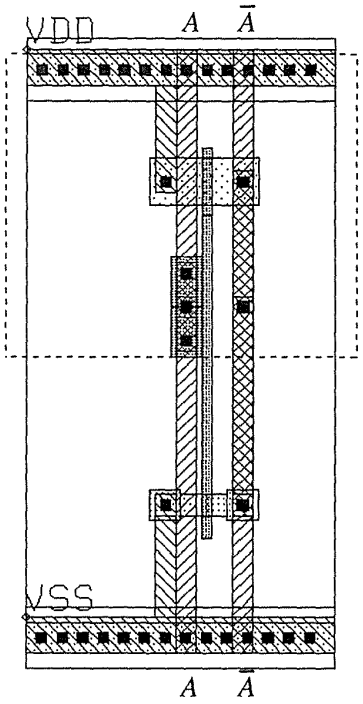



**Figure 15.17** Layout of the inverter shown in Fig. 15.16.

Suppose the device sizes of the inverter circuit in Figure 15.16a were quadrupled ($p$ = 36/2, $n$ = 12/2). Furthermore, let's assume that the maximum recommended poly1 gate width is 20 μm (due to the sheet resistance of the poly) and that exceeding that maximum could introduce significant unwanted RC delays. Let's also suppose that we are to optimize the layout for size and speed (as most digital circuits are). To meet these criteria, it will be necessary to split transistors M1 and M2 in half and lay them out as two parallel "stripes" of 18/2 for the p-channel and two parallel "stripes" of 6/2 for the n-channel. Figures 15.18a-d show the schematics, stick diagram, and layout for this scenario. The output node (drain of M1 and M2) is shared between the stripes so as to minimize the output capacitance. Taking the output in metal2 also helps in this regard. Notice that the stick diagram for this circuit looks like the previous inverter plus its mirror image along the output node. Also observe that the layout of this inverter is mirrored as shown in the stick diagram. This is a common layout technique.

Incidentally, LASICKT will need a schematic similar to Fig. 15.18d to verify the connectivity of the layout. More sophisticated (and much more expensive) CAD software could use the schematic of Fig. 15.18a.

Figure 15.19 shows stick diagrams and layouts for two more common circuits: the two-input NAND and the two-input NOR. Compare the stick diagrams of Figs. 15.19a and c to the layouts of Figs. 15.19b and d. Observe that the output nodes share the active area just as in the previous example. Also note that the spacing between the gates of the series-connected devices is minimum (for the CN20 process).

Microchip Ex. 1055, p. 328
Microchip v. Aptiv
IPR2024-00494

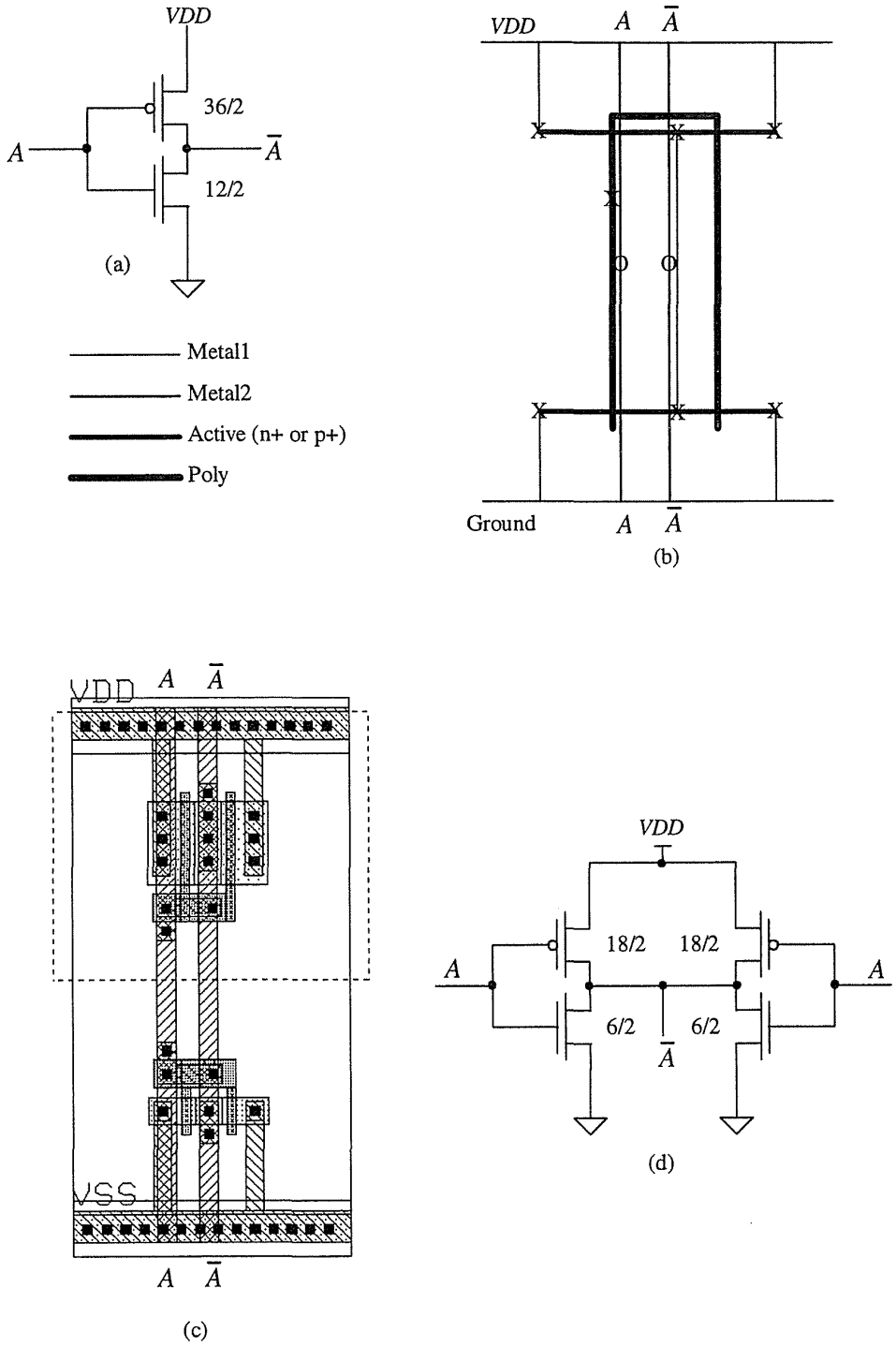



**Figure 15.18** (a) Inverter, (b) stick diagram used for layout, (c) layout, and (d) equivalent schematic.

Microchip Ex. 1055, p. 329
Microchip v. Aptiv
IPR2024-00494

Take another look at the two circuits from a geometrical rather than an electrical viewpoint. Compare the NAND gate layout to the NOR gate layout. Do you see that each can be created from the other by simply "flipping" the metal and poly connections about the x-axis?

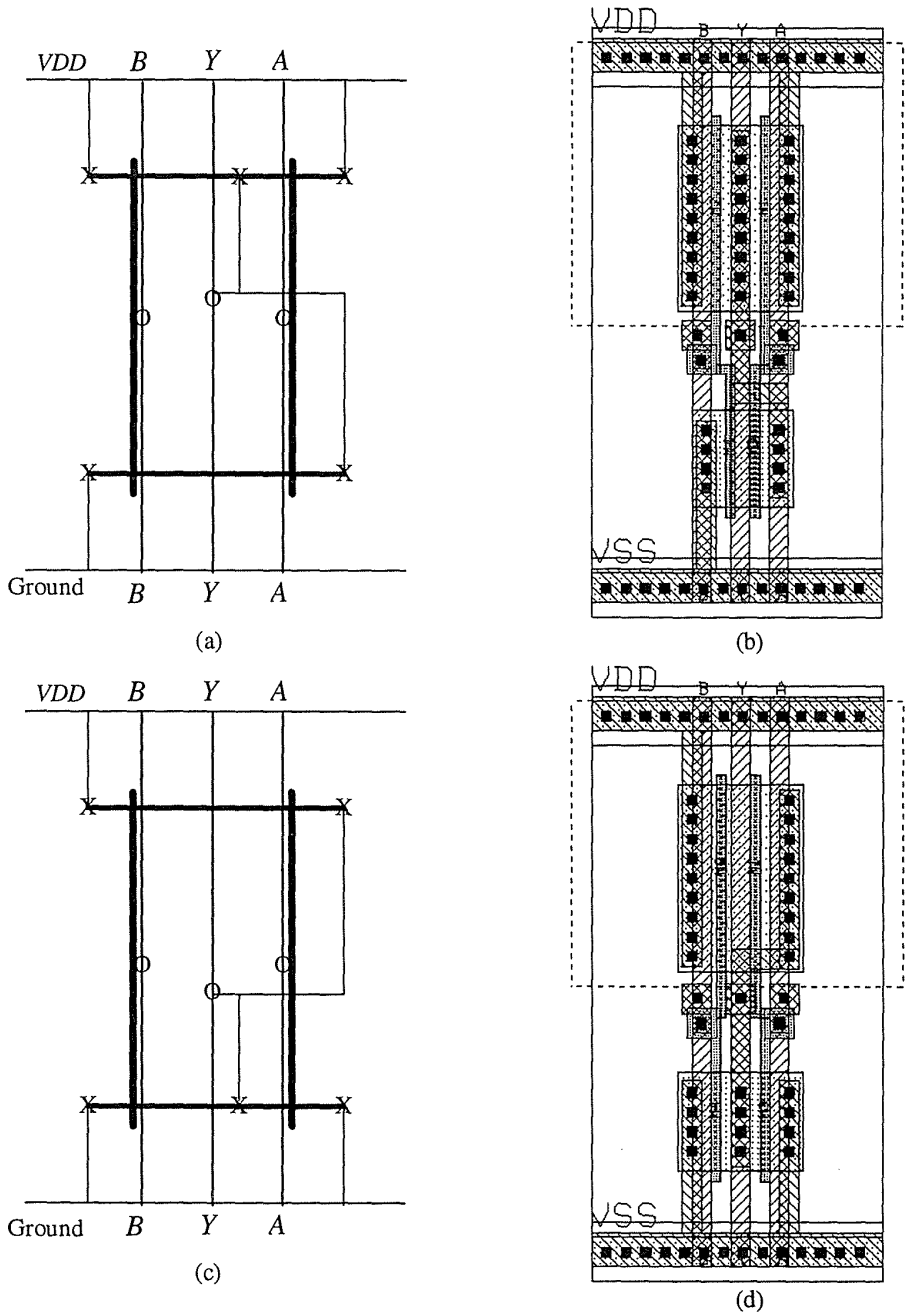



**Figure 15.19** (a) NAND stick diagram, (b) layout, (c) NOR stick diagram, and (d) layout.

Microchip Ex. 1055, p. 330
Microchip v. Aptiv
IPR2024-00494

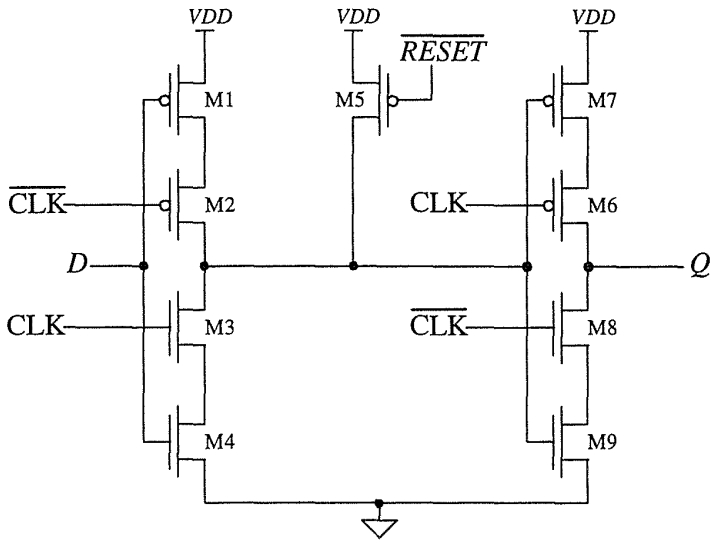



**Figure 15.20** Schematic of a dynamic register cell.

*Device Placement*

Figure 15.20 shows the schematic of a dynamic register cell, while Figs. 15.21 a-c show the stick diagrams and layout for a dynamic register. Compare the schematic of Fig. 15.20 to the stick diagram of Fig. 15.21a. We have labeled this stick diagram "preliminary" for reasons that will soon become apparent. Notice that there is a break or gap in the active area which will form our n-channel devices. Also note that the clock signals CLK and $\overline{\text{CLK}}$ must be "cross connected" from one side of the layout to the other. We don't have to think this through very far to notice that, with this placement of devices, hooking up the clock signals is going to be very difficult. Now look at the stick diagram shown in Fig 15.21b. Notice that we have rearranged the devices so that the active area is a continuous unbroken line. Normally, this "unbroken line" approach to device placement is preferred. It usually results in the most workable device placement. We say "usually" because at times your layout has to fit in an area defined by other blocks around it and you have no control over it. Also observe from Fig. 15.21b that the clock signal hook-up is more straightforward. Compare this stick diagram to the layout of Fig. 15.22c. Obviously, the device sizes used for this circuit are not practical; its purpose is merely to illustrate a layout concept. We can also see that the stick diagram is a useful tool throughout the layout process.

*Polish*

After your layout is basically finished, it is time to step back and take a look at it from a purely aesthetic point of view. Is it pleasing to the eye? Is the hook-up as straightforward as possible, or is it "busy" and hard to follow? Are the spaces between poly gates and contacts minimum? What about the space between diffusions? Are there

Microchip Ex. 1055, p. 331
Microchip v. Aptiv
IPR2024-00494

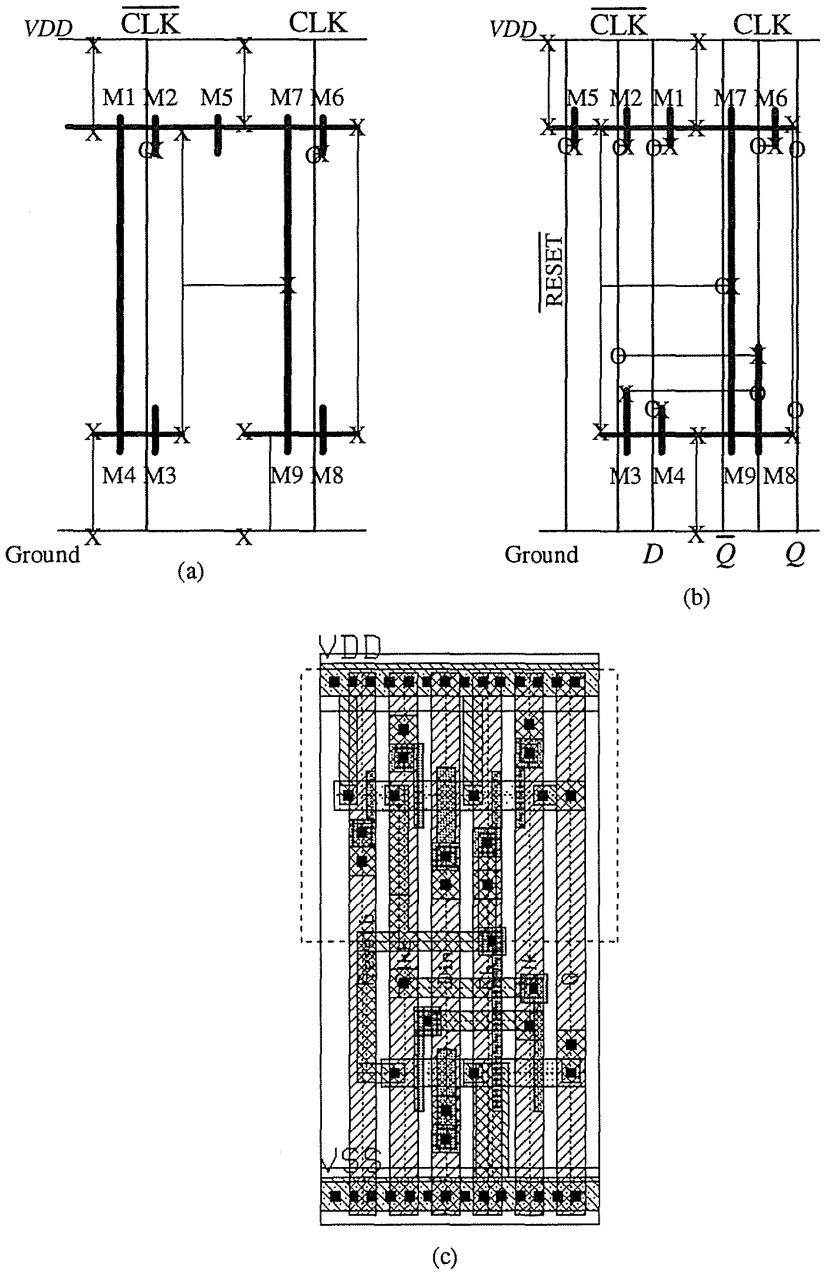



**Figure 15.21** Layout of a dynamic register cell.

enough contacts? Did you share all of the source and drain diffusions that can be shared? Are there sufficient well and substrate ties? If you have planned well and followed the plan described here, you shouldn't run into too many problems.

Microchip Ex. 1055, p. 332
Microchip v. Aptiv
IPR2024-00494

*Standard Cells Versus Full-Custom Layout*

The standard-cell approach to physical design usually dictates that cell height be fixed and the width be variable when implementing the circuit. Furthermore, standard cells are designed to abut on two sides, usually left and right, and that abutment scheme must be quite regular so that any cell can reside next to any other cell without creating a design rule violation. The standard-cell approach to layout is very useful and is always an excellent place to start. However, in the real world, area on a wafer translates directly into profit and loss (money). Wafer costs are relatively fixed whether they're blank or as tightly packed with circuitry as possible. Therefore, it follows that we want a layout that is as small as possible so that there can be as many die per wafer as possible. These are the economics of the situation. There are also technical advantages to be gained from having as small a layout as possible: interconnecting wires can be as short as possible, thereby reducing parasitic loading and crosstalk effects.

Figure 15.22 shows a typical standard-cell block that has been placed and routed by an automatic tool. Most of the individual cells have been omitted for clarity. Notice the interconnect channels between the rows of standard cells. Power, ground, and clock signal trunks run vertically to both sides of the block by means of a special cell called an "end cap." Cell rows are connected to power and ground through horizontal busses that are part of the standard cells themselves. All remaining connections are made via the routing channels. The standard-cell layouts are designed to accommodate metal2 feedthroughs that run vertically through each cell. The autorouter makes use of this space and adds the feedthroughs as needed in order to connect or pass signals from one routing channel to another. The routing channels and their associated interconnecting wires are the limiting factors for both the density and circuit performance of this type of layout.

Before we continue our discussion of relative layout density, we need to define a metric with which to quantify the matter. It is customary to use the number of transistors per square millimeter of area for this purpose. Because it is a raw number and the common denominator of all circuit layouts, we can use it even when comparing different types of circuitry or even unlike processes.

The density of the standard-cell route shown in Fig. 15.22 is approximately 5,000 transistors per square millimeter. This is fairly representative of the possible density for the channel-based routing approach and the process used (0.8 μm). Figure 15.23 shows a full-custom layout for a digital filter. The circuit area is approximately 2.1 square millimeters. The density is approximately 17,500 transistors per square millimeter, representing a 3.5-fold increase. This circuit, too, is fairly representative of the attainable density of full-custom layout using this particular 0.8 μm process. Both of these circuits were laid out using the same process, and in fact they are from the same die. The device sizes within each block would probably average out to minimum or close to minimum. The main difference affecting density is the interconnect wiring. This overhead associated with interconnect wiring is commonly referred to as the "interconnect burden." The designer must bear this burden in terms of both physical

Microchip Ex. 1055, p. 333
Microchip v. Aptiv
IPR2024-00494

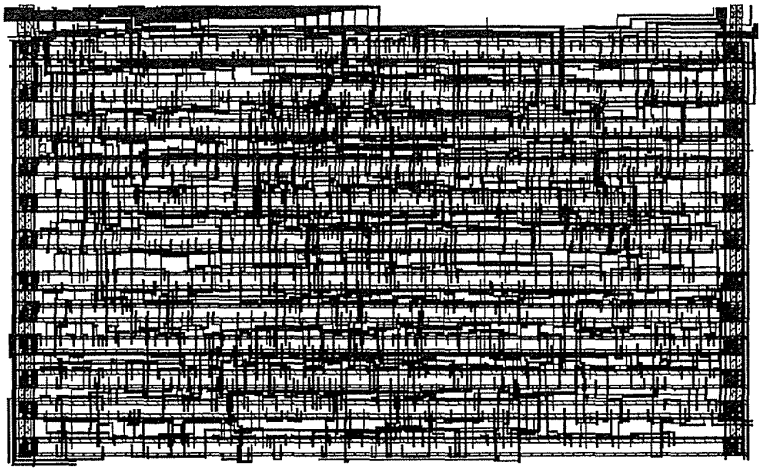


**Figure 15.22** Layout based on standard cells.

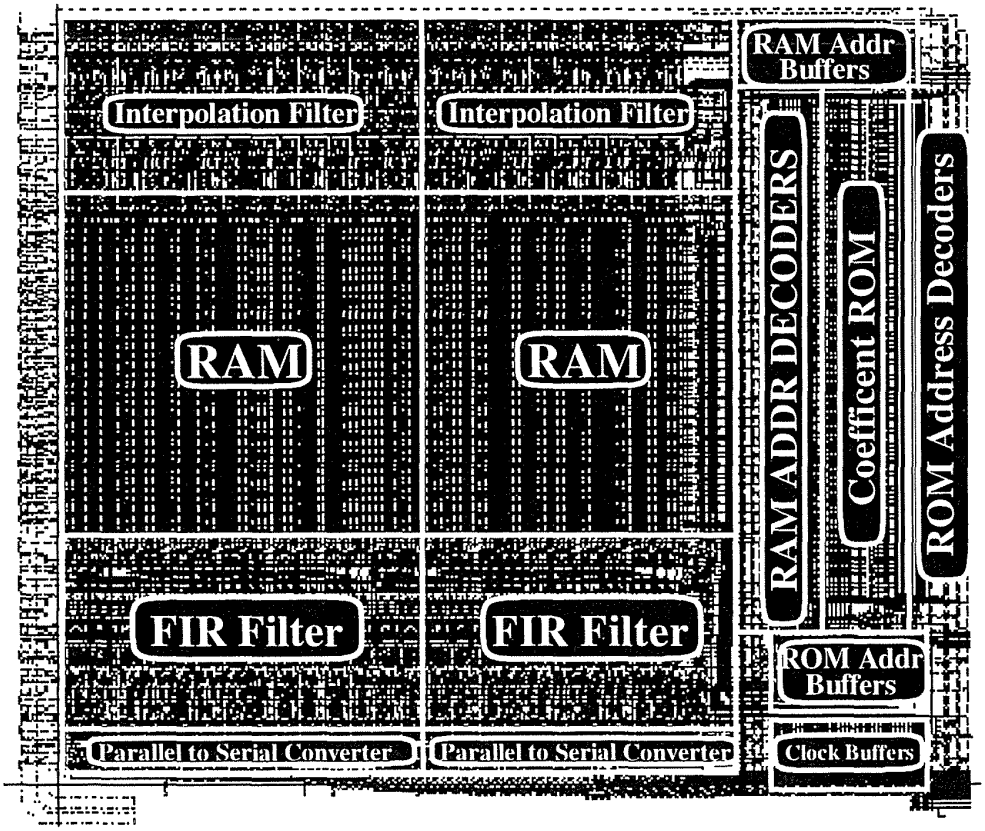



**Figure 15.23** Full-custom layout of a digital filter.

Microchip Ex. 1055, p. 334
Microchip v. Aptiv
IPR2024-00494

(wasted area) and electrical parameters (parasitic loading). Let us examine one method of creating a high-density custom layout that will minimize interconnect burden and circuit area.

Figures 15.24 a-c show a small section of the interpolation filter from Fig. 15.23. In Fig. 15.24a, we see an exploded view of four cells that form part of a data-path: an input data register, a t-gate, a full adder, and an output data register. These are instantiated (placed as a cell) twice, creating a view of eight cells. The two adder cells are slightly different: the carry inputs and outputs are on opposite sides, so that the carry-out can cascade to the carry-in of the next adder by abutting (placing next to one another) the cells. Unlike standard cells, the height and width constraints placed on custom layouts are contextual. In other words, a cell's aspect ratio depends on that of its neighbors. In this case, the width of each cell depended on the maximum allowable width of the widest cell in the group: the data register. Notice the top, bottom, left, and right boundaries of each cell in Fig. 15.24a. Data enter the register cell from the top and are output at the bottom. Clocks, power, ground, and control signals route across all the cells. The adder receives its A and B inputs from the top and outputs their SUM at the bottom. As already mentioned, carry-out and carry-in are available on the left and right edges of the adder, respectively. Figure 15.24b shows a two-bit slice of this data-path with all connections made by cell abutment. Figure 15.24c illustrates how all four edges of each cell join together to complete the hook-up.

We have seen how circuits can be implemented by means of standard cells or custom layout. The time needed to produce a standard-cell route is far less than that of

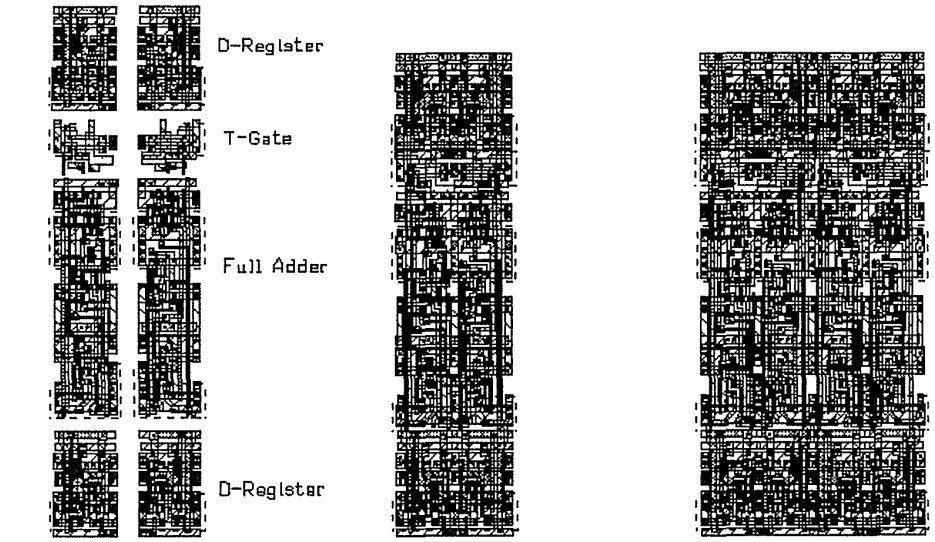



**Figure 15.24** Sections of the digital interpolation filter.

Microchip Ex. 1055, p. 335
Microchip v. Aptiv
IPR2024-00494

a full-custom implementation. The tradeoffs are area and performance. Automatic place and route tools based on routing area rather than routing channels are now coming into use. These promise a compromise solution between the two extremes. The density of their results rivals that of full-custom layout. Perhaps the hand-rendered full-custom layout will someday become a thing of the past. Nevertheless, process technology will continue to advance, circuit designers will continue to design circuits that test the outermost limits of this technology, and the marketplace will still be there demanding ever cheaper, more powerful, and faster products. It is likely then that we will all still have the opportunity to "push a polygon" or two for the forseeable future. There remains no doubt that the future will bring us ever more powerful software tools that will take over the tedious aspects of placing and connecting layouts, leaving to us the more creative aspects of planning and polishing them.

## REFERENCES

[1] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Addison-Wesley, 2nd ed., 1993. ISBN 0-201-53376-6.

[2] D. V. Heinbuch, *CMOS3 Cell Library*, Addison-Wesley, 1988. ISBN 0-201-11257-4.

[3] Kerth, Donald A. *"Floorplanning-Lecture Notes"* Crystal Semiconductor, Inc.

[4] Kerth, Donald A. *"Analog Tricks of the Trade-Lecture Notes."* Crystal Semiconductor, Inc.

[5] J. Uyemura, *Physical Design of CMOS Integrated Circuits Using L-EDIT*, PWS Publishing Co., 1995. ISBN 0-534-94326-8.

## PROBLEMS

**15.1** The standard-cell height can be reduced to make standard-cell-based layouts smaller (Fig. P15.1). Using this cell as an approximate height reference, lay out a double inverter.

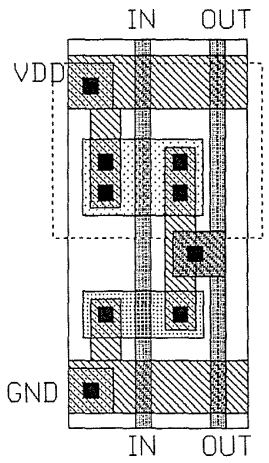



**Figure P15.1**

Microchip Ex. 1055, p. 336
Microchip v. Aptiv
IPR2024-00494

**15.2** Repeat Problem 15.1 for a two-input NAND gate.

**15.3** Repeat Problem 15.1 for a two-input NOR gate.

**15.4** Repeat Problem 15.1 for a transmission gate (with the same functionality as Fig. 15.5).

**15.5** Repeat Problem 15.1 for a NAND-based SR flip-flop.

**15.6** Repeat Problem 15.1 for a NOR-based SR flip-flop.

**15.7** Design, lay out, and simulate the operation of a D FF to replace the one described in Fig. 15.12. Assume that the FF uses mainly minimum-size MOSFETs and that a pass transistor is used for the clocking element.

**15.8** Lay out the two-input MUX using TGs shown in Fig. P15.8.

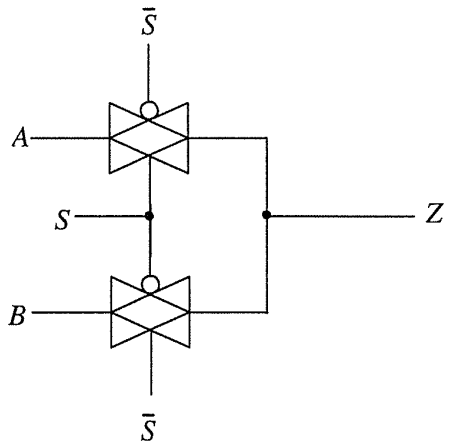



**Figure P15.8**

**15.9** Using the reduced standard-cell frame height described in Problem 15.1, lay out the XOR/XNOR gates of Fig. P15.9. Assume that both the inputs and outputs to the gates are on poly1.

**15.10** Lay out a 16-to-1 MUX/DEMUX based on the layout topology given in Fig. 15.15. Lay out another 16 to 1 MUX/DEMUX in as small an area as possible. DRC your final layouts.

**15.11** Sketch a stick diagram for the layouts of Fig. 15.5.

**15.12** Point out the high-impedance nodes for the schematic of Fig. 15.20. Discuss the concerns one must consider when laying out a circuit with high-impedance nodes.

**15.13** List and discuss three reasons to have small layout size.

**15.14** Lay out the D FF shown in Fig. P15.14. Show the stick diagram for your layout.

Microchip Ex. 1055, p. 337
Microchip v. Aptiv
IPR2024-00494

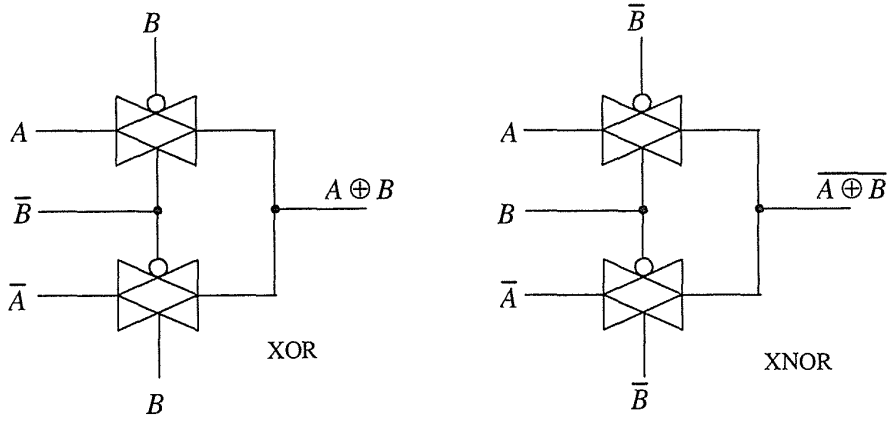



**Figure P15.9**

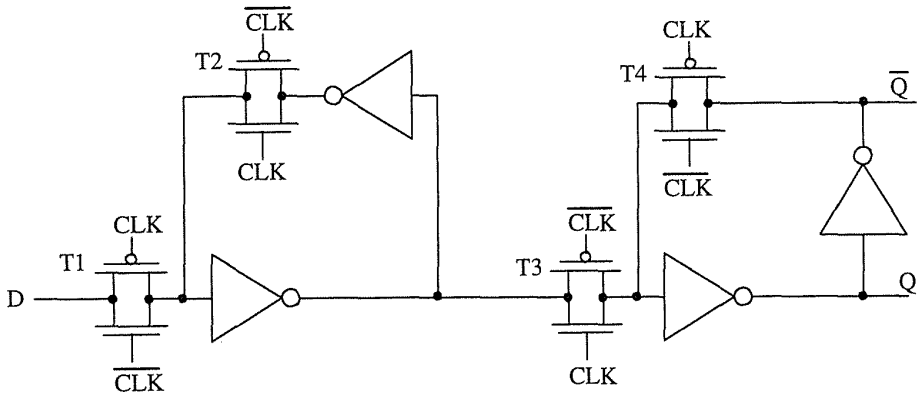



**Figure P15.14**

Microchip Ex. 1055, p. 338
Microchip v. Aptiv
IPR2024-00494

Chapter

16

# BiCMOS Logic Gates

Modern BiCMOS technology began in the early 1980s with high expectations [1]. The name BiCMOS comes from the fact that the logic is made using CMOS and bipolar junction transistors (BJT). The BJTs are used for their high-current capability, while CMOS is used because of its small layout size and ease of implementing logic. With the best of both worlds on a single substrate, high-speed, high-current-driving bipolar transistors and low-power, high-impedance CMOS devices, every major semiconductor foundry now possesses some form of BiCMOS process. Strategies for developing BiCMOS have evolved from the bipolar and the CMOS directions, with advantages and disadvantages associated with each. Bipolar device capabilities have been added to some CMOS processes to improve speed, while CMOS device capabilities have been added to some bipolar processes to minimize power dissipation. A chart comparing CMOS, BiCMOS, and bipolar (with $I^2L$) technologies can be seen in Fig. 16.1 [2,3]. This chapter focuses on the CMOS process with bipolar capabilities. Although the CN20 process is not a true BiCMOS process, it does contain some BJT options that will allow demonstration of basic digital BiCMOS circuit design. It should be noted that the CMOS14TB process contains no provisions for BJT devices.

Microprocessors are particularly well suited for BiCMOS technology. Typically, three generic categories limit microprocessor performance [1]: (1) Instructions per task, (2) cycles per instruction, and (3) time per cycle. The third category can be greatly improved by increasing the speed critical blocks. A PC microprocessor [4] was developed using a bipolar-based BiCMOS process. Operating at 533 MHz, the microprocessor used high-density CMOS devices that were added to a bipolar process. The floor plan can be seen in Fig. 16.2 [5] in which the speed critical blocks such as the integer and floating point units utilized BJT transistors, while power-consuming cache arrays and I/O cells (for system compatibility) were constructed using CMOS technology.

Microchip Ex. 1055, p. 339
Microchip v. Aptiv
IPR2024-00494

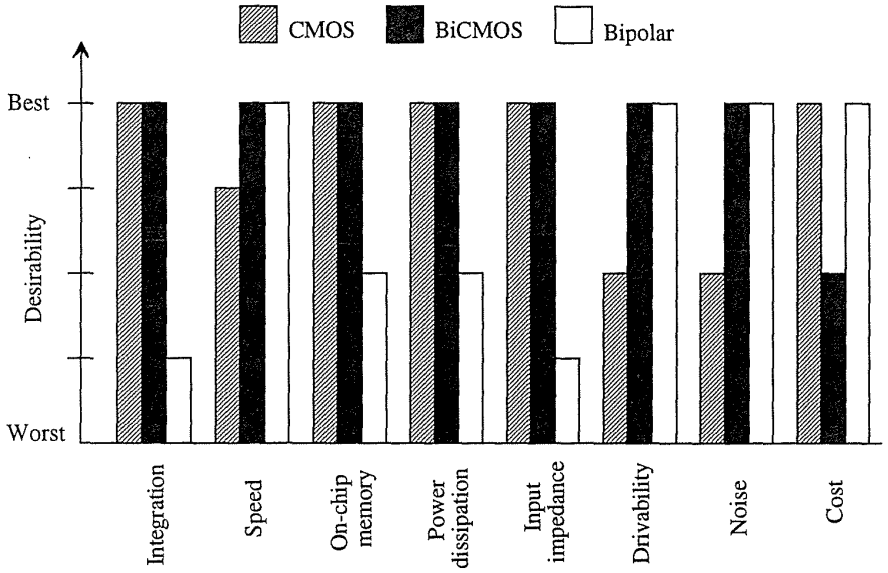



**Figure 16.1** Comparison of CMOS, BiCMOS, and bipolar technologies.

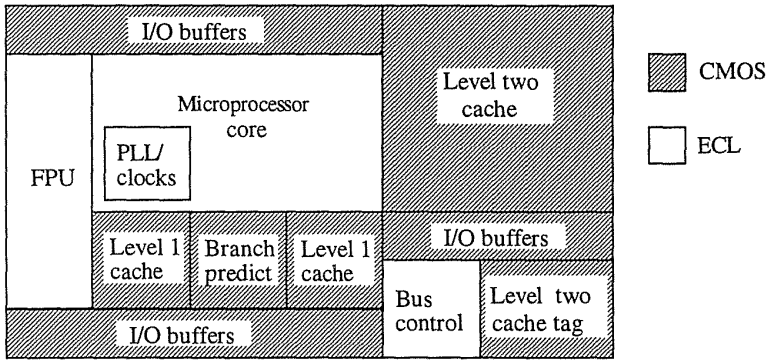


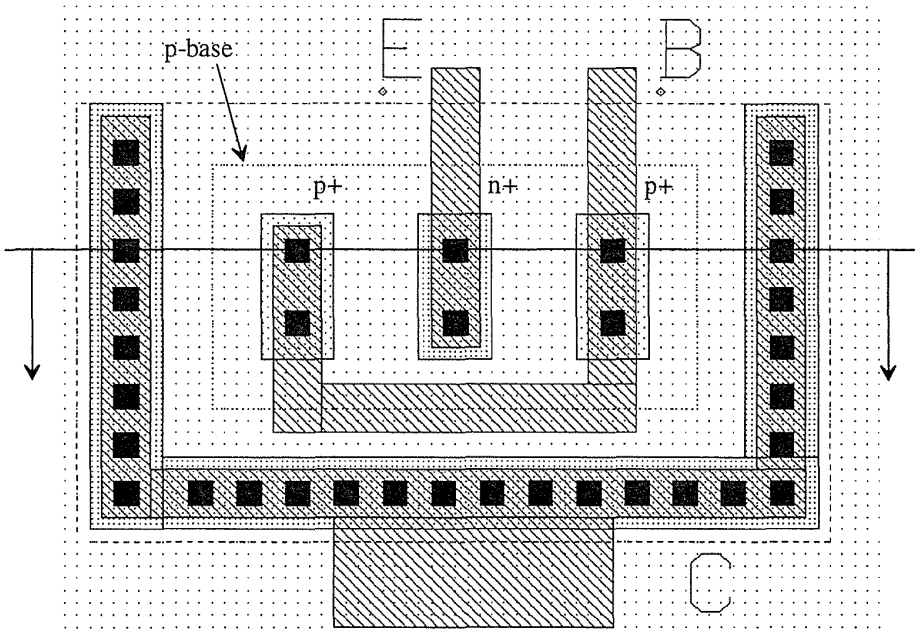

**Figure 16.2** Floor plan of a high-performance BiCMOS microprocessor.

## 16.1 Layout of the Lateral BJT

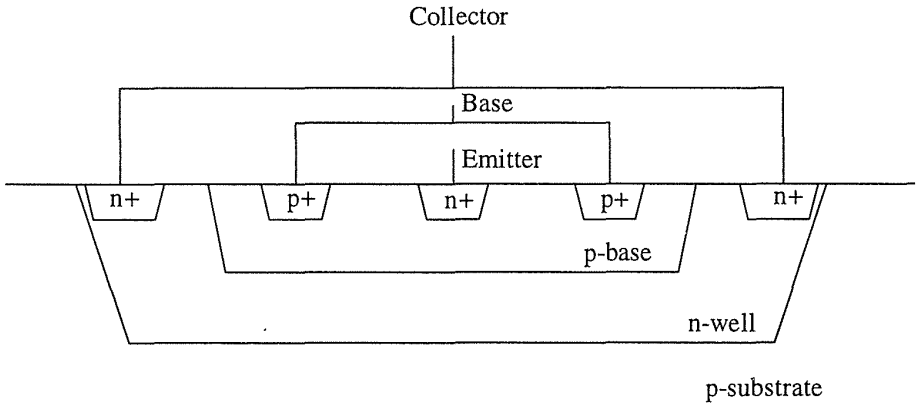
We begin this chapter with a discussion of the lateral NPN bipolar junction transistor that is available in CN20 with the addition of a p-diffusion layer called p-base. The layout and operation of the lateral BJT are discussed. Most of the applications of the BJT in digital circuits are in the design of buffer circuits for driving large capacitive loads. The BJT buffer requires less area than the CMOS counterpart.

The lateral NPN BJT in the CN20 process uses the n-well as the collector, the p-base diffusion for the base, and the n+ implant for the emitter. The layout of a 2 x 1

Microchip Ex. 1055, p. 340
Microchip v. Aptiv
IPR2024-00494




(a)




(b)

**Figure 16.3** Layout of a 2 x 1 lateral NPN BJT (a) and associated cross-sectional view (b).

Microchip Ex. 1055, p. 341
Microchip v. Aptiv
IPR2024-00494

BJT is shown in Fig. 16.3a. The first number in the description of the BJT indicates the number of contacts on the emitter of the BJT, while the second number indicates the number of fingers. The cross-sectional view of the 2 x 1 BJT, is shown in Fig. 16.3b. Note that the metal1 connection to well and p-base require n+ and p+ implants, respectively. The BJT discussed here is called lateral because the majority of the emitter current flows out the sides of the n+ implant emitter to the n-well collector. In a bipolar only process or a true BiCMOS process, the current flow is in the vertical direction. Because the CN20 process uses the n-well as the collector, only a NPN transistor can be formed. If the process possessed a p-well and an n-base layer, a PNP transistor could also be formed.

The collector, made with the n-well, of the lateral NPN transistor forms a diode, with the p-type substrate limiting the negative values of collector voltages allowable. (This is very important to keep in mind.) We also know that the sheet resistance of the n-well is on the order of 2,500 Ω/square. For the 2 x 1 BJT shown in Fig. 16.3, the series collector resistance is in the neighborhood of 500 Ω. The U-shaped connection of n+ to the well is used to reduce this resistance and provide better collection of carriers. The p-base diffusion used to make the base of the NPN is also highly resistive. The base contacts on each side of the emitter, instead of on a single side, are used to reduce the base spreading resistance. Typical values of the base spreading resistance, for this small lateral NPN, are 1,000 Ω. The emitter series resistance is tens of ohms and is usually negligible compared to the collector and base series resistances. In short, the problem with using a lateral NPN transistor in a CMOS process to design BiCMOS circuits is the associated large parasitic resistances.

The design rules for the p-base layer are shown in Fig. 16.4. These rules correspond to checks 33 through 36 in the CN20.DRC file.

## 16.2 Modeling the Lateral NPN

The lateral NPN bipolar transistor operation is very similar to normal BJT operation, with the exception of large parasitic resistances associated with the base and collector. The symbol for a lateral BJT is identical to the normal BJT symbol and is seen in Fig. 16.5.

To develop a digital model for the BJT which is similar to the model we developed for the MOSFET, we will assume that the effective switching resistance between the collector and the emitter is simply $R_c$. We can define the variable $R_{npn}$ by

$$R_{npn} = R_c \tag{16.1}$$

The input resistance of the lateral BJT can be estimated by

$$R_{innpn} = R_b \tag{16.2}$$

A SPICE model for the 2 x 1 BJT in the CN20 process is located on the accompanying disk in the file "spice.inf" in the \CN20\ directory.

Microchip Ex. 1055, p. 342
Microchip v. Aptiv
IPR2024-00494

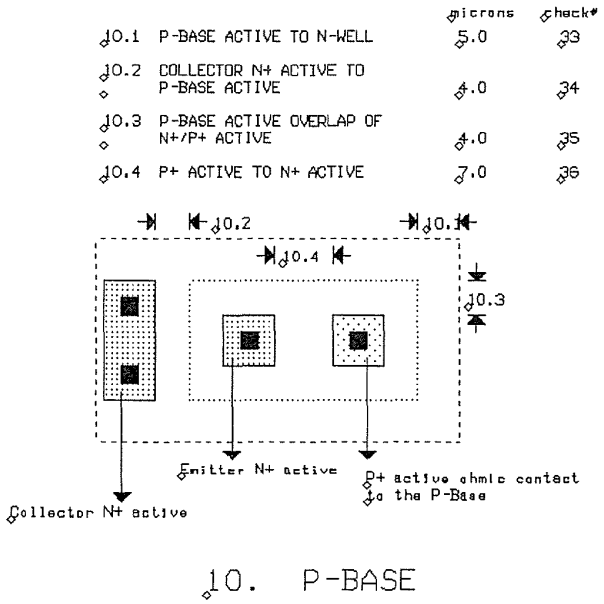



**Figure 16.4** Design rules for the p-base layer.

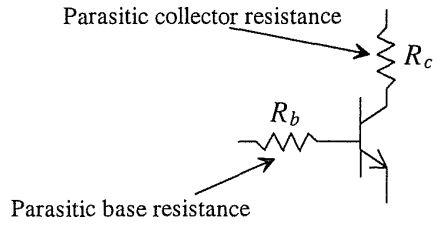



**Figure 16.5** Symbol of lateral BJT showing parasitic resistances.

**Example 16.1**

Estimate the falltime of the circuit shown in Fig. 16.6. Compare your hand calculations to SPICE. Note that the input 5 V pulse is applied directly to the base of the BJT.

From the SPICE model located in spice.inf $R_c$ = 420 Ω and $R_b$ = 1.2 kΩ. The high-to-low time can be estimated by

$$t_{PHL} = 420 \cdot 5 \text{ pF} = 2.1 \text{ ns}$$

The base current that must be supplied from the input voltage source is 5/1.2 k or approximately 4 mA. The simulation results are shown in Fig. 16.7. ■

It should be pointed out that the BJT of this example was driven with an ideal voltage source, something that is not available in practice. However, the collector

Microchip Ex. 1055, p. 343
Microchip v. Aptiv
IPR2024-00494

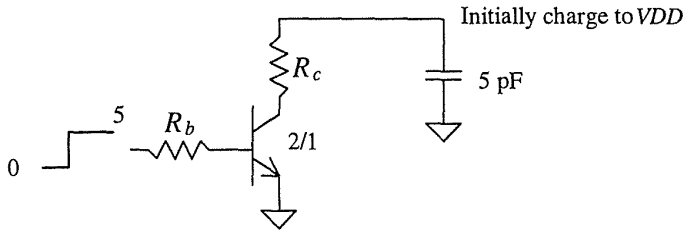



**Figure 16.6** Circuit used in Ex. 16.1.

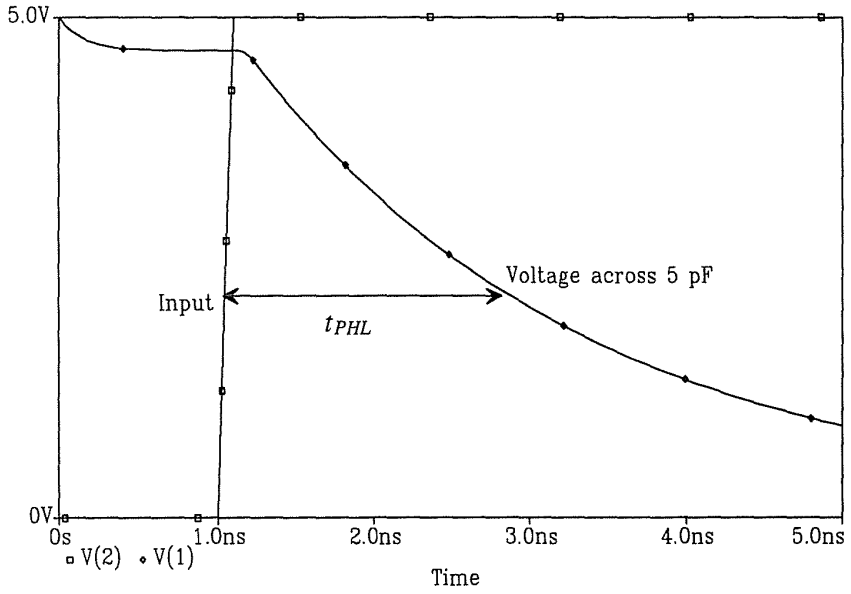



**Figure 16.7** Simulation results for Ex. 16.1.

current supplied for an input voltage (the voltage applied to the base of the transistor) of *VDD* is significantly larger than the drain current supplied using the MOSFET switch.

Modeling larger lateral BJTs using the 2 x 1 model can be accomplished using the *A* (area) parameter in SPICE. For example, if you lay out a 2 x 2 BJT, that is, two emitter contacts with two emitter "fingers," we simply set $A = 2$ and use the 2 x 1 model. For a 10 x 3 BJT we set $A = 15$. Here *A* can be thought of as the ratio of the large BJT to the size of the 2 x 1 BJT model. An example of the layout of a 5 x 2 BJT is shown in Fig. 16.8. When using the larger BJT models Eqs., (16.1) and (16.2) are modified by *A*. The resistances now become

$$R_{npn} = \frac{R_c}{A} \tag{16.3}$$

and

Microchip Ex. 1055, p. 344
Microchip v. Aptiv
IPR2024-00494

$$R_{innpn} = \frac{R_b}{A} \tag{16.4}$$

Here $R_c$ and $R_b$ are the series resistances specified in the SPICE model used. If $A$ is not specified, SPICE assumes $A = 1$.

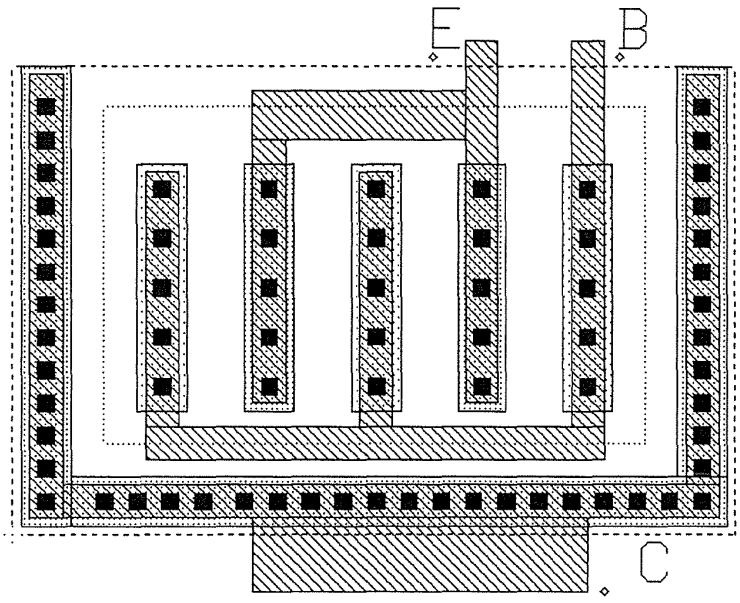



**Figure 16.8** Layout of a 5 x 2 lateral NPN transistor.

*BJT Capacitances*

The BJT capacitances result from the depletion capacitances of the implant regions and from the forward-biased base-emitter junction (the storage capacitance). The base-collector depletion capacitance is estimated either from the SPICE model or from use of the n+ to p-substrate depletion capacitance specified in the n-channel BSIM model. The base-emitter depletion capacitance is estimated using the depletion capacitance from the p+ to n-well specified in the p-channel BSIM model.

More important than these depletion capacitances is a storage capacitance associated with the base-emitter forward-biased diode. If the minority carrier lifetime of the base-emitter junction is called $\tau_F$, then, from Ch. 2, the storage capacitance is given by

$$C_{bestor} = \tau_F \cdot \frac{I_E}{V_T} \tag{16.5}$$

Here $I_E$ is the DC emitter current and $V_T$ is the thermal voltage ($kT/q$). As the emitter current increases, the storage capacitance increases. The SPICE model supplied in the

Microchip Ex. 1055, p. 345
Microchip v. Aptiv
IPR2024-00494

file "spice.inf" does not specify the $\tau_F$ parameter, so simulations will not show charge storage effects (such as the transistor not turning off until the stored charge is removed from the base-emitter junction).

## 16.3 The BiCMOS Inverter

A BiCMOS inverter is shown in Fig. 16.9. One important aspect of the basic BiCMOS inverter is that the output cannot go to *VDD* or ground, as was the case in the CMOS inverter. This has the effect of lowering the noise margins of the logic. The maximum output voltage is approximately *VDD* – 0.7 V, while the minimum logic output voltage is approximately 0.7 V. The 0.7 V drop for the high and low side comes from the base-emitter voltage drop of Q2 and Q1, respectively. Caution should be exercised when using the output of BiCMOS gates with CMOS logic. The low-output voltage of 0.7 V is very close to the threshold voltage of the n-channel transistor. CMOS gates with switching point voltages close to the threshold voltage are susceptible to noise.

To understand the operation of the basic BiCMOS inverter, consider the case when the input is grounded. MOSFETs M4 and M1 are on, while M2 and M3 are off. The lateral BJT Q1 is off, its base being held at ground potential by M1. The base of Q2 is held at *VDD* by M4. The output voltage is *VDD* – 0.7 V or a logic high. When the input of the BiCMOS inverter is held high, MOSFETs M3 and M2 are on, while M4 and M1 are off. The base of Q2 is held at ground potential, so it is off. Since M2 is on, the output is connected to the base of Q1. This causes Q1 to turn on and pull the output voltage down to 0.7 V.

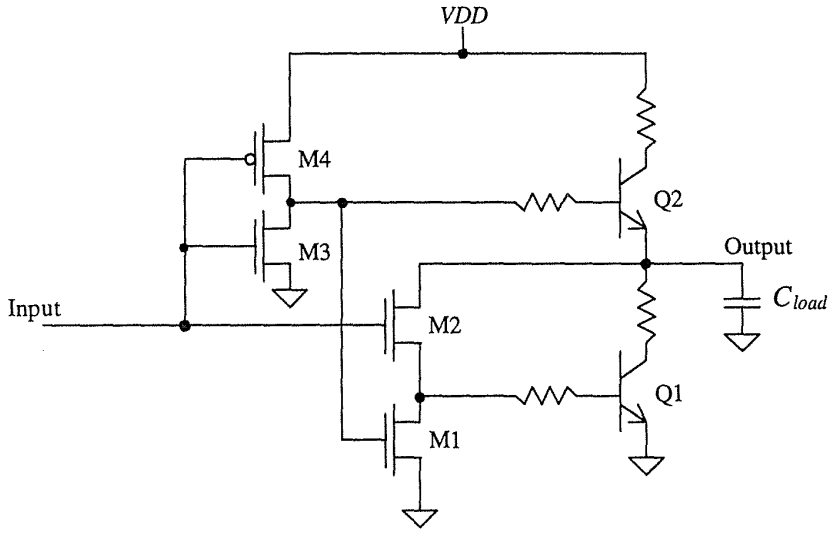



**Figure 16.9** Basic BiCMOS inverter showing parasitic collector and base resistances.

Microchip Ex. 1055, p. 346
Microchip v. Aptiv
IPR2024-00494

*Switching Characteristics*

The delay associated with the BiCMOS inverter discharging a capacitance, $C_{load}$, consists of two parts: the delay in Q1 turning on and the delay once Q1 is on discharging $C_{load}$. The delay associated with discharging $C_{load}$ is given by

$$t_{PHL} = R_{npn} \cdot C_{load} \tag{16.6}$$

The delay associated with turning Q1 on is comparable to the minority carrier lifetime $\tau_F$, which is in general well under 1 ns. The MOSFET M2 is used to turn Q1 on. A general design rule can be applied to the sizing of M2. We require the effective resistance of M2, $R_{n2}$ to be equal to the base resistance of Q1, $R_b$. This is a very simple method of sizing the transistors, which gives very good results.

The low-to-high delay-time can be estimated in much the same way as the high-to-low delay. The delay in charging $C_{load}$ is given by

$$t_{PLH} = R_{npn} \cdot C_{load} = t_{PHL} \tag{16.7}$$

The fact that the propagation delay-times are equal is a side benefit of the BiCMOS inverter. The MOSFET M4 drives the base of Q2. Sizing M4's effective resistance equal to the base resistance of Q2 is a general design rule for the sizing of the MOSFETs. MOSFETs M1 and M3's effective resistance can be made larger than M2 and M4's effective resistance because these MOSFETs simply hold the bases of Q1 and Q2 at ground during the off period. In general, to attain a switching point voltage close to *VDD*/2, M2 and M4 are sized the same as M3.

---

**Example 16.2**

Estimate and simulate the delay of the BiCMOS inverter (Fig. 16.10) driving a 10 pF load. Calculate the input capacitance of the inverter.

The propagation delays are simply given by

$$t_{PHL} = t_{PLH} = R_{npn} C_{load} = 420 \cdot 10 \text{ pF} = 4.2 \text{ ns}$$

The SPICE simulation results are shown in Fig. 16.11. Notice how the output of the BiCMOS inverter stays at approximately 0.7 V from the power and ground rails.

The input capacitance is the sum of the input capacitances of the three MOSFETs connected to the input, or

$$C_{in} = \frac{3}{2} \cdot 800 \frac{\text{aF}}{\mu\text{m}^2} (2 \cdot 10 + 2 \cdot 10 + 16 \cdot 2) = 86.4 \text{ fF}$$

This input capacitance can easily be driven from logic on chip without excessive delay. The delay of this inverter driving a 10 pF load can be reduced by using larger BJTs with corresponding larger MOSFET drivers.

Microchip Ex. 1055, p. 347
Microchip v. Aptiv
IPR2024-00494

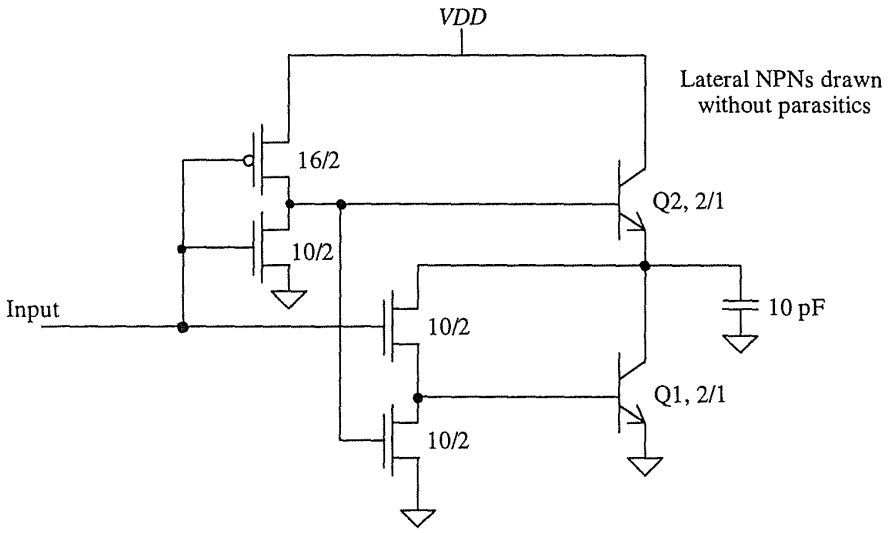



**Figure 16.10** Circuit used in Ex. 16.2.

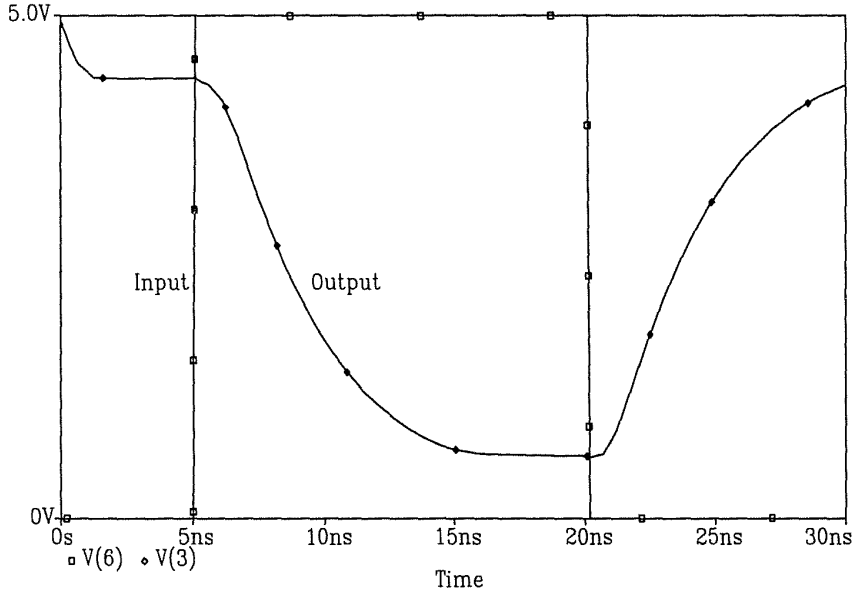



**Figure 16.11** Simulation results of a BiCMOS inverter driving a 10 pF capacitive load.

An important characteristic of the BJT has been neglected in the analysis and in the SPICE models. The carrier lifetime, $\tau_F$ (resulting in a storage capacitance), was not modeled, and the actual delays can be longer than predicted by SPICE.■

Microchip Ex. 1055, p. 348
Microchip v. Aptiv
IPR2024-00494

*Full-Swing BiCMOS Inverters*

Figures 16.12 and 16.13 show two implementations of a full-swing BiCMOS inverter [4, 5]. Consider the first inverter (Fig. 16.12) with its input grounded. MOSFETs M2 and M4 are off while MOSFET M5 is on. MOSFETs M1 and M3 can be thought of as resistors. Since M5 is on, the base of Q2 is pulled to *VDD*. The transistor Q2 is on and pulls the output to *VDD* – 0.7. MOSFET M3, which behaves like a resistor, then pulls the output up to *VDD*. When the input to the inverter is high, M2 and M4 are on and M5 is off. This pulls the base of Q2 to ground, turning it off. At the same time M2 turns on, with the output high, causing Q1 to turn on. Q1 pulls the output down to 0.7 V. From there M1, which behaves like a resistor, pulls the output down to ground. Note that if M1 or M3 does not have a large effective resistance (long L), the circuit will not operate correctly.

The inverter shown in Fig. 16.13 performs better (lower dynamic power dissipation) than the inverter of (a) at the cost of more complicated circuit design. The inverter operation is similar to that given in (a) with the exception that switches M1 and M3 are controlled by the output of the inverter. This topology improves performance by making the MOSFET sizes less critical.

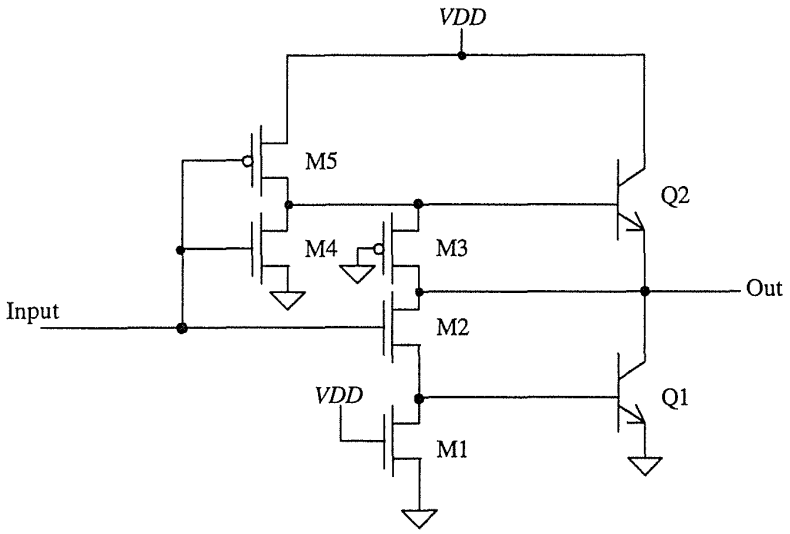



**Figure 16.12** A wide-swing BiCMOS inverter.

## 16.4 Other BiCMOS Logic Gates

The NAND and NOR gates using BiCMOS technology [6, 7] are easily constructed from the standard CMOS static logic gate. As seen in Fig. 16.14, blocks $P_1$ and $P_2$ are identical to the all CMOS NAND gate presented in Ch. 12 (Fig. 12.1). Node C

Microchip Ex. 1055, p. 349
Microchip v. Aptiv
IPR2024-00494

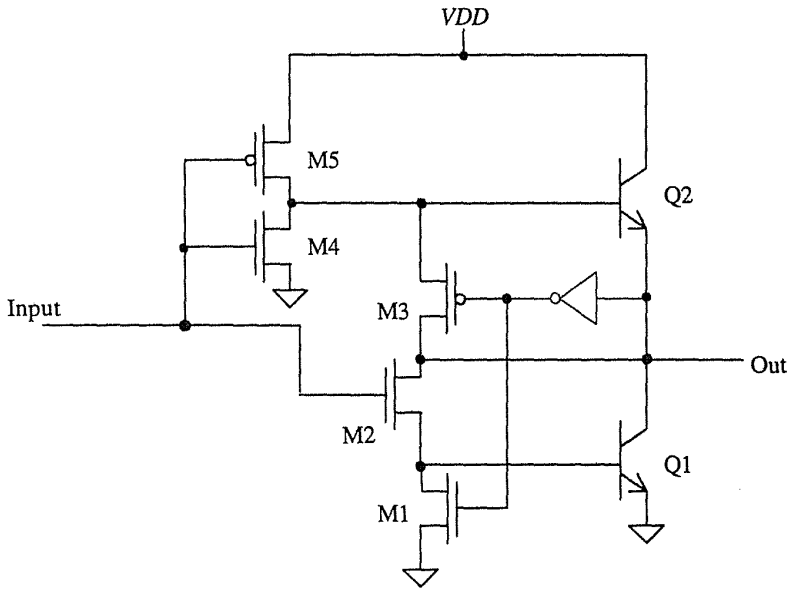



**Figure 16.13** A full-swing BiCMOS inverter with improved power dissipation.

represents the output of the all CMOS NAND. Therefore, the output of the BiCMOS gate is identical to the logic state of node C. The transistor pair composed of M5 and M6 ($P_3$) ensures that if the output is low (both inputs are a logic 1), the sinking transistor, Q1, is biased correctly to discharge the load capacitor. The transistor, M7, connects the base of Q1 to *VSS* (or ground) only if the output is high. Examine Table 16.1 for the states of the important nodes and transistors of the NAND gate.

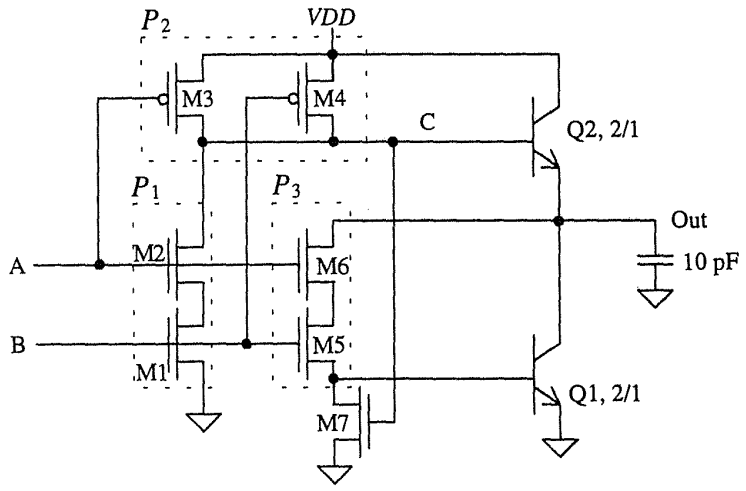



**Figure 16.14** A BiCMOS NAND gate.

Microchip Ex. 1055, p. 350
Microchip v. Aptiv
IPR2024-00494

| A B | Node C | M5 | M6 | M7 | Q1 | Q2 | Out |
|---|---|---|---|---|---|---|---|
| 0 0 | 1 | OFF | OFF | ON | OFF | ON | 1 |
| 0 1 | 1 | ON | OFF | ON | OFF | ON | 1 |
| 1 0 | 1 | OFF | ON | ON | OFF | ON | 1 |
| 1 1 | 0 | ON | ON | OFF | ON | OFF | 0 |

**Table 16.1** States of key nodes and components in Fig. 16.14.

The basic BiCMOS NOR gate is seen in Fig. 16.15. Notice again that the blocks P1 and P2 represent the all CMOS NOR presented in Ch. 12 (Fig. 12.1) and that the logic state of node C will again be identical to the output of the BiCMOS gate. Transistor block $P_3$, composed of M5 and M6, allows Q1 to be biased correctly in order to discharge the output capacitance, while M7 again ensures that the base of Q1 is connected to ground should output be a logic 1. Table 16.2 shows that states of the critical nodes and transistors in Fig. 16.15.

| A B | Node C | M5 | M6 | M7 | Q1 | Q2 | Out |
|---|---|---|---|---|---|---|---|
| 0 0 | 1 | OFF | OFF | ON | OFF | ON | 1 |
| 0 1 | 0 | ON | OFF | OFF | ON | OFF | 0 |
| 1 0 | 0 | OFF | ON | OFF | ON | OFF | 0 |
| 1 1 | 0 | ON | ON | OFF | ON | OFF | 0 |

**Table 16.2** States of key nodes and components in Fig. 16.15.

The switching analysis for these gates is very similar to the BiCMOS inverter since the delay-time is dominated by the time necessary to charge and discharge the load capacitance.

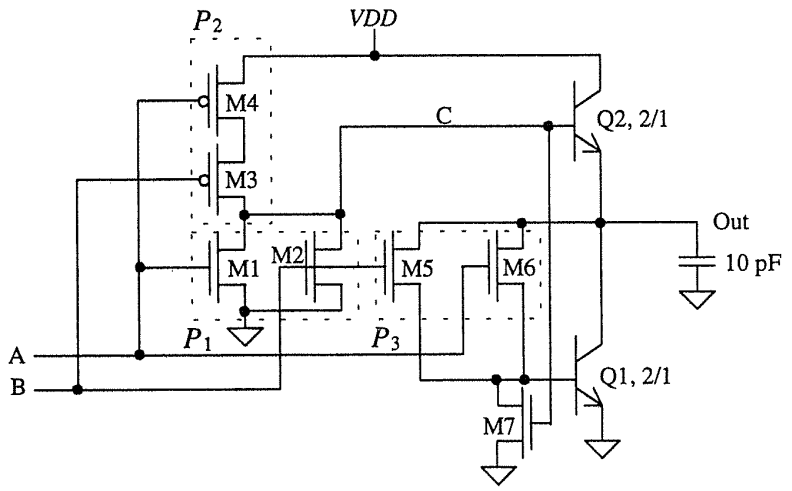



**Figure 16.15** A BiCMOS NOR gate.

Microchip Ex. 1055, p. 351
Microchip v. Aptiv
IPR2024-00494

## 16.5 CMOS and ECL Conversions Using BiCMOS

One major advantage of using BiCMOS is the ability to mix technologies such as CMOS, ECL, and BiCMOS logic. One advantage of using emitter-coupled logic (ECL) circuits is that the bipolar transistors can double their output current for every 25 mV of change in the base-emitter voltage. This is simply because the collector current, $I_C$, through a BJT, can be described as

$$I_C = I_S e^{(v_{BE}/V_T)} \tag{16.8}$$

where $I_s$ is the saturation current, $V_T$ is the thermal voltage, and $v_{BE}$ is the instantaneous base-emitter voltage. The expression for the transconductance, which relates the amount of drive current to the input voltage, is

$$g_m(BJT) = \frac{I_C}{V_T} \text{ A/V} \tag{16.9}$$

and is also obviously exponential. As a result, the BJTs can sink or source large amounts of load currents with very small input voltage swings.

Now examine the relationship between gate-to-source voltage and drain current through an MOS device:

$$I_D \approx \frac{\beta}{2}(V_{GS} - V_{THN})^2 \tag{16.10}$$

The transconductance for the MOSFET is

$$g_m(MOS) = \sqrt{I_D 2\beta} = \beta(V_{GS} - V_{THN}) \text{ A/V}$$

which is linear with respect to the input voltage. The amount of input voltage necessary to switch an output from a low to a high or a high to a low is much greater than the BJT case. Typically, the CMOS input signal must overcome the threshold voltage by around 1 V before the output begins to change state. The different switching speeds can be seen in Fig. 16.16 [5].

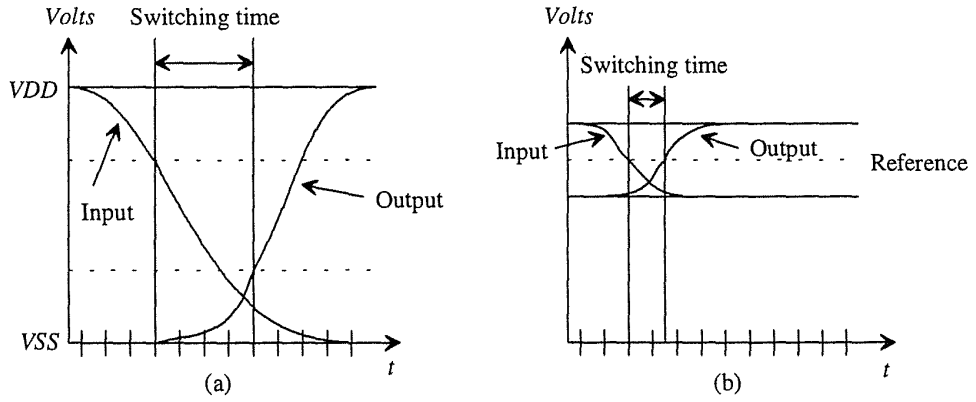



**Figure 16.16** Switching times for (a) CMOS logic and (b) ECL.

Microchip Ex. 1055, p. 352
Microchip v. Aptiv
IPR2024-00494

Notice the differing reference levels shown between the two types of technologies. While CMOS logic typically swings between *VDD* and *VSS*, ECL logic has a much smaller signal swing defining the logic levels. Interface circuits are needed that convert ECL to CMOS logic levels and vice versa.

An ECL to CMOS converter circuit can be seen in Fig. 16.17 [3, 8]. Here, the ECL input signal is level shifted by 2 $V_{BE}$ drops to a current-mode logic (CML) circuit that drives the CMOS output shifter stage. The stepped-down ECL input causes the CML circuit to become imbalanced so that one collector is considered high, while the other collector is considered low. The critical issues in minimizing the delay time are the output swing of the CML stage and the sizes chosen for the CMOS shifter. However, the two specifications are inversely proportional, for increasing the output swing of the CML stage decreases the delay through the CMOS shifter but increases the delay through the CML stage.

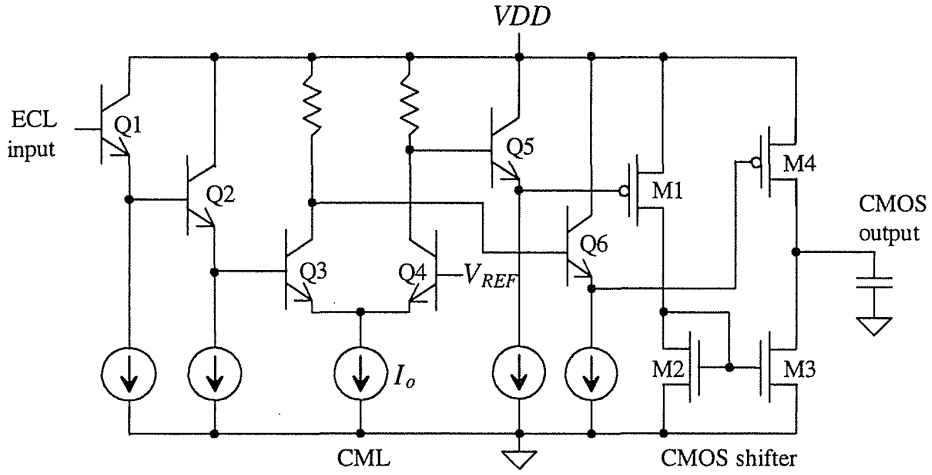



**Figure 16.17** ECL to CMOS conversion circuit.

An improved ECL to CMOS translator can be seen in Fig. 16.18 [9]. This circuit has significantly reduced total delay and power dissipation and improved driving capability for capacitive loads over the circuit shown in Fig. 16.17. However, a process allowing both NPN and PNP BJT devices is required. The input can be either complementary or single-ended (with one input tied to a reference voltage). Notice that the two complementary differential input stages drive the output BJT devices. One reason why this circuit has such improved delay-time is that the base of the output transistors, Q5 and Q6, are driven directly from their respective differential input stages. For example, if the ECL inputs were such that ECL1 was high and ECL2 was low, then all the current through M1 would flow through Q4 and into the base of Q6, thus discharging the output node very rapidly. Similarly, if ECL1 was low and ECL2 was high, then all the quiescent current through M2 would flow through Q2, which results in a large amount of current through Q5, thus charging the output node quickly.

Microchip Ex. 1055, p. 353
Microchip v. Aptiv
IPR2024-00494

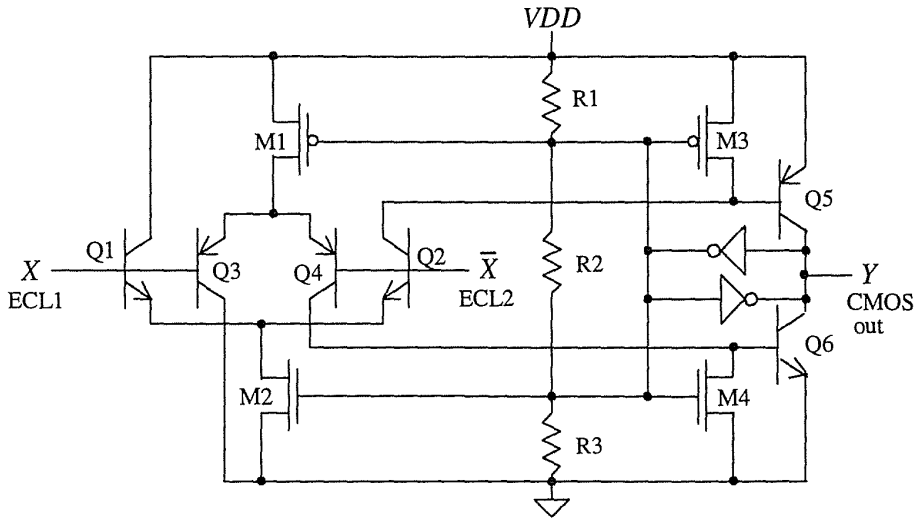



**Figure 16.18** An improved ECL to CMOS conversion circuit.

Once the output devices, Q5 and Q6, are turned on, they become saturated until the feedback circuit, composed of inverters I1 and I2, turns off the respective device. For example, if the output node is high, then the output of I1 is low, which in turn pulls the gate of M3 down, thus shorting the base of Q5 to *VDD*. Similarly, if the output node is low, then the output of I1 is high, thus causing M4 to turn on and short the base of Q6 to VSS. Therefore, the output devices are saturated for only a short period of

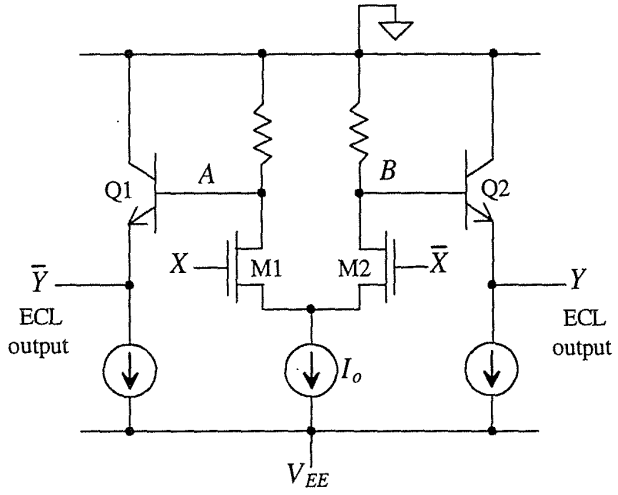



**Figure 16.19** CMOS to ECL conversion circuit.

Microchip Ex. 1055, p. 354
Microchip v. Aptiv
IPR2024-00494

time, while the feedback circuit preserves the logic value at the output. The resistor string is used to keep the base-emitter voltages of the output transistors on the edge of being turned on, thus improving speed even further.

Another conversion circuit is seen in Fig. 16.19 [10]. This circuit translates CMOS signals to ECL logic levels and requires a complemented CMOS input. The input signal causes an imbalance in the source coupled pair, since the current, $I_o$, is constant. The output swing of the source coupled pair appearing at nodes $A$ and $B$ can be adjusted by wisely choosing the resistance values and input MOS device sizes.

## REFERENCES

[1] A. R. Alvarez, "BiCMOS-Has the Promise Been Fulfilled?," *IEDM 1991*, pp.13.1.1-13.1.4, 1991.

[2] M. Kubo, I. Masuda, K. Miyata, and K. Ogiue, "Perspective on BiCMOS VLSI's," *IEEE Journal of Solid State Circuits*, vol. 23, no. 1, pp. 5-11, February 1988.

[3] M. I. Elmasry, "Introduction to BiCMOS Integrated Circuits: A Tutorial," *IEEE BiCMOS Integrated Circuit Design*, IEEE Press, 1994. ISBN 0-7803-0430-6.

[4] "Exponential Unveils World's Fastest PC Microprocessor at 533 MHz," Press Release by Exponential Technology, Inc., October 21, 1996.

[5] "Exponential's BiCMOS Technology: Bipolar-Based BiCMOS instead of CMOS-Based BiCMOS," http://www.exp.com/products/x704/bicmos.html, Exponential Technology, Inc., November 1996.

[6] M. I. Elmasry, *BiCMOS Integrated Circuit Design,* IEEE Press, 1992. ISBN 0-7803-0430-6, IEEE order number: PC0346-7.

[7] J. P. Uyemura, *Circuit Design for Digital CMOS VLSI,* Kluwer Academic Publishers, 1992.

[8] S. H. K. Embabi, A. Bellaouar, and M. I. Elmasry, "Analysis and Optimization of BiCMOS Digital Circuit Structures," *IEEE Journal of Solid State Circuits*, vol. 26, no. 4, pp. 676-679, April 1991.

[9] M. Rau and H. J. Pfleiderer, "An ECL to CMOS Level Converter with Complementary Bipolar Output Stage," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 7, pp. 781-787, July 1995.

[10] K. Gopalan, *Introduction to Digital Microelectronic Circuits*, Irwin, 1996. ISBN 0-256-12089-7.

## PROBLEMS

**16.1** Lay out and DRC a lateral 5 x 1 NPN BJT.

**16.2** Repeat Ex. 16.1 using a 5 x 1 BJT.

Microchip Ex. 1055, p. 355
Microchip v. Aptiv
IPR2024-00494

**16.3** Verify, using SPICE, the operation of the inverter given in Ex. 16.2.

**16.4** Design a full-swing BiCMOS output buffer that has an input capacitance of 100 fF or less and will drive 10 pF with a $t_{PHL} + t_{PLH}$ less than 15 ns.

**16.5** Design and simulate the operation of an ECL to CMOS converter based on the circuit topology shown in Fig. P16.5. Assume that the ECL input varies from 4.2 V (a logic high) down to 3.4 V (a logic low).

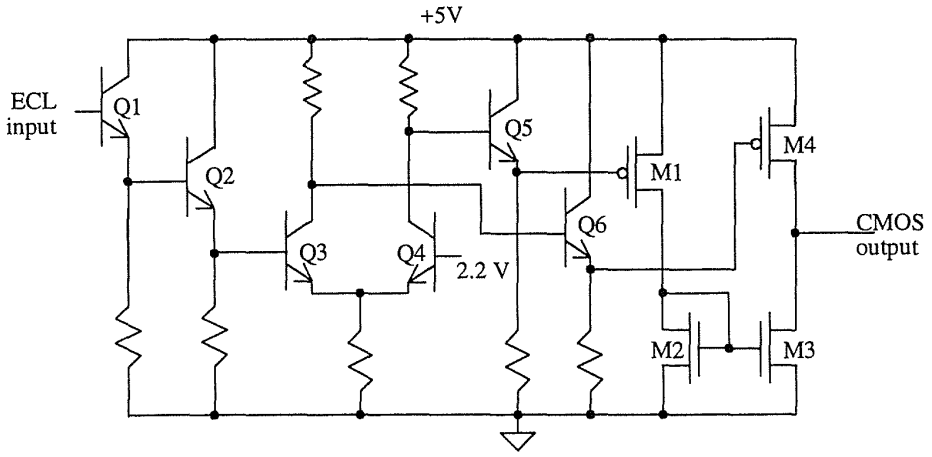



**Figure P16.5**

Microchip Ex. 1055, p. 356
Microchip v. Aptiv
IPR2024-00494

Chapter

17

# Memory Circuits

In this chapter we look into the design of semiconductor memory circuits [1, 2], in particular static random access memory (SRAM) and dynamic random access memory (DRAM). These types of memories are termed random access because any bit of data can be accessed at any time. A block diagram of a RAM is shown in Fig. 17.1.

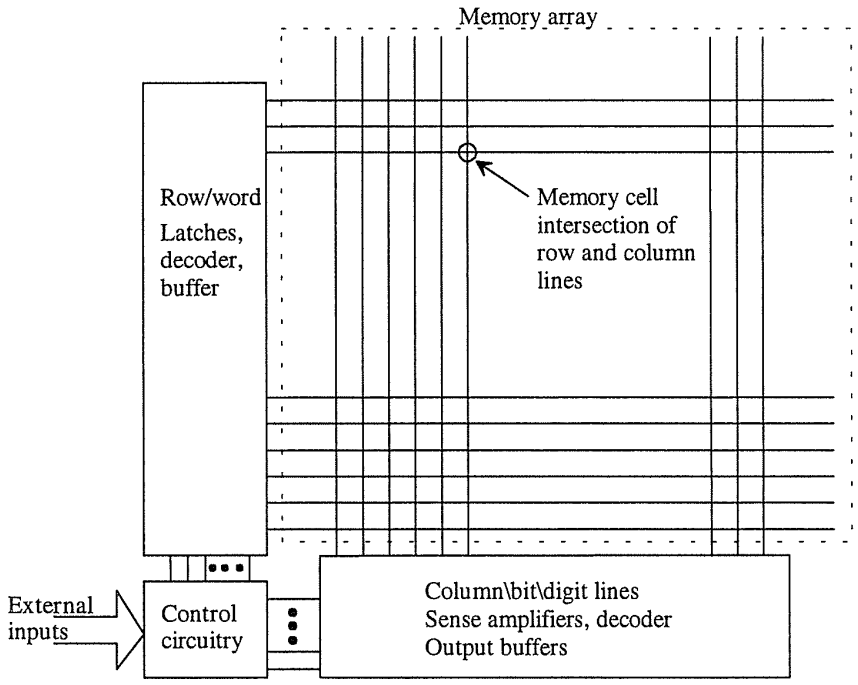



**Figure 17.1** Block diagram of random access memory.

Microchip Ex. 1055, p. 357
Microchip v. Aptiv
IPR2024-00494

External to the memory array are the row and column logic. The row lines are sometimes called word lines, while the column lines are sometimes called bit or digit lines. Referring to the row lines, the row address is latched, decoded, and then buffered. A particular row line will, when selected, go high. This selects the entire row of the array. Since the row line may be long and loaded periodically with the capacitive memory cells, a buffer is needed to drive the line. The address is latched with signals from the control logic. After a particular row line is selected, the column address is used to decode which of the bits from the row are the addressed information. At this point, data can be read into or out of the array through the column decoder. The majority of this chapter will concentrate on the circuits used to implement a RAM.

## 17.1 RAM Memory Cells

The memory array is at the center of the RAM design. Examples of DRAM and SRAM memory cells are shown in Fig. 17.2. The DRAM memory cell is made up of a pass transistor and a storage capacitor, while the CMOS SRAM memory cell is a cross-coupled connection of inverters. The cross-coupled inverters form a positive feedback circuit, forcing the outputs in opposite directions. The basic operation of the DRAM memory cell was discussed in Ch. 14.

For the SRAM cell shown in Fig. 17.2, consider the case when the row line (word line) is low. Both pass transistors are off, and the datum in the cell is latched, as long as power is applied to the cell. When the row line goes high, the pass transistors turn on. If the pass transistors, width/length ratio is approximately four times that of the SRAM cell transistors' width/length ratio, the data on the bit lines (column lines) are written to the cell. If the widths are comparable in size, the effective resistance of the pass transistor is too large to allow overwriting the cell.

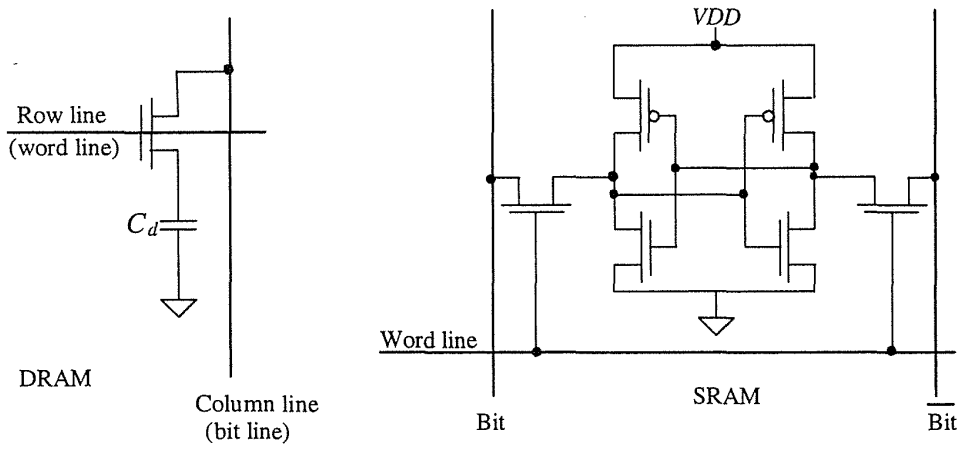



**Figure 17.2** Schematic diagrams of RAM memory cells.

Microchip Ex. 1055, p. 358
Microchip v. Aptiv
IPR2024-00494

### 17.1.1 The DRAM Cell

Consider the implementation of the DRAM cell shown in Fig. 17.3a. The capacitor is implemented using a MOSFET operating in the inversion region. Schematically, this would be represented by the capacitor in Fig. 17.2 connected to *VDD* instead of ground. The size of the DRAM storage capacitor $C_d$, is simply the sum of the oxide capacitance of the MOSFET used as a capacitor and the depletion capacitance of the n+ implant to substrate. The row/word line and *VDD* connections to the cell are made using poly1. This greatly simplifies layout.

In order to minimize the layout area of the DRAM array, the scheme shown in Fig. 17.3b is used. In this configuration, the bit line is shared between two cells. When writing to cell0, the word0 line goes high and the datum on the bit line is written to the capacitor M01. This configuration is termed a folded bit line configuration for reasons that are not obvious at this point.

Modern memory processes do not use a MOSFET as the storage capacitor but rather fabricate the storage capacitor using several layers of poly. For the sake of understanding memory circuit design, we will use the CN20 process, for the moment, to describe the implementation of the DRAM.

Consider the layout of the DRAM cell shown in Fig. 17.4. Since the word and *VDD* lines are implemented in poly, the connection to the MOSFETs occurs when the poly runs over the active n+ area. The bit line is implemented in metal1. The DRAM cells are placed adjacent to each other to form the array. A complete DRAM memory array is shown in Fig. 17.5. Notice how adjacent bit lines do not have a word line in common.

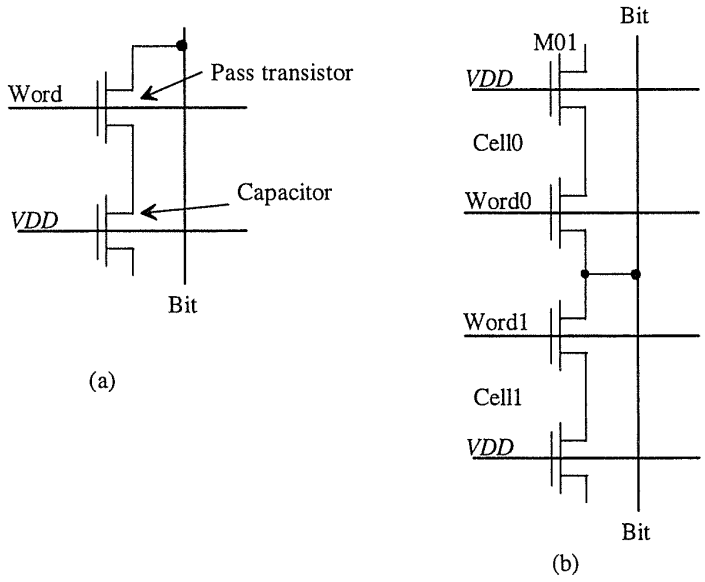



**Figure 17.3** Implementation of the DRAM cell.

Microchip Ex. 1055, p. 359
Microchip v. Aptiv
IPR2024-00494

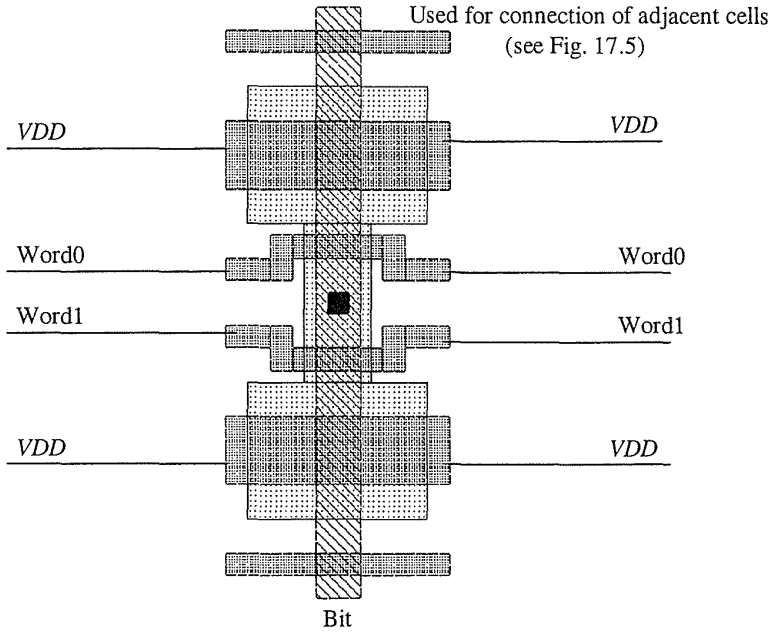



**Figure 17.4** Basic layout of a folded bit line DRAM cell.

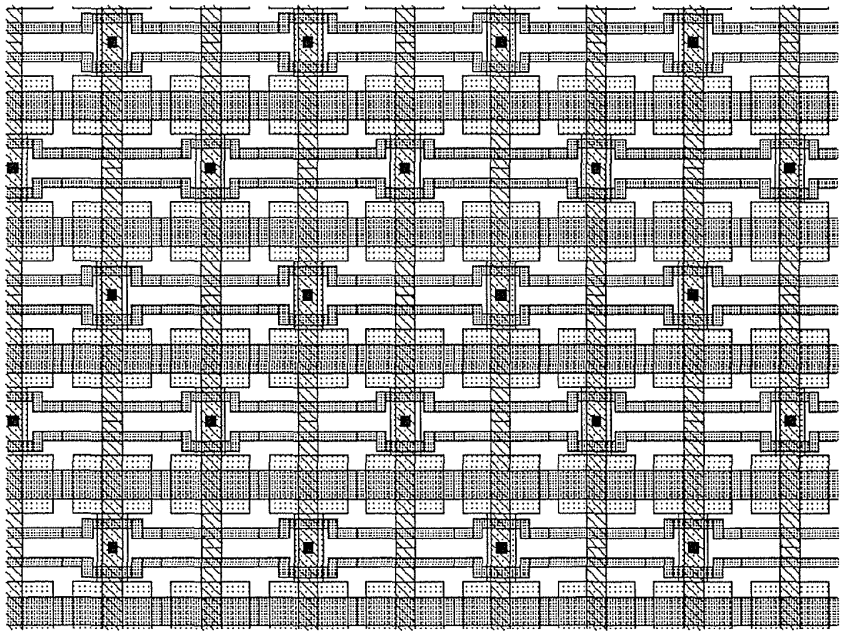


**Figure 17.5** Array of DRAM cells.

Microchip Ex. 1055, p. 360
Microchip v. Aptiv
IPR2024-00494

**Example 17.1**

This example illustrates the charge sharing principle. Consider the equivalent circuit of the basic DRAM cell shown in Fig. Ex17.1. The unwanted bit line capacitance, $C_{bit}$, is the sum of the metal1 to substrate capacitance of the bit line and the capacitance of the n+ implant depletion capacitance from each DRAM cell connected to the bit line. Assuming the initial voltage on the DRAM storage capacitance, $C_d$, is $V_d$ and the initial voltage on the bit line is $V_{bit}$,

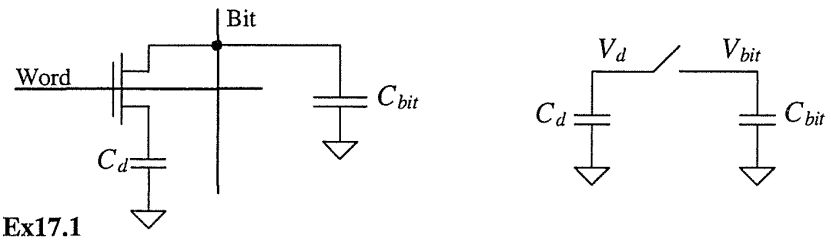



**Figure Ex17.1**

calculate the change in voltage on $C_{bit}$ when the word line is taken high.

The initial charge on $C_d$ is $V_d \cdot C_d$, and the initial charge on $C_{bit}$ is $V_{bit} \cdot C_{bit}$. When the word line goes high, the access MOSFET turns on, or in other words the switch closes, connecting the DRAM storage cell capacitance, $C_d$, directly to the bit line (and its associated capacitance). The voltage across each capacitor must be the same after the switch is closed. This voltage will be called $V_{final}$ and is given by

$$(C_d + C_{bit}) \cdot V_{final} = V_d \cdot C_d + V_{bit} \cdot C_{bit}$$

or

$$V_{final} = \frac{V_d \cdot C_d + V_{bit} \cdot C_{bit}}{C_d + C_{bit}} \tag{17.1}$$

The change in the bit line voltage is simply

$$\Delta V_{bit} = V_{final} - V_{bit} \tag{17.2}$$

Normally, the bit line is precharged to approximately 2.5 V before a read operation, and with typical values of $C_d$ and $C_{bit}$ of 75 fF and 500 fF, respectively, the change in bit line voltage with $V_d$ of 4 V is only 200 mV. The amplifier used to sense this small voltage change is called a sense amplifier or simply sense amp. ■

**Example 17.2**

Estimate the delay through a word line that is 2 mm long connected to MOSFET gates measuring 2 μm by 6 μm spaced every 20 μm.

Microchip Ex. 1055, p. 361
Microchip v. Aptiv
IPR2024-00494

The sheet resistance of poly1, from Appendix A, is 21 Ω/square, while the capacitance of poly1 to substrate is 58 aF/μm². The capacitance of each MOSFET gate connected to the word line is

$$C_{inn} = \frac{3}{2} \cdot 6 \cdot 2 \cdot 800 \text{ aF} = 14.4 \text{ fF}$$

If we break the word line into 20 μm increments that are 2 μm wide, the following circuit shown in Fig. Ex17.2 is the result.

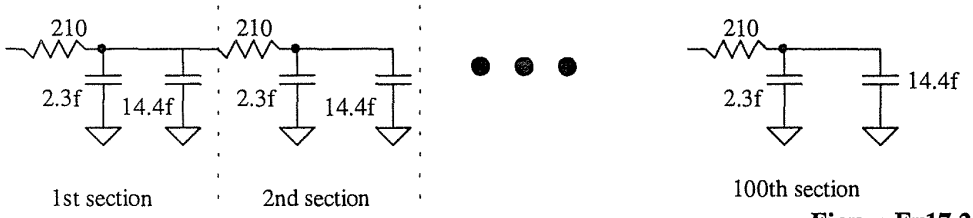



**Figure Ex17.2**

The delay through the entire line can be estimated by

$$t_d = 0.35rcN^2 = 0.35 \cdot 210 \cdot 16.7\text{f} \cdot (100)^2 = 12.5 \text{ ns}$$

As the density of the layout is increased, that is, the number of transistors connected to the word line increases over a given distance, this delay can become even larger. What is normally done in DRAM to circumvent this problem is to break the DRAM into smaller arrays multiplexing the data together. The smaller DRAM array supporting circuitry is easier to design for high-speed operation. ■

Often the substrate is "pumped" to a negative voltage, typically –2 V, in DRAM. This reduces the depletion capacitance between the n+ implant and the substrate, having the effect of lowering the bit line capacitance. Two other benefits of the negative substrate bias are that it increases latch-up immunity and allows the inputs on the chips to go negative without forward biasing the n+/substrate diode. A negative substrate potential has the unwanted effect of increasing the threshold voltage of the access transistor and increasing the junction leakage current. The threshold voltage drop of the access transistor when writing a high level to the storage cell (see Ch. 14) can be compensated for by using a voltage of $VDD + 2$ for the word line drivers. This puts the gate potential of the access transistors above $VDD + V_{THN}$.

Modern DRAM processes do not use a MOSFET in the inversion region for the storage capacitor but rather use a stacked or trenched capacitor (more on this later in the chapter). In the design of these storage capacitors, it is desirable to keep the oxide thickness as thin as possible in order to have the largest capacitance/area. The main limitation, from a circuit designer's point of view, on the oxide thickness is the breakdown voltage. The maximum electric field that should be applied to $SiO_2$ for reliable operation is 7 MV/cm, or 0.7 V/10 Å (see Ch. 6). For a 100 Å oxide thickness,

Microchip Ex. 1055, p. 362
Microchip v. Aptiv
IPR2024-00494

the maximum potential across the oxide for reliable operation is then limited to 7 V. If the substrate is pumped to –3 V and *VDD* is 5 V, poor reliability is probable when the process $t_{ox}$ is 100 Å. To limit the voltage across the storage capacitor, one plate is held at *VDD*/2 while the other plate, that is, the plate connected to the access transistor, can swing from ground to *VDD*.

### 17.1.2 The SRAM Cell

The basic SRAM cell shown in Fig. 17.2 is not popular because of its large layout size. The n-well needed for the p-channel transistors is the main cause. The SRAM cell shown in Fig. 17.6 is very popular because of its small layout size. The resistors are made using n+ and p+ polysilicon and are therefore not available in the process described in Appendix A (CN20). The layout of the polysilicon resistor is shown in Fig. 17.7. The resistor can be thought of as a leaky bipolar transistor. Typical resistance

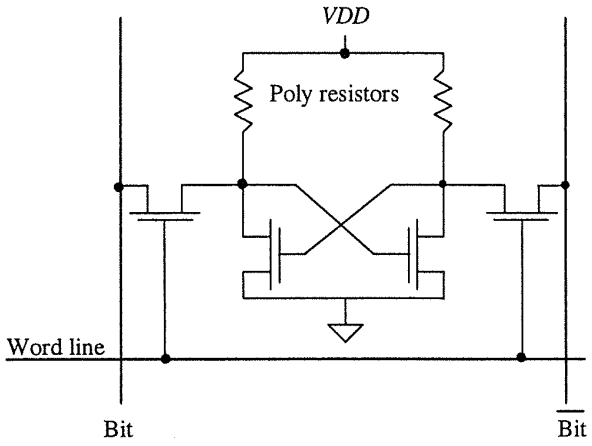



**Figure 17.6** SRAM memory cell with poly resistors.

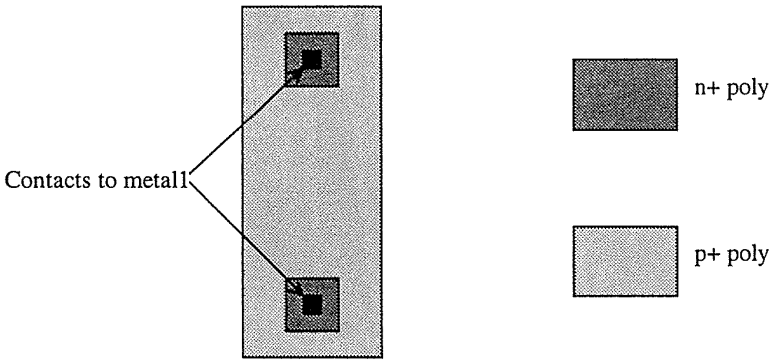



**Figure 17.7** Layout of the polysilicon resistor.

Microchip Ex. 1055, p. 363
Microchip v. Aptiv
IPR2024-00494

values are about 10 MΩ (or more). The CMOS SRAM cell dissipates essentially zero static power, while the power dissipated by the resistor/n-channel SRAM cell is $VDD^2/10$ MΩ, which is 2.5 μW for each cell when *VDD* is 5 V. An array of 1 million SRAM cells (128 k by 8) dissipates a power of 2.5 W when the resistor/n-channel SRAM cell is used.

## 17.2 The Sense Amplifier

Consider the portion of a DRAM memory array shown in Fig. 17.8. Two bit lines are shown in this figure. From Ex. 17.1 we found that the bit line may change very little when we are reading a cell. We use the sense amplifier to sense changes on the bit lines and determine whether a 1 or 0 was written to the cell.

Before we illustrate the operation of a sense amplifier, consider what happens to the bit lines when the word line, labeled wordA, in Fig. 17.8 is high. The charge, or absence of charge, on the storage capacitor is used to change the voltage of the bit line. The $\overline{\text{bit}}$ line potential remains unchanged and is used as a reference in the sense amplifier. The near proximity of the lines gives excellent noise rejection from coupled signals. Note that the line labeled bit in this figure corresponds to the bit line of the accessed transistor. If the wordB row line were high, the labeling of the bit lines in Fig. 17.8 would be switched. Also note that only one row line can go high, or in other words be selected, at a given time, while all even or odd bit lines intersecting a selected row line can change.

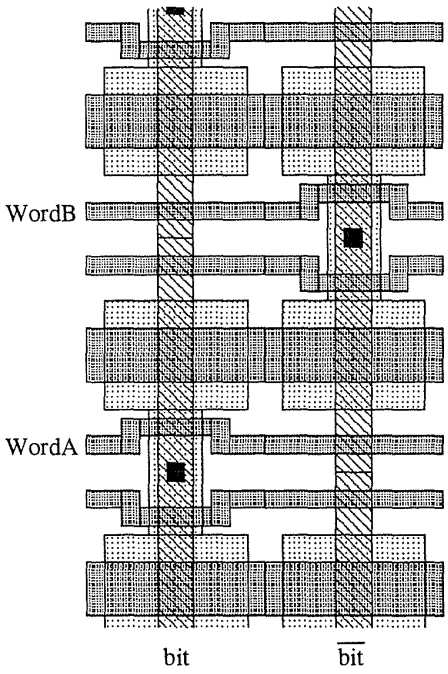



**Figure 17.8** Part of a DRAM array showing two bit lines.

Microchip Ex. 1055, p. 364
Microchip v. Aptiv
IPR2024-00494

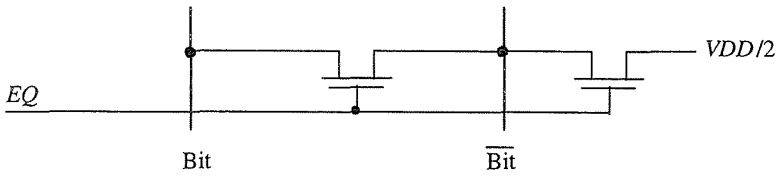



**Figure 17.9** Schematic of circuit used to precharge bit lines to *VDD*/2.

The first step in reading out a DRAM cell is to precharge all bit lines to, typically, *VDD*/2. The circuit used to precharge the lines is shown in Fig. 17.9. The control signal *EQ* (equilibrate) is used to turn the two MOSFETs on prior to a read operation, setting both bit lines to *VDD*/2. The MOSFET between the two bit lines is used to ensure that the potential of the lines is equal. This turns out to be very important since the sense amp must discriminate less than 100 mV voltage changes on the bit line.

The n- and p-sense amplifier schematics are shown in Fig. 17.10. The p-sense amplifier in this figure is placed on the top of the array, while the n-sense amplifier is located at the bottom of the array. During precharge of the bit lines, the signals *NLAT* (n-sense amplifier latch) and $\overline{ACT}$ (active pull up) are charged to *VDD*/2. The signals *NSA* and *PSA* are low and high, respectively.

After the digit (bit) lines have been charged to *VDD*/2, the row address decoder selects a word line. If the memory cell at the intersection of the word line and the bit line contains a low and the storage capacitor is discharged, the bit line voltage will fall from the precharged voltage. The $\overline{\text{bit}}$ line voltage remains unchanged. Evaluation of the bit line begins when *NSA* is driven high, causing *NLAT* to go low. If the bit line fell from *VDD*/2 (in other words, the accessed bit was a low), the n-sense amplifier would pull the bit line low. If the bit line increased in potential from *VDD*/2, the n-sense amplifier would pull the $\overline{\text{bit}}$ to a low. The n-sense amp can only pull a digit or column line low. The final stage in evaluating the digit lines is to fire the p-sense amp. This is done by pulling *PSA* low, causing $\overline{ACT}$ to go high. If the potential of the bit line, after the word line went high, is above *VDD*/2, the p-sense amplifier will pull the bit line high. If the bit line is below *VDD*/2, the p-sense amplifier will pull $\overline{\text{bit}}$ up to *VDD*. The p-sense amp can only pull a bit line high.

The sensitivity of the p-sense amplifier is far less critical than that of the n-sense amplifier. After the n-sense amplifier is fired, one of the bit lines is at ground potential while the other is at *VDD*/2. When the p-sense amplifier is fired, it sees this large potential difference. The n-sense amplifier must be capable of discriminating < 100 mV voltage differences, while the p-sense must discriminate *VDD*/2 voltage differences.

Another important attribute of the sense amplifier is its ability to restore or refresh the datum in the memory cell. If the word line stays high during the evaluate phase, the forcing of the bit line high or low by the sense amplifier also forces a high or

Microchip Ex. 1055, p. 365
Microchip v. Aptiv
IPR2024-00494

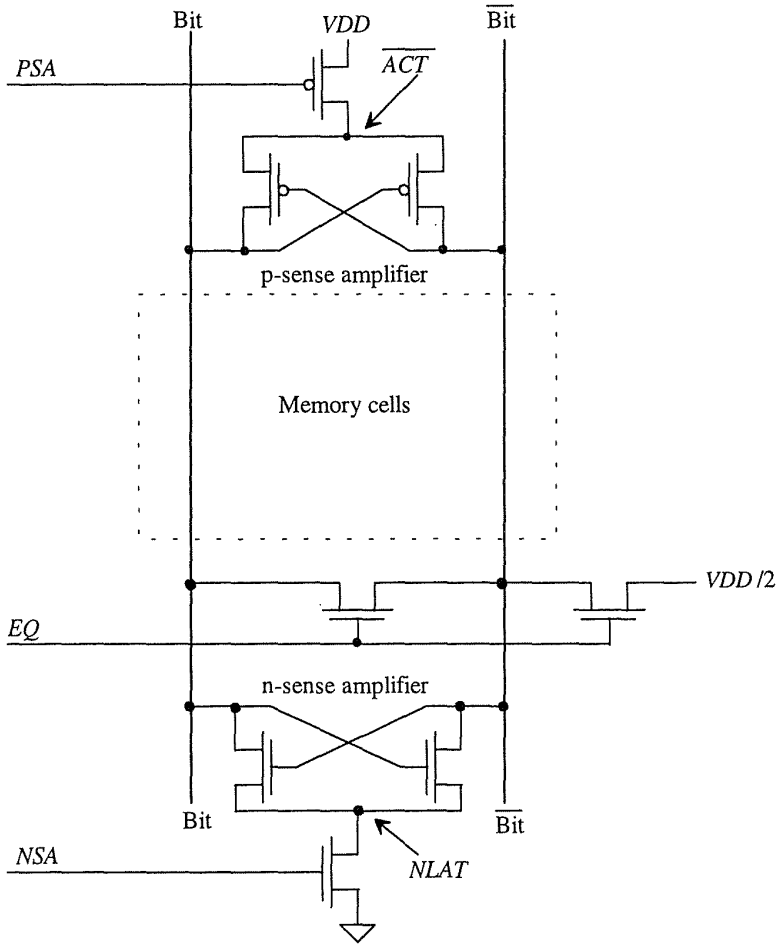



**Figure 17.10** Schematic of sense amplifiers.

low potential on the storage capacitor. The change in potential on the $\overline{bit}$ line has no effect on any of the memory cells since the word line does not access any data on this line. DRAM must be refreshed periodically to keep from losing data. Refreshing is accomplished by sequentially reading each row in the memory array since there is a sense amplifier for each pair of bit lines. The sense amplifier can be shared between different memory arrays on the chip. When the sense amp is shared, isolation transistors are used to select which array the sense amplifier will be used with. Isolation transistors also help to isolate the sense amp from the large bit line capacitance, keeping the sense amp from oscillating.

Microchip Ex. 1055, p. 366
Microchip v. Aptiv
IPR2024-00494

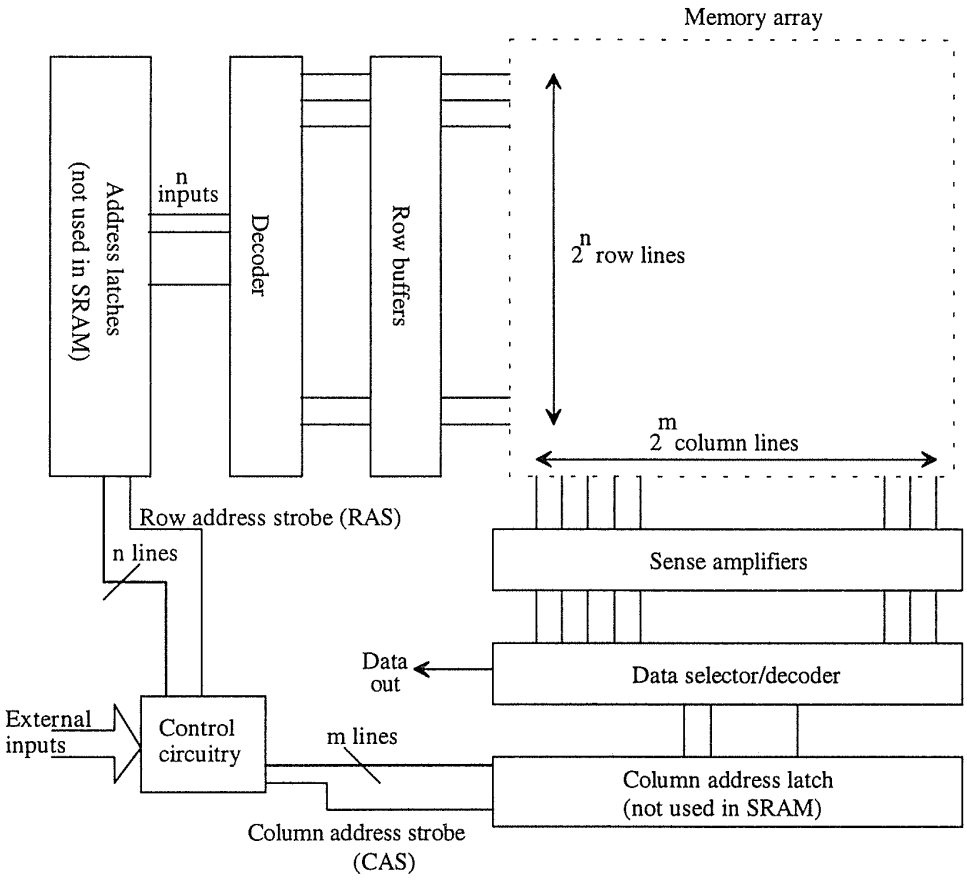



**Figure 17.11** Detailed block diagram of a RAM.

## 17.3 Row/Column Decoders

Consider the more detailed block diagram of a RAM shown in Fig. 17.11. If there are $2^n$ row lines and $2^m$ column lines (where a column line is bit and $\overline{\text{bit}}$), the total memory size is $2^{n+m}$ bits of data. In order to minimize the length of the row and column lines, it is desirable to make $n = m$, or in other words, a square array.

Selecting a bit of data begins by latching and decoding the row and column addresses. The row line that goes high as a result of the row decode is buffered to drive the word line across the array (see Ex. 17.2). The column decoder output is fed to a pass transistor. The pass transistor acts like a data selector, allowing the selected bit to be accessed for a read or write operation.

A common row decoder configuration, sometimes called a tree decoder, used in RAM is shown in Fig. 17.12a. The three-bit address A0, A1, and A2 is used to enable the pass transistors used in the decoder. One of the eight outputs is pulled high, that is,

Microchip Ex. 1055, p. 367
Microchip v. Aptiv
IPR2024-00494

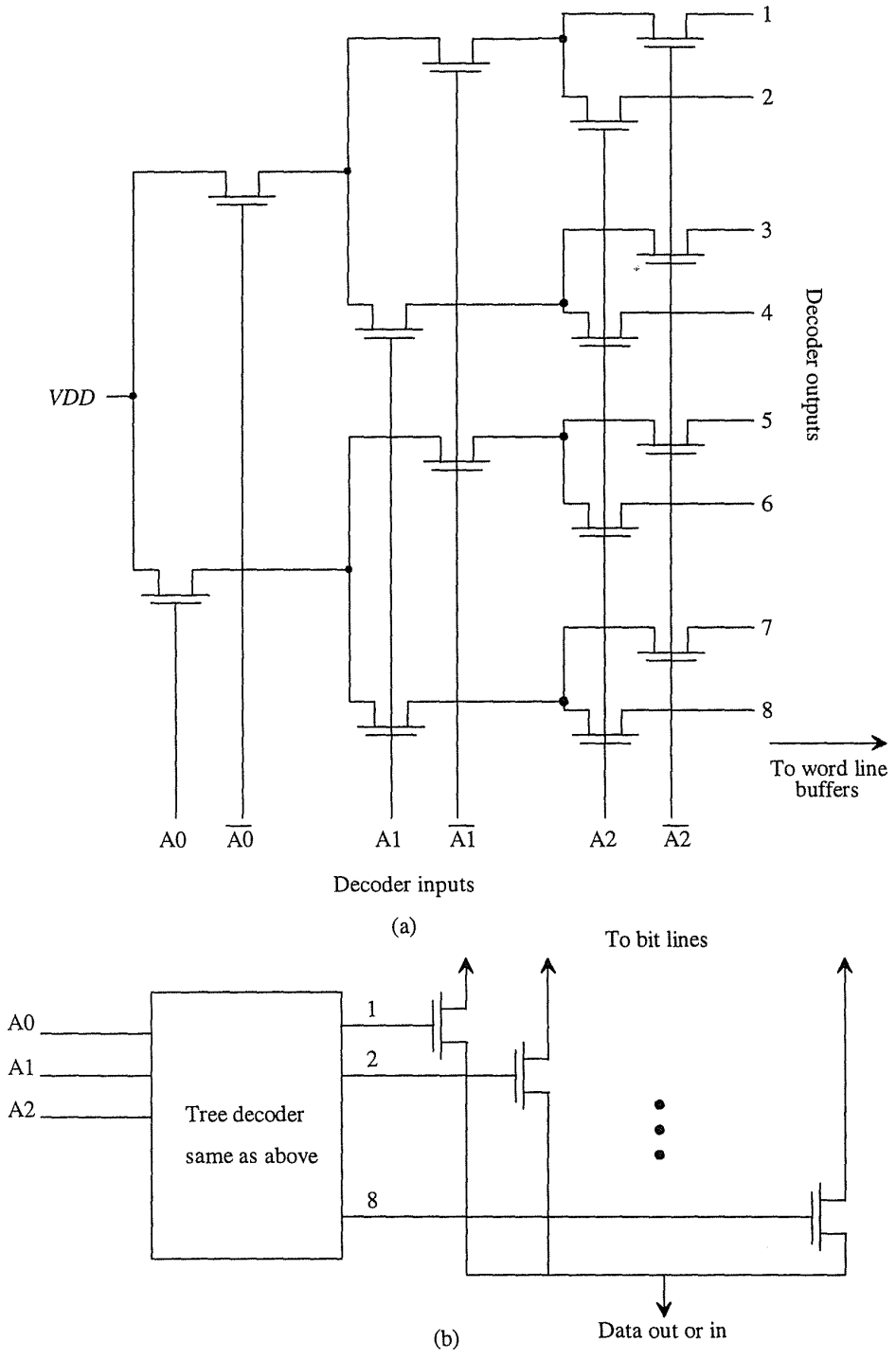



**Figure 17.12** Row (a) and column (b) decoders.

Microchip Ex. 1055, p. 368
Microchip v. Aptiv
IPR2024-00494

$VDD - V_{THN}$, depending on the three-bit address. Placing another pass transistor in series with *VDD* in this figure allows the decoder to be disabled.

The column address decoder, Fig. 17.12b is similar to the row address decoder with the addition of a pass transistor. The pass transistor is connected to the bit lines. The tree decoder portion of the column decoder is sometimes called a pre-decode. An enable can be used here as well.

Delay through the decoder is an important factor in large DRAM arrays. Consider a ten-bit row address. A total of ten tiers of pass transistors are used with each transistor connected to two others. Since the high output is well below *VDD* due to the body effect, a buffer circuit must regenerate the logic levels.

Figure 17.13 shows how the output of the decoder is pulled low when it is not selected. A long *L* MOSFET is used to pull the output of the decoder low when that particular output is not selected. The result is that all decoder outputs are zero except for the output that is selected by the input address. Two inverters are then used to drive the word line capacitance. The switching point voltage of the first inverter is set to compensate for the threshold voltage drop through the pass transistors.

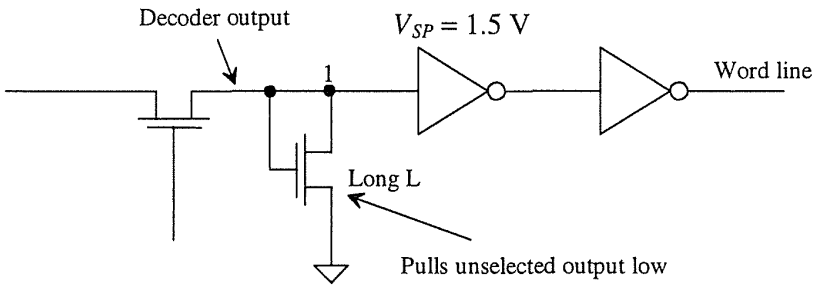



**Figure 17.13** Output buffer used in row decoder.

## 17.4 Timing Requirements for DRAMs

The timing signals used in a normal DRAM read are shown in Fig. 17.14. With the help of Fig. 17.11 these signals will be discussed. The first signal shown, $\overline{RAS}$, is the row address strobe and is active low. The row and column addresses are normally multiplexed together on the die so that one set of address inputs can be used. This reduces the number of pins in the packaged IC. Assuming the row address is set up on the address lines of the DRAM, the falling edge of $\overline{RAS}$ strobes the row address into the row address latches. After the row address has been latched, the address inputs of the chip are changed to the column address. At the falling edge of the column address strobe, $\overline{CAS}$, the column address is strobed into the column address latches. A finite time later, the data are available on the output pins of the DRAM. All internal signals, for example, *EQ*, *NLAT*, *ACT*, are derived from $\overline{RAS}$ and $\overline{CAS}$. Not shown in this

Microchip Ex. 1055, p. 369
Microchip v. Aptiv
IPR2024-00494

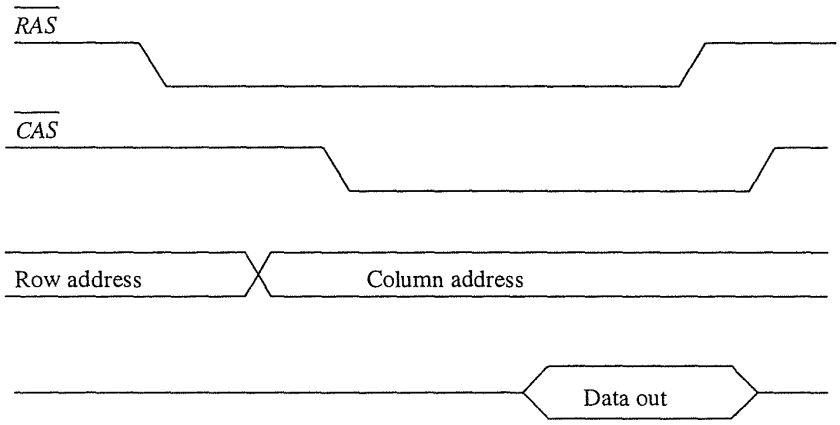



**Figure 17.14** Normal read cycle for a DRAM ($\overline{WE}$ = high).

timing diagram is the $\overline{WE}$ signal or write enable signal. This signal is used to put the DRAM in the read or write mode. Normal write operations timing waveforms are shown in Fig. 17.15.

Refreshing the DRAM is accomplished by sequentially addressing the word lines. When a word line is accessed, all bits on this line are refreshed by the sense amplifier as was discussed in Sec. 17.2. Figure 17.16 shows the timing signals for a $\overline{RAS}$ only refresh. The row addresses are sequentially applied to the DRAM during the refresh period.

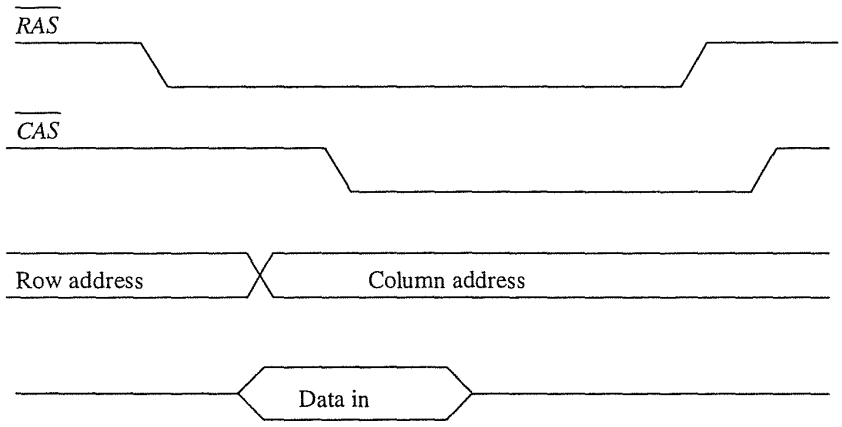



**Figure 17.15** Normal write cycle for a DRAM ($\overline{WE}$ = low).

Microchip Ex. 1055, p. 370
Microchip v. Aptiv
IPR2024-00494

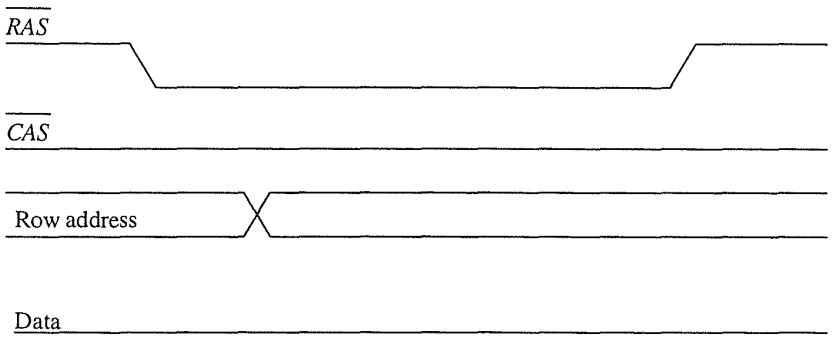



**Figure 17.16** $\overline{RAS}$ only refresh cycle.

## 17.5 Modern DRAM Circuits

The CN20 process is a general-purpose CMOS process. Memory manufactures, however, use CMOS processes specifically designed for memory circuit production. A memory process is specifically engineered to enhance yield. This section will discuss some aspects of modern CMOS memory design.

### 17.5.1 DRAM Memory Cell Layout

Modern DRAM memory cell layout, often called an mbit (memory bit), is much more involved than the memory cell layout given in Fig. 17.4. Figure 17.17 shows the layout of a modern DRAM memory cell. Again, a shared digit (bit) line contact is used to connect the access transistors to the digit line. The word lines are, again, made using poly1. An important aspect of the modern DRAM cell is the compactness of the mbit. A different set of design rules are used for the mbit layout.

The layout of the capacitor in a process specifically designed for memory is different as well. Figure 17.18 contains a process cross section for the buried capacitor mbit depicted in Fig. 17.17 (a vertical cross section). This type of mbit, employing a buried capacitor structure, places the digit line physically above the storage capacitor [4]. The digit line is constructed from either metal or polycide, while the digit line contact is formed using metal or polysilicon plug technology. The mbit capacitor is formed with polysilicon (poly2) as the bottom plate, an oxide-nitride-oxide (ONO) dielectric, and a sheet of polysilicon (poly3) which forms the common node shared by all mbit capacitors. The capacitor shape can be simple, such as a rectangle, or complex, such as concentric cylinders or stacked discs. Exotic capacitor structures are the topic of many DRAM process papers [5, 6, 7]. The ONO dielectric undergoes optimization to achieve maximum capacitance with minimum leakage. It must also tolerate the maximum DRAM operating voltage without breakdown. A cross-sectional view of an mbit using a trench capacitor is shown in Fig. 17.19 [6, 7].

Microchip Ex. 1055, p. 371
Microchip v. Aptiv
IPR2024-00494

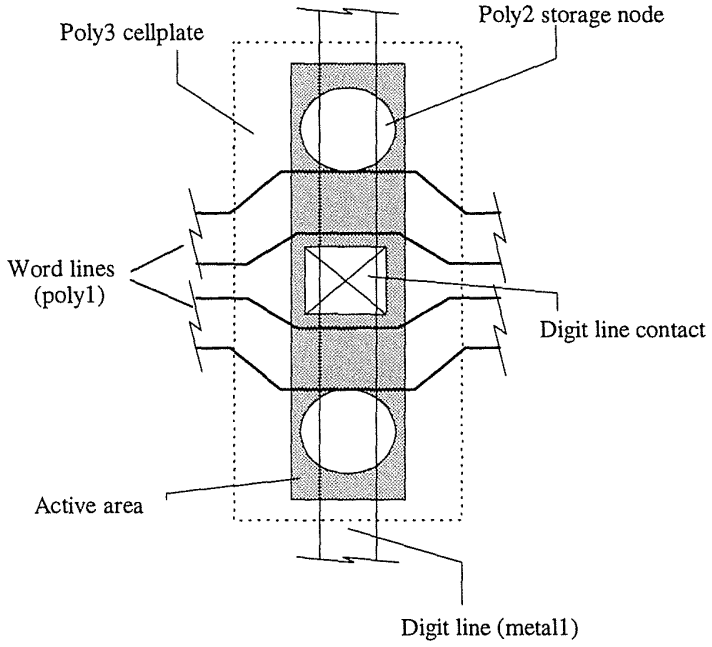



**Figure 17.17** Mbit pair layout.

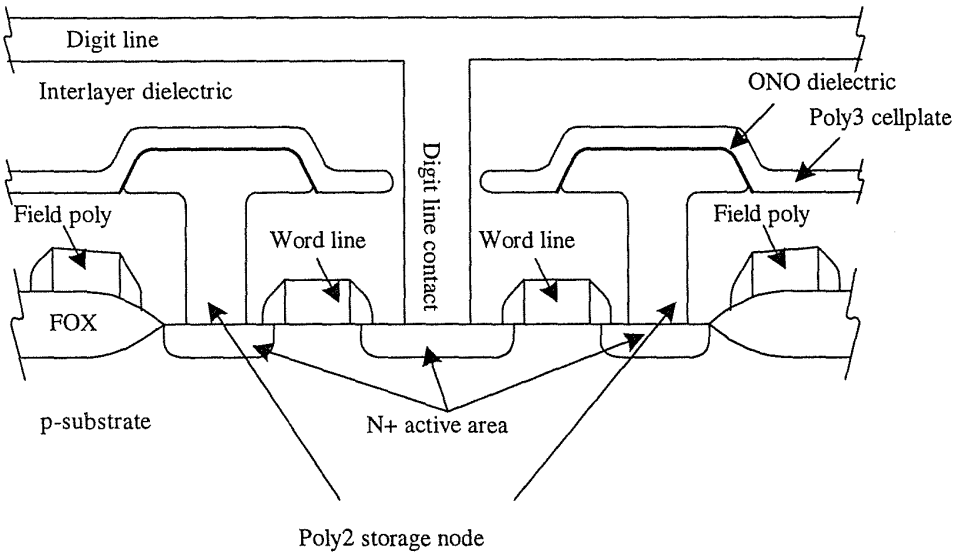



**Figure 17.18** Cross-sectional view of a buried capacitor cell, Fig. 17.17, cross section from top to bottom.

Microchip Ex. 1055, p. 372
Microchip v. Aptiv
IPR2024-00494

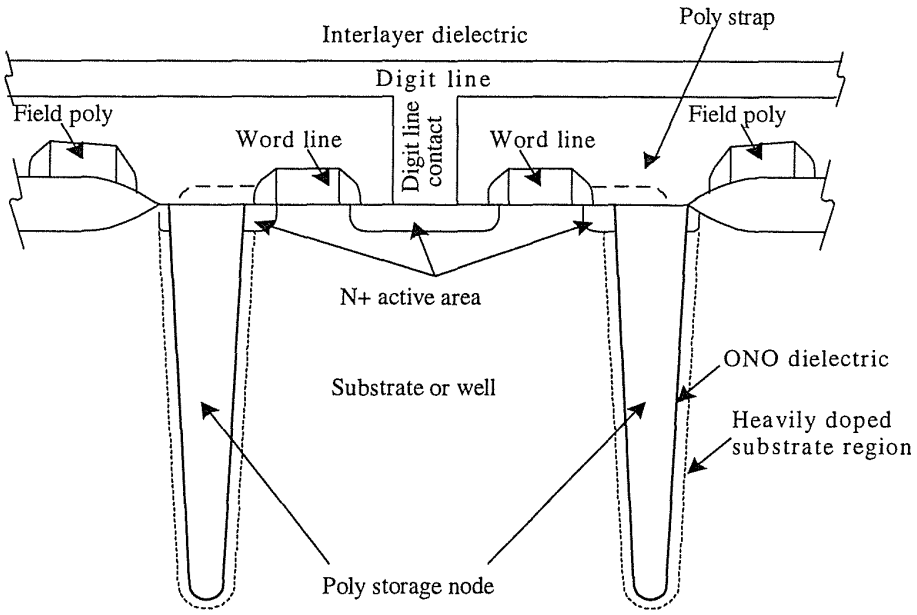



**Figure 17.19** Cross-sectional view of a trench capacitor cell.

Mbits organized in an array are shown in Fig. 17.20. A term used often in a periodic layout is *pitch*. The distance between like points in the array is the pitch length. In a memory array, the pitch is usually specified between adjacent bit lines as shown in Fig. 17.20. This figure is also useful to show how the size of a memory cell is specified. If we draw an imaginary box around an individual memory cell (see Fig. 17.20), the area of the box is the cell size. The feature size, $F$, is normally half the width of the bit line or word line pitch. The area of the memory cell of Figs. 17.17 and 17.20 is $8F^2$.

### 17.5.2 Folded/Open Architectures

The memory cells discussed thus far are used in the folded bit line architecture shown in Fig. 17.21. The p- and n-sense amplifiers are placed on the top and bottom of the memory array. First-generation DRAM, 64 Kbits and smaller, used the open bit line architecture shown in Fig. 17.22. The sense amplifier's inputs were from two separate memory arrays. The main benefit of the open bit line configuration is the resulting $6F^2$ memory cell (Fig. 17.23). The main drawback (and the reason it is no longer used) is the susceptibility to noise. Noise from the substrate and adjacent word and bit lines does not couple equally into the sense amplifier's inputs. The folded architecture essentially takes the arrays used in the open bit line configuration and "folds" them into a single array. The results are an array where the adjacent bit lines are not used at the same time for reading or writing and a larger array size for a given memory size. (This characteristic is, of course, unwanted.) The noise immunity is increased because of the close proximity. Substrate and coupled noise will ideally feed into each bit line equally.

Microchip Ex. 1055, p. 373
Microchip v. Aptiv
IPR2024-00494

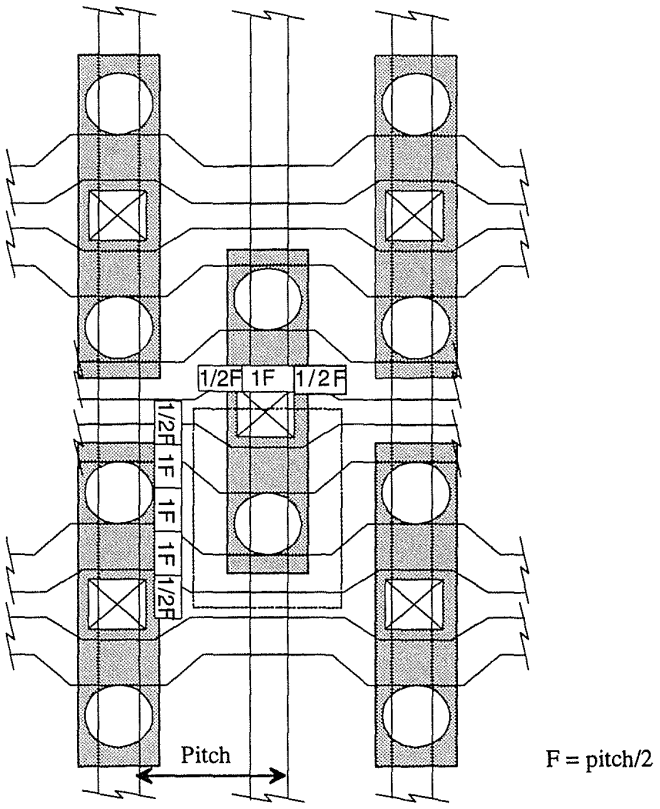



**Figure 17.20** How memory cell size is specified (not to scale).

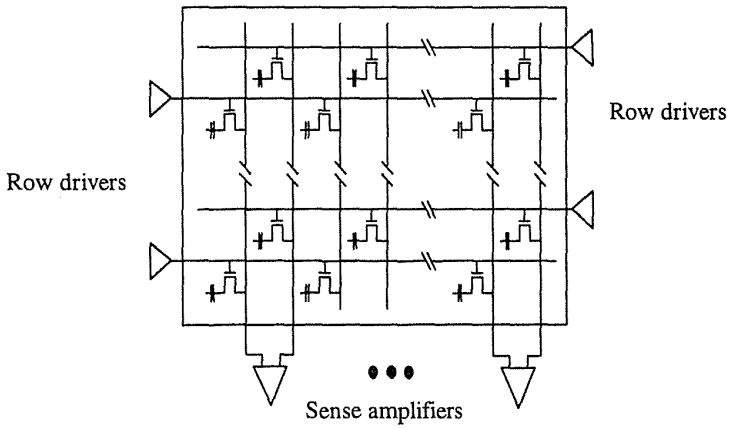



**Figure 17.21** Architecture for folded bit lines.

Microchip Ex. 1055, p. 374
Microchip v. Aptiv
IPR2024-00494

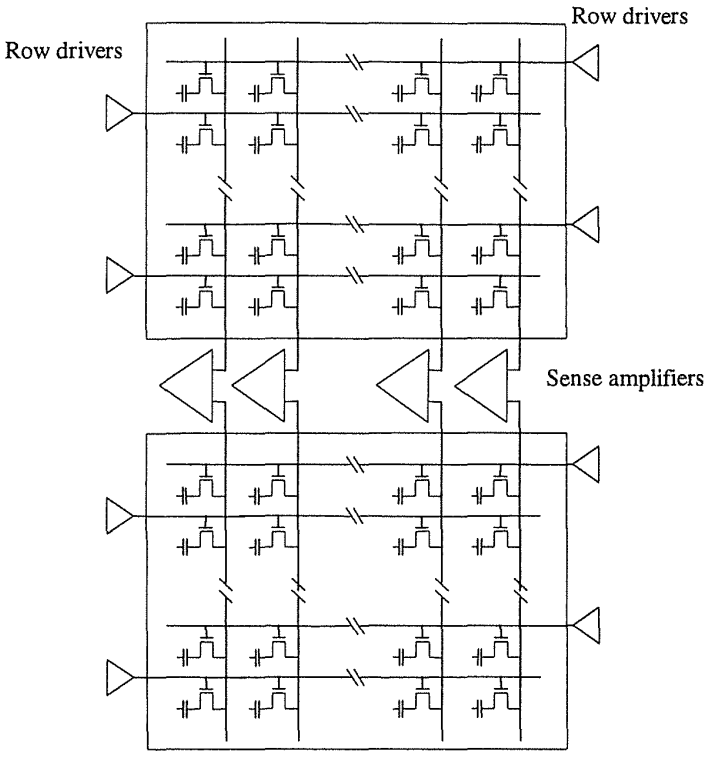



**Figure 17.22** Architecture used in an open bit line architecture.

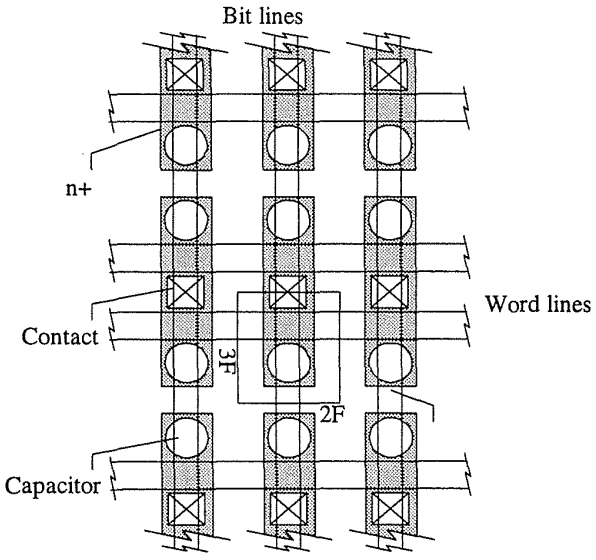



**Figure 17.23** Layout of mbit used in an open bit line configuration.

Microchip Ex. 1055, p. 375
Microchip v. Aptiv
IPR2024-00494

## 17.6 Other Memory Cells

In addition to the DRAM and SRAM memory cells discussed throughout this chapter, several other memory cells exist [1, 2, 8, 9]. These include read-only-memory (ROM), erasable programmable ROM (EPROM), electrically erasable programmable ROM (EEPROM), and Flash memory.

*Read-Only Memory (ROM)*

ROM is the simplest semiconductor memory. It is used mainly to store instructions or constants for use in a digital system. The basic operation of a ROM can be explained with the ROM memory shown in Fig. 17.24. Remembering that only one word line (row line) can be high at a time, we see that $R_1$ going high causes the column lines $C_1$, $C_2$, and $C_4$ to be pulled low. Column lines $C_3$ and $C_5$ are pulled high through the long L MOSFET loads at the top of the array. If the information that is to be stored in the ROM memory is not known prior to fabrication, the memory array is fabricated with an n-channel MOSFET at every intersection of a row and column line (Fig. 17.25a). The ROM is programmed (PROM) by cutting (or never making during fabrication) the connection between the drain of the MOSFET and the column line Fig. (17.25b). Because it is not easy to program ROM, it is limited to applications where it is mass produced.

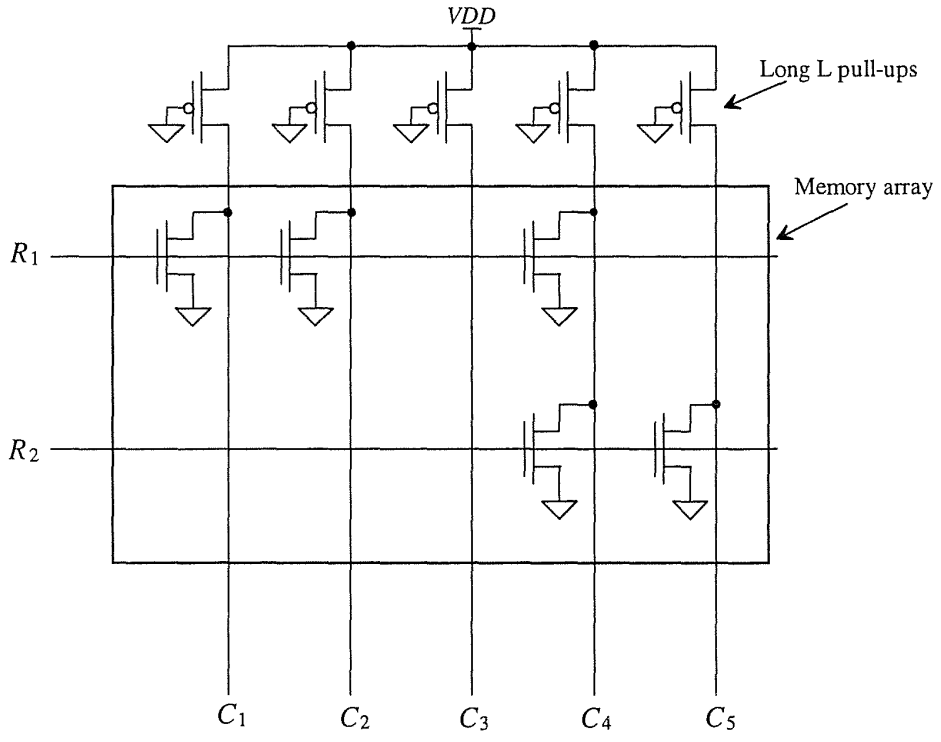



**Figure 17.24** A ROM memory array.

Microchip Ex. 1055, p. 376
Microchip v. Aptiv
IPR2024-00494

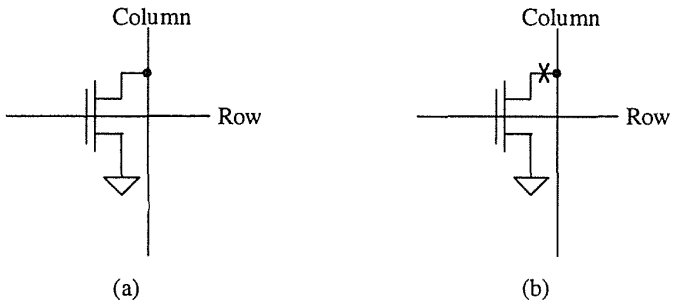



**Figure 17.25** (a) n-channel MOSFET at the intersection of every column and row line and (b) eliminating the connection between the drain and column line to program the ROM.

*Erasable Programmable Read-Only Memory (EPROM)*

EPROMs make programming the ROM significantly easier. Consider the cross-sectional view of an EPROM cell shown in Fig. 17.26. This modified n-channel MOSFET is used at the interesection of the column and row lines in the ROM memory array shown in Fig. 17.24. A second layer of polysilicon is added directly above the original polysilicon layer. The original poly layer is floating (i.e., not connected to anything). The second layer (poly2) is now used for connection to the row lines. Note that this is simply a poly2-poly1 capacitor where the bottom plate of the capacitor (poly1) is used in MOSFET formation.

To understand how to program the MOSFET, let's begin by assuming that both gates, poly1 (the bottom) and poly2, are at 0 V. A capacitance exists between poly2 and poly1 as well as between poly1 and the substrate. If the potential of poly2 is increased above 0 V, the voltage between these two capacitors ideally divides evenly since they should be approximately the same value. The result is an increase in the potential on poly1. If the potential on poly2 is raised to approximately $2 \cdot V_{THN}$, then the potential of the poly1 is raised to approximately $V_{THN}$. Increasing the row line voltage (the voltage

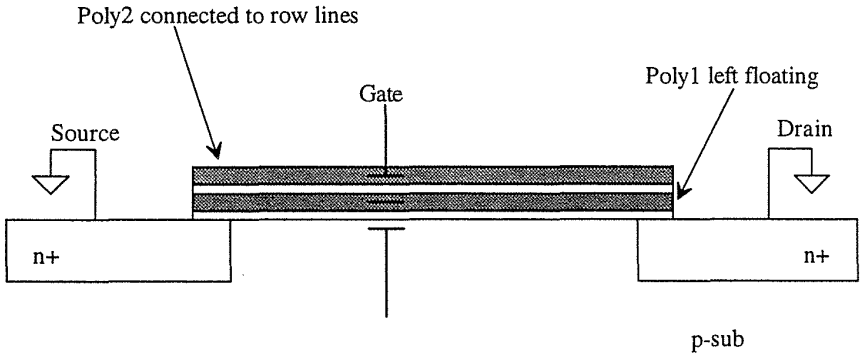



**Figure 17.26** EPROM memory cell.

Microchip Ex. 1055, p. 377
Microchip v. Aptiv
IPR2024-00494

on poly2) to 5 V turns the MOSFET on and pulls the column line low. In other words, the action of the row line going high is enough to turn the MOSFET on and pull the column line low.

To ensure that the MOSFET remains off when the row line goes high, a large voltage, around 25 V, is applied to the row lines to program the MOSFET to remain off during normal operation. The large voltage causes a large current to flow in the MOSFET. In addition, avalance multiplication occurs in the substrate, causing hot-carriers to become trapped on the poly1 [10]. When the large voltage is removed from poly2 (the word line), the voltage on poly1 will drop down to a negative voltage, typically – 5 V. This keeps the MOSFET from turning on under normal operation, allowing the column line to stay high. Since both gates are surrounded by $SiO_2$, a very good dielectric, the charge trapped on poly1 remains trapped for possibly several years. The cells can be reprogrammed by first illuminating the chip with UV light. This causes electron-hole generation in the $SiO_2$ , having the effect of increasing the insulator's conductivity. This in turn allows the charge trapped on the poly1 to leak off, until equilibrium is reached.

*Electrically Erasable Programmable Read-Only Memory (EEPROM)*

The inability to reprogram EPROM quickly has led to the use of EEPROM in many applications requiring nonvolatile memory. A voltage generator (see Ch. 18) is used on chip to generate the large voltage needed to program the EEPROM memory cell. The gate-oxide used in EEPROM is thinner than that used in EPROM. The result, when a large voltage, 10 V, is applied to poly2, is a conduction mechanism called Fowler-Nordheim tunneling, [11] between the substrate and poly1. This mechanism, unlike avalance breakdown, can conduct current in both directions between poly1 and substrate. A logic high in an EEPROM is programmed by raising the voltage on poly2 to 10 V, while a logic low is programmed by lowering the voltage to –10 V.

Flash memory is based on both EPROM and EEPROM technologies. Flash memories are programmed in a fashion similar to EPROM where hot electrons are used to accumulate charge on poly2 (see Fig. 17.26). Structurally, the Flash memory cell is the same as the EPROM cell except for the oxide thickness [9]. The oxide thickness is on the order of 100 Å, whereas the oxide thickness used in EPROM is 200-400 Å thick. Unlike EPROM, Flash memories can be erased in a fashion similar to EEPROMs. In other words, the Flash memory is programmed using hot electrons and erased using Fowler-Nordheim tunneling.

**REFERENCES**

[1] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits,* McGraw-Hill Publishing Co., 2nd ed., 1988. ISBN 0-07-029158-6.

[2] R. L. Geiger, P. E. Allen, and N. R. Strader, *VLSI - Design Techniques for Analog and Digital Circuits,* McGraw-Hill Publishing Co., 1990. ISBN 0-07-023253-9.

**Microchip Ex. 1055, p. 378**
**Microchip v. Aptiv**
**IPR2024-00494**

[3] B. Keeth, "A Novel Architecture for Advanced High Density Dynamic Random Access Memories," Master's Thesis, University of Idaho, May 1996.

[4] T. Hamada, "A Split-Level Diagonal Bit-Line (SLDB) Stacked Capacitor Cell for 256Mb DRAMs," *1992 IEDM Technical Digest*, pp. 799-802.

[5] T. Mohihara et al., "Disk-Shaped Capacitor Cell for 256Mb Dynamic Random-Acess Memory," *Japan Journal of Applied Physics*, Vol. 33, Part 1, No. 8, pp. 4570-4575, August 1994.

[6] J. H. Ahn et al., "Micro Villus Patterning (MVP) Technology for 256Mb DRAM Stack Cell," *1992 Symposium on VLSI Technical Digest of Technical Papers*, pp. 12-13.

[7] K. Sagara et al., "Recessed Memory Array Technology for a Double Cylindrical Stacked Capacitor Cell of 256M DRAM," *IEICE Trans. Electron.*, Vol. E75-C, No. 11, pp. 1313-1322, November 1992.

[8] M. I. Elmasry, *Digital MOS Integrated Circuits II,* IEEE Press, 1992. ISBN 0-87942-275-0, IEEE order number: PC0269-1.

[9] R. D. Pashley and S. K. Lai, "Flash Memories: The Best of Two Worlds," *IEEE Spectrum,* 1989, pp. 30-33.

[10] D. Frohman-Bentchkowsky, "FAMOS-A New Semiconductor Charge Storage Device," *Solid-State Electronics,* Vol. 17, pp. 517-529, 1974.

[11] E. H. Snow, "Fowler-Nordheim Tunneling in $SiO_2$ Films," *Solid-State Communications,* Vol. 5, pp. 813-815, 1967.

## PROBLEMS

**17.1** Estimate the capacitance that the pass transistor of Fig. 17.3 contributes to the word line in Fig. 17.3a. Assume that the MOSFET is minimum size, that is, 3 μm by 2 μm.

**17.2** Verify the results of Ex. 17.1 using SPICE.

**17.3** What layer in the CN20 process is used to make the word lines used in Ex. 17.2? Why can't we use metal1 with contacts to poly for the word lines? If we could, would this improve the delay through the word lines? Why?

**17.4** Sketch the cross-sectional view at the location shown for the layout shown in Fig. P17.4.

**17.5** Simulate the operation of the SRAM cell shown in Fig. 17.6. Assume that the poly n+/p+ resistors have a value of 10 MEGΩ.

**17.6** Explain why it is important to equilibrate the bit lines prior to reading the data from the memory cells. Refer to Fig. 17.10.

Microchip Ex. 1055, p. 379
Microchip v. Aptiv
IPR2024-00494

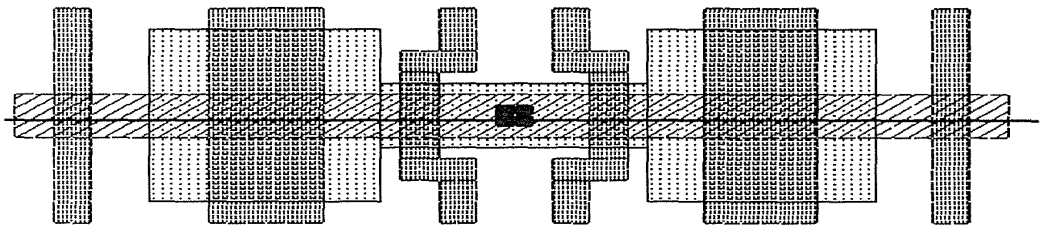


**Figure P17.4**

**17.7** Which DRAM architecture, that is, folded or open, results in the smallest memory bit (mbit) layout size? What is the major advantage of using the folded bit line architecture over the open?

**17.8** Suppose it is desirable to fabricate a ROM memory in a process used to make SRAMs. If the n+/p+ poly resistor is available, comment on the concerns that would arise if this resistor were used in place of the p-channel pull-up MOSFET shown in Fig. 17.24. Would there be any advantage to using the n+/p+ pull-up?

**17.9** Could we use the CN20 process to fabricate an EEPROM? If so, with the help of the poly2 design rules given in Appendix A, lay out a possible EEPROM cell.

Microchip Ex. 1055, p. 380
Microchip v. Aptiv
IPR2024-00494

Chapter

# 18

# Special-Purpose Digital Circuits

In this chapter we discuss some special-purpose integrated circuits. We begin with the Schmitt trigger, a circuit useful in generating clean pulses from a noisy input signal or in the designing oscillator circuits. Next, we discuss multivibrator circuits, both astable and monostable types. We end this chapter with a discussion of on-chip voltage generators.

## 18.1 The Schmitt Trigger

The schematic symbol of the Schmitt trigger is shown in Fig. 18.1 along with typical transfer curves [1]. We should note the similarity to the inverter transfer characteristics with the exception of a steeper transition region. Curve A in Fig. 18.1 corresponds to

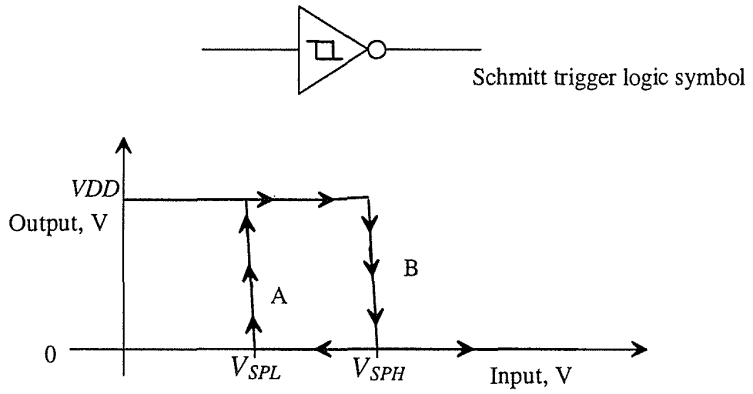



**Figure 18.1** Transfer characteristics of a Schmitt trigger.

Microchip Ex. 1055, p. 381
Microchip v. Aptiv
IPR2024-00494

the output of the Schmitt trigger changing from a low to a high, while curve B corresponds to the output changing from a high to a low. The hysteresis present in the transfer curves is what sets the Schmitt trigger apart from the basic inverter.

Figure 18.2 shows a possible input to a Schmitt trigger and the resulting output. When the output is high and the input exceeds $V_{SPH}$, the output switches low. However, the input voltage must go below $V_{SPL}$ before the output can switch high again. Note that we get normal inverter operation when $V_{SPH} = V_{SPL}$. The hysteresis of the Schmitt trigger is defined by

$$V_H = V_{SPH} - V_{SPL} \tag{18.1}$$

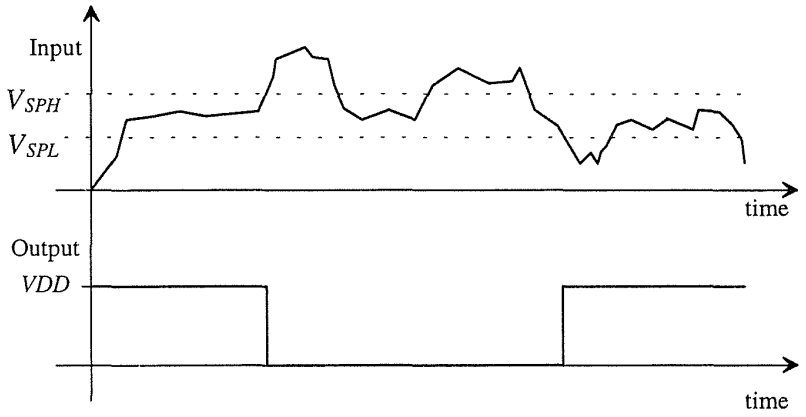



**Figure 18.2** Input, top trace, and output of a Schmitt trigger.

### 18.1.1 Design of the Schmitt Trigger

The basic schematic of the Schmitt trigger is shown in Fig. 18.3. We can divide the circuit into two parts, depending on whether the output is high or low. If the output is low, then M6 is on and M3 is off and we are concerned with the p-channel portion when calculating the switching point voltages, while if the output is high, M3 is on and M6 is off and we are concerned with the n-channel portion. Also, if the output is high, M4 and M5 are on, providing a DC path to *VDD*.

Let's begin our analysis of this circuit, assuming that the output is high (= *VDD*) and the input is low (= 0 V). Figure 18.4 shows the bottom portion of the Schmitt trigger used in calculating the upper switching point voltage, $V_{SPH}$. MOSFETs M1 and M2 are off, with $V_{in}$ = 0 V while M3 is on. The source of M3 floats to $VDD - V_{THN}$, or approximately 4 V for *VDD* = 5 V. We can label this potential $V_x$ as shown in the figure.

Microchip Ex. 1055, p. 382
Microchip v. Aptiv
IPR2024-00494

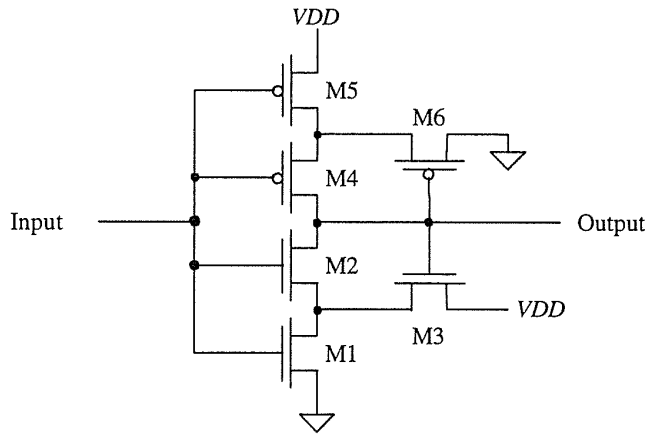



**Figure 18.3** Schematic of the Schmitt trigger.

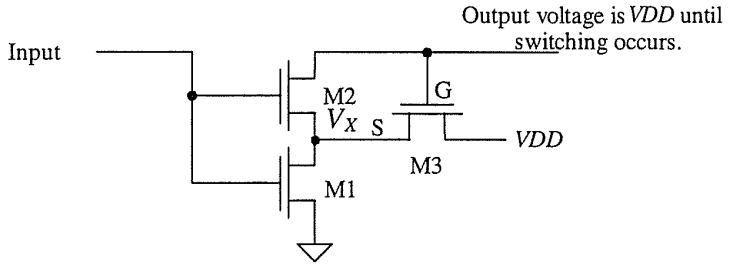



**Figure 18.4** Portion of the Schmitt trigger schematic used to calculate upper switching point voltage.

With $V_{in}$ less than the threshold voltage of M1, $V_X$ remains at approximately 4 V. As $V_{in}$ is increased further, M1 begins to turn on and the voltage, $V_X$, starts to fall toward ground. The high switching point voltage is defined when

$$V_{in} = V_{SPH} = V_{THN2} + V_X \tag{18.2}$$

or when M2 starts to turn on. As M2 starts to turn on, the output starts to move toward ground, causing M3 to start turning off. This in turn causes $V_X$ to fall further, turning M2 on even more. This continues until M3 is totally off and M2 and M1 are on. This positive feedback causes the switching point voltage to be very well defined.

When Eq. (18.2) is valid, the currents flowing in M1 and M3 are essentially the same. Equating these currents gives

$$\frac{\beta_1}{2}(V_{SPH} - V_{THN})^2 = \frac{\beta_3}{2}(VDD - V_X - V_{THN3})^2 \tag{18.3}$$

Since the sources of M2 and M3 are tied together, $V_{THN2} = V_{THN3}$ the increase in the threshold voltages from the body effect is the same for each MOSFET. The combination of Eqs. (18.2) and (18.3) yields

Microchip Ex. 1055, p. 383
Microchip v. Aptiv
IPR2024-00494

$$\frac{\beta_1}{\beta_3} = \left[\frac{VDD - V_{SPH}}{V_{SPH} - V_{THN}}\right]^2 \tag{18.4}$$

The threshold voltage of M1, given by $V_{THN}$ in this equation, is the zero body bias threshold voltage, or 0.83 V for a CN20 n-channel MOSFET. Given a specific upper switching point voltage, the ratio of the MOSFET transconductors is determined by solving this equation. A general design rule for selecting the size of M2, that is, $\beta_2$, is to require that

$$\beta_2 \geq 5\beta_1 \text{ and } 5\beta_3 \tag{18.5}$$

Since M2 is used as a switch, we require that it be larger than M1 or M3.

A similar analysis can be used to determine the lower switching point voltage $V_{SPL}$, resulting in the following design equation:

$$\frac{\beta_5}{\beta_6} = \left[\frac{V_{SPL}}{VDD - V_{SPL} - V_{THP}}\right]^2 \tag{18.6}$$

The following example illustrates the design procedure for a Schmitt trigger.

**Example 18.1**

Design and simulate a Schmitt trigger with $V_{SPL}$ = 2 V and $V_{SPH}$ = 3 V using the CN20 process.

We begin by solving Eqs. (18.4) and (18.6) for the transconductance ratios. For the upper switching point voltage,

$$\frac{\beta_1}{\beta_3} = \left[\frac{5-3}{3-0.83}\right]^2 = 0.85 = \frac{KP_N\frac{W_1}{L_1}}{KP_N\frac{W_3}{L_3}} = \frac{W_1 L_3}{W_3 L_1}$$

and for the lower switching point voltage,

$$\frac{\beta_5}{\beta_6} = \left[\frac{2}{5-2-0.91}\right]^2 = 0.92 = \frac{KP_P\frac{W_5}{L_5}}{KP_P\frac{W_6}{L_6}} = \frac{W_5 L_6}{W_6 L_5}$$

An infinite number of solutions exist for these equations. Keeping in mind that we are drawing our MOSFETs on a 1 μm grid[1], one possible set of sizes is

$$W_1 = L_1 = W_5 = L_5 = 3 \text{ μm}$$

$$W_3 = 7 \text{ μm and } L_3 = 6 \text{ μm}$$

$$W_6 = 12 \text{ μm and } L_6 = 11 \text{ μm}$$

[1] In practice, for the CN20 process, we can draw on a finer grid. A good alternative grid size for precision design in this process is 0.1 μm. Using the **Set** command in LASI, the number of grids can be set as well as the grid spacing.

Microchip Ex. 1055, p. 384
Microchip v. Aptiv
IPR2024-00494

Selection of the sizes of M2 and M4 follows the design rule given in Eq. (18.5). Noting that the *W/L* ratios of M1, M3, M5, and M6 are close to unity, we can require that $W_2/L_2$ and $W_4/L_4 = 5$. Setting $W_2 = W_4 = 10$ μm and $L_2 = L_4 = 2$ μm is a good choice for these device sizes. The simulation results are shown in Fig. 18.5. Two DC simulations were run on the circuit to show both an output high to low and an output low to high transition. ■

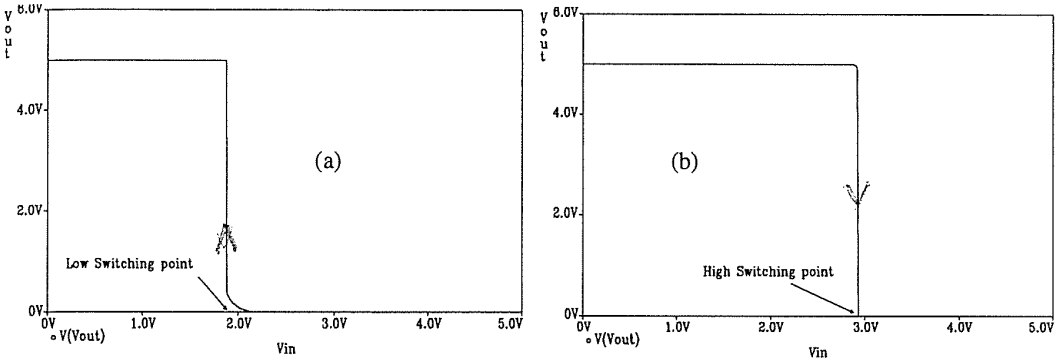



**Figure 18.5** Simulation results showing (a) sweeping the input from 5 V to 0 and (b) sweeping the input from 0 to 5 V (showing high switching point).

### 18.1.2 Switching Characteristics

The propagation delays of the Schmitt trigger can be calculated in much the same way as the inverter of Ch. 11. Defining equivalent digital resistances for M1, M2, M4, and M5 as $R_{n1}$, $R_{n2}$, $R_{p4}$, and $R_{p5}$, respectively, gives a high to low propagation delay-time, neglecting the Schmitt trigger output capacitance of

$$t_{PHL} = (R_{n1} + R_{n2}) \cdot C_{load} \tag{18.7}$$

and

$$t_{PLH} = (R_{p4} + R_{p5}) \cdot C_{load} \tag{18.8}$$

### 18.1.3 Applications of the Schmitt Trigger

Consider the waveform shown in Fig. 18.6. A pulse with ringing is a common voltage waveform encountered in busses or lines interconnecting systems. If this voltage is applied directly to a logic gate or inverter input with a $V_{SP}$ of 2.5 V, the output of the gate will vary with the period of the ringing on top of the pulse. Using a Schmitt trigger with properly designed switching points can eliminate this problem.

The Schmitt trigger can also be used as an oscillator (Fig. 18.7). The delay-time in charging and discharging the capacitor is used to set the oscillation frequency. At the moment in time when the output of the Schmitt trigger switches low, the voltage across the capacitor is $V_{SPH}$. The capacitor will start to discharge toward ground. The voltage across the capacitor is given by

$$V_c(t) = V_{SPH} \cdot e^{-t/RC} \tag{18.9}$$

Microchip Ex. 1055, p. 385
Microchip v. Aptiv
IPR2024-00494

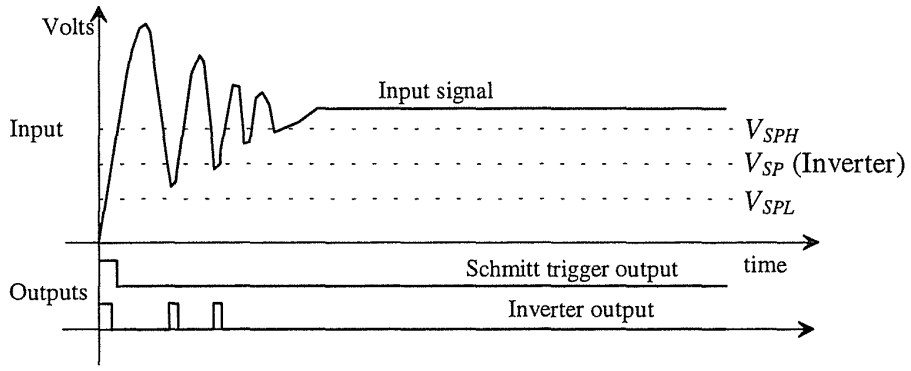



**Figure 18.6** Applying a Schmitt trigger to clean up an interconnecting signal.

At the time when $V_c(t) = V_{SPL}$, the output of the Schmitt trigger changes state. This time is given by solving Eq. (18.9) by

$$t_1 = RC \cdot \ln \frac{V_{SPH}}{V_{SPL}} \tag{18.10}$$

A similar analysis for the case when the capacitor is charged from $V_{SPL}$ to $V_{SPH}$ gives

$$V_c(t) = V_{SPL} + (VDD - V_{SPL})\left(1 - e^{\frac{-t}{RC}}\right) \tag{18.11}$$

and

$$t_2 = RC \cdot \ln \frac{VDD - V_{SPL}}{VDD - V_{SPH}} \tag{18.12}$$

The oscillation frequency, neglecting the intrinsic delay of the Schmitt trigger, is given by

$$f_{osc} = \frac{1}{t_1 + t_2} \tag{18.13}$$

The capacitance used in these equations is the sum of the input capacitance of the Schmitt trigger and any external capacitance.

An alternative oscillator using the Schmitt trigger is shown in Fig. 18.8 [2]. Here the MOSFETs M1 and M4 behave as current sources (see Ch. 20) mirroring the

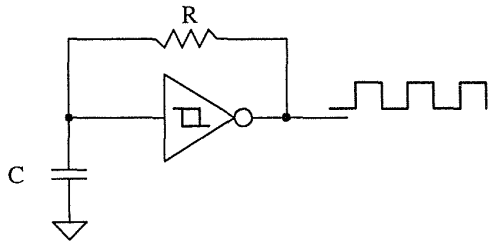



**Figure 18.7** Oscillator design using a Schmitt trigger.

Microchip Ex. 1055, p. 386
Microchip v. Aptiv
IPR2024-00494

current in M5 and M6. When the output of the oscillator is low, M3 is on and M2 is off. This allows the constant current from M4 to charge $C$. When the voltage across $C$ reaches $V_{SPH}$, the output of the Schmitt trigger swings low. This causes the output of the oscillator to go high and allows the constant current from M1 to discharge $C$. When $C$ is discharged down to $V_{SPL}$, the Schmitt trigger changes states. This series of events continues, generating the square wave output.

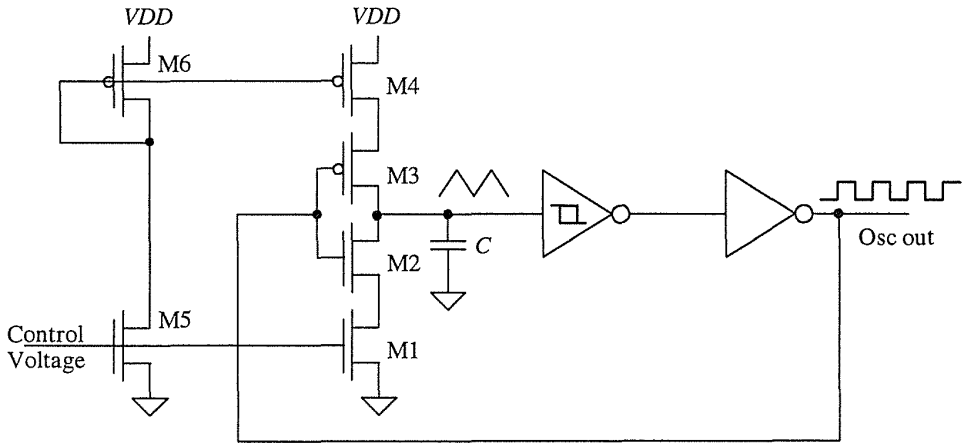



**Figure 18.8** Voltage-controlled-oscillator using Schmitt trigger and current sources. MOSFETs M2 and M3 are used as switches.

If we label the drain currents of M1 and M4 as $I_{D1}$ and $I_{D4}$, we can estimate the time it takes the capacitor to charge from $V_{SPL}$ to $V_{SPH}$ as

$$t_1 = C \cdot \frac{V_{SPH} - V_{SPL}}{I_{D4}} \tag{18.14}$$

and the time it takes to charge from $V_{SPH}$ to $V_{SPL}$ is

$$t_2 = C \cdot \frac{V_{SPH} - V_{SPL}}{I_{D1}} \tag{18.15}$$

The period of the oscillation frequency is, as before, the sum of $t_1$ and $t_2$.

This type of oscillator is termed a voltage-controlled oscillator (VCO) since the output frequency can be controlled by an external voltage. The currents $I_{D1}$ and $I_{D4}$, (Fig. 18.8) are directly controlled by the control voltage. As we will see in Ch. 20, the current in M5 is mirrored in M1, M4, and M6, with an appropriate scaling factor dependent on the size of the transistors.

### 18.1.4 High-Speed Schmitt Trigger

It turns out in practice that the Schmitt trigger of Fig. 18.3 is not easily optimized for high speed. The effective switching resistances of the MOSFETs are difficult to reduce

Microchip Ex. 1055, p. 387
Microchip v. Aptiv
IPR2024-00494

without changing the switching point voltages. An alternative design uses the basic inverter latch, or SRAM storage cell, shown in Fig. 18.9. Although the propagation times of this Schmitt trigger are less than the previous design, the switching point voltages are more difficult to define, which makes it less useful in some cases.

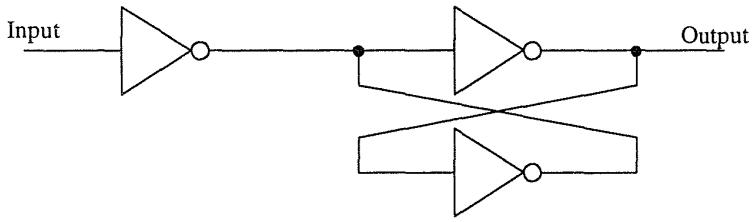



**Figure 18.9** Schmitt trigger using the basic latch.

## 18.2 Multivibrator Circuits

Multivibrator circuits (Fig. 18.10) are circuits that employ positive feedback. There are three types of multivibrators: astable, bistable, and monostable. Astable multivibrator circuits are unstable in either output (high or low) state. The oscillators we have discussed are examples of astable multivibrators. The bistable multivibrator is stable in either the high or low state. Flip-flops and latches are examples of the bistable multivibrator. Monostable multivibrators are stable in a single state. Monostable multivibrators are also called one-shots. In this, section we will discuss the monostable and astable multivibrators.

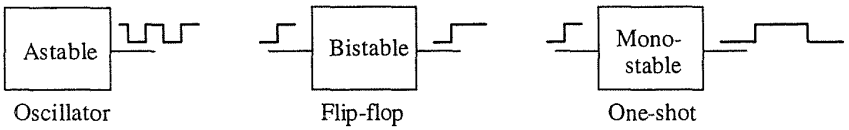



**Figure 18.10** Multivibrator circuits.

### 18.2.1 The Monostable Multivibrator

A CMOS implementation of the monostable multivibrator is shown in Fig. 18.11. Under normal conditions, $V_{in}$ is low and the output of the NOR gate, $V_1$, is high. The voltage $V_2$ is pulled high through the resistor, and the output of the inverter, $V_3$, is a low. Upon application of a trigger pulse, that is, $V_{in}$ going high, both $V_1$ and $V_2$ drop to zero volts and the output of the inverter, which is also the output of the monostable, goes high. This output is fed back to the input of the NOR gate holding $V_1$ at ground potential.

Microchip Ex. 1055, p. 388
Microchip v. Aptiv
IPR2024-00494

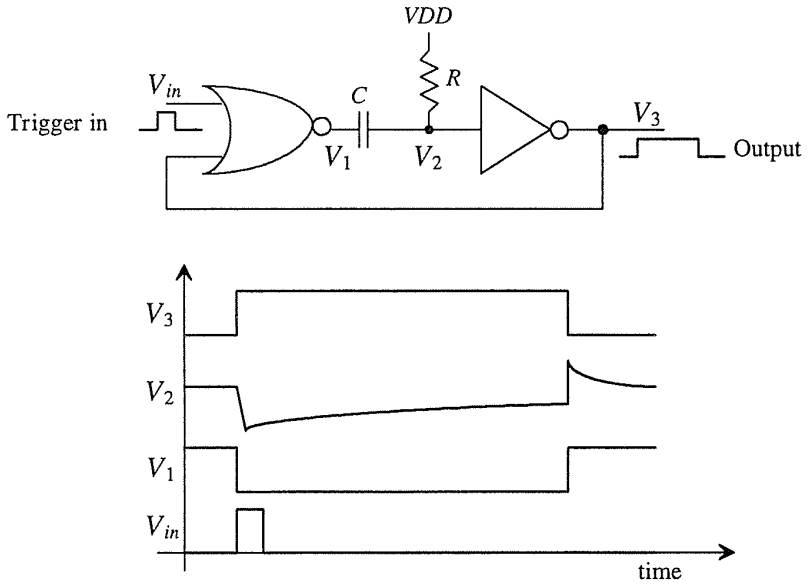



**Figure 18.11** Operation of the monostable multivibrator.

After triggering, the potential $V_2$ will start to increase because $C$ is charged through $R$. The potential across the capacitor after triggering takes place is given by

$$V_c(t) = V_2(t) - \overbrace{V_1(t)}^{=0} = VDD(1 - e^{\frac{-t}{RC}}) \tag{18.16}$$

If we assume that the $V_{SP}$ of the inverter is $VDD/2$, then the time it takes for the capacitor to charge to $V_{SP}$ is given by

$$t = RC \cdot \ln \frac{VDD}{VDD - V_{SP}} = RC \cdot \ln(2) \approx 0.7RC \tag{18.17}$$

This time also defines the output pulse width, neglecting gate delays, since the inverter switches low. The inverter output, $V_3$, going low causes $V_1$ to go back to $VDD$ and $V_2$ to go to $VDD + VDD/2$. If the resistor or capacitor is bonded out (connected to the output pads), the ESD diodes may keep $V_2$ from going much above $VDD + 0.7$. The time it takes $V_2$ to decay back down to $VDD$ limits the rate at which the one-shot can be retriggered. Also note that the trigger input can be longer than the output pulse width. Longer output pulse widths may cause $V_2$ to go as high as 10 V as well as limit the maximum trigger rate.

### 18.2.2 The Astable Multivibrator

An example of an astable multivibrator is shown in Fig. 18.12. This circuit has no stable state and thus oscillates. To analyze the behavior of this multivibrator, let's begin by assuming that the output, $V_3$, has just switched high. The output going high causes

Microchip Ex. 1055, p. 389
Microchip v. Aptiv
IPR2024-00494

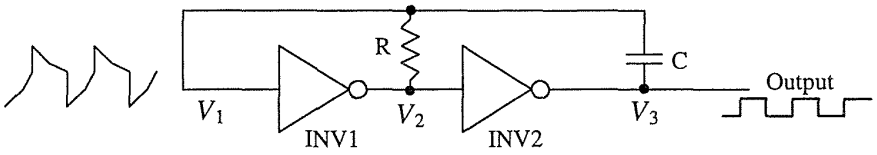



**Figure 18.12** An astable multivibrator.

$V_1$ to go high (actually $VDD + V_{SP1}$), forcing $V_2$ low. The voltage across the capacitor after this switching takes place is given by

$$V_c(t) = V_1(t) - \overbrace{V_3(t)}^{=VDD} = (VDD + V_{SP1}) \cdot e^{\frac{-t}{RC}} - VDD \qquad (18.18)$$

The output of the astable will go low, $V_3 = 0$, when $V_1 = V_{SP1}$. Substituting this condition into the previous equation gives the time the output is high (or low) and is given by

$$t_1 = RC \cdot \ln \frac{VDD + V_{SP1}}{V_{SP1}} = t_2 \qquad (18.19)$$

If $V_{SP1} = VDD/2$, then

$$t_1 = t_2 = 1.1RC \qquad (18.20)$$

and the frequency of oscillation is

$$f_{osc} = \frac{1}{t_1 + t_2} = \frac{1}{2.2RC} \qquad (18.21)$$

Again, if the resistor and capacitor are bonded out, the ESD diodes on the pads will limit the voltage swing of $V_1$.

## 18.3 Voltage Generators

Often when designing CMOS circuits, positive and negative DC voltages are needed that do not lie between *VSS* and *VDD*. A simple example of an application where a larger DC voltage source is used was given in Fig. 11.25. In Fig. 11.25 an inverter was powered with a positive voltage supply of *VDD* + 2V. A simple circuit, sometimes called a voltage pump, useful in generating a voltage greater than *VDD*, is given in Fig. 18.13.

The operation of the voltage pump of Fig. 18.13 can be explained by first realizing that both M1 and M2 operate like a diode. M1 is simply used to pull point A to a voltage of $VDD - V_{THN}$. (The threshold voltage with body effect using the CN20 process with a $V_{SB}$ of 5 V is approximately 1.5 V.) Let's begin the description of the circuit operation by assuming that the output of the inverter is low and point A is at a potential of $VDD - V_{THN}$. When the output of the inverter goes high, the potential at

Microchip Ex. 1055, p. 390
Microchip v. Aptiv
IPR2024-00494

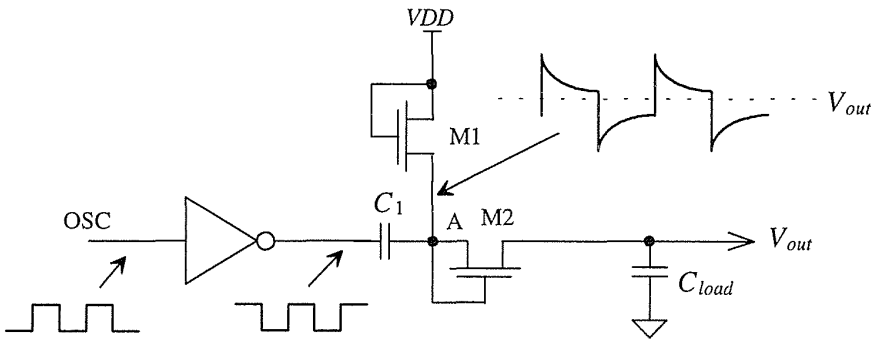



**Figure 18.13** Simple pump used to generate a voltage greater than *VDD*.

point A increases to $VDD + (VDD - V_{THN})$. This turns M2 on and charges $C_{load}$ to $2 \cdot (VDD - V_{THN})$ ($\approx$ 7 V for CN20), provided $C_1 >> C_{load}$ and the oscillator frequency allows the capacitors to fully charge or discharge before changing states. In most practical situations, $C_{load}$ and $C_1$ are comparable in size, and the output of the pump is loaded with a DC load. The result is an output voltage with a startup time; that is, $V_{out}$ does not immediately rise to $2 \cdot (VDD - V_{THN})$ but requires several oscillator cycles to reach steady state. Also, the output has a ripple dependent on the DC load. Figure 18.14 shows the simulation results for the circuit of Fig. 18.13 using minimum-size MOSFETs with $C_{load} = C_1 = 1$ pF and an oscillator frequency of 10 MHz. Using this simple pump with a DC load, such as a resistor, would require employing larger MOSFETs and capacitors.

The positive voltage pump uses n-channel MOSFETs, while the negative voltage pump, Fig. 18.15, uses p-channel MOSFETs. The reason for this comes from the requirement that the diode formed with the n+ (p+) implant used in the drain/source of the MOSFET combined with the p-substrate (n-well) does not become forward biased.

The capacitors used in Figs. 18.13 and 18.15 can be replaced with MOSFETs. An example of using n-channel MOSFETs in place of the capacitors is shown in Fig. 18.16. The main requirement on a MOSFET used as a capacitor is that its $V_{GS}$ remain greater than $V_{THN}$ at all possible operating conditions. In other words, the MOSFET must remain in the strong inversion region so that its capacitance is a constant $C'_{ox} \cdot W \cdot L$. The capacitor, $C_{load}$, of Fig. 18.16 must remain in strong inversion since $V_{out} >> V_{THN}$. Capacitor $C_1$ remains in strong inversion by realizing that when the inverter output is low, the voltage on the gate of $C_1$ is $VDD - V_{THN}$ ($= V_{GS}$). When the output of the inverter is high (*VDD*), the voltage on the gate of $C_1$ is $2 \cdot VDD - V_{THN}$. In both cases (the output of the inverter high and low), $V_{GS} = VDD - V_{THN}$, and the MOSFET remains in the strong inversion region.

### 18.3.1 Improving the Efficiency

Figure 18.17 shows a more efficient voltage pump. The improvement in efficiency comes from eliminating the threshold voltage drop at the gate and drain of M7. This allows the output to swing up to $2 \cdot VDD - V_{THN}$.

Microchip Ex. 1055, p. 391
Microchip v. Aptiv
IPR2024-00494

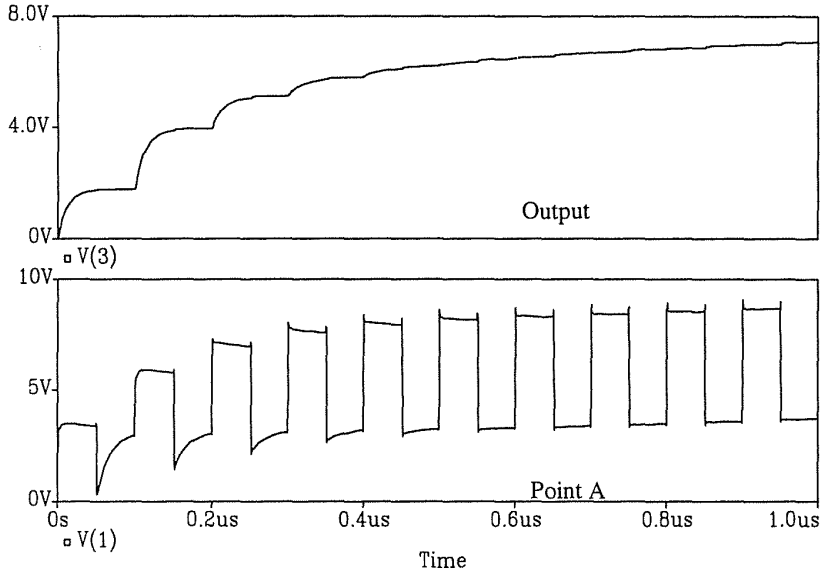



**Figure 18.14** Output of simple voltage pump.

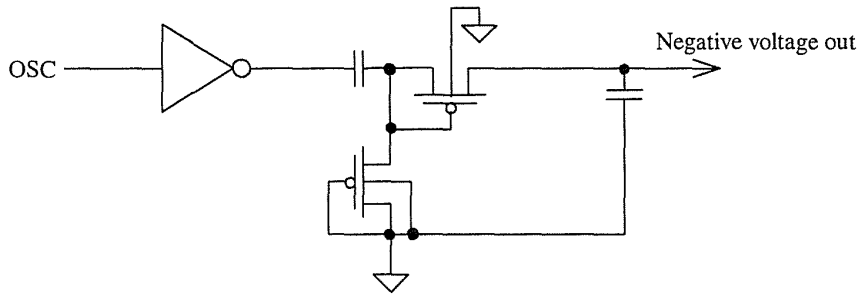



**Figure 18.15** Pump used to generate a voltage less than *VSS*.

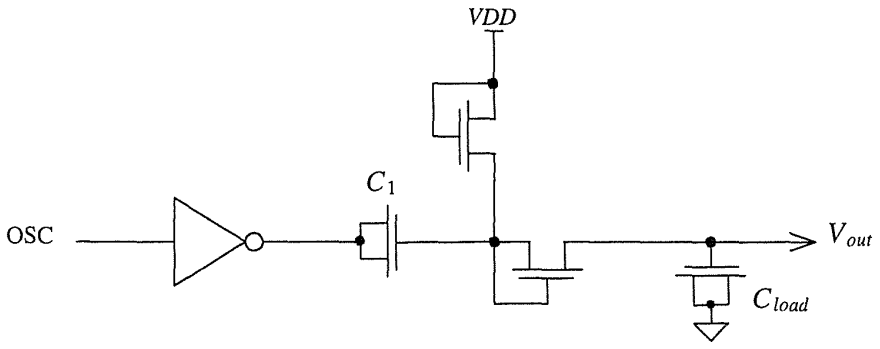



**Figure 18.16** Using MOSFETs as capacitors.

Microchip Ex. 1055, p. 392
Microchip v. Aptiv
IPR2024-00494

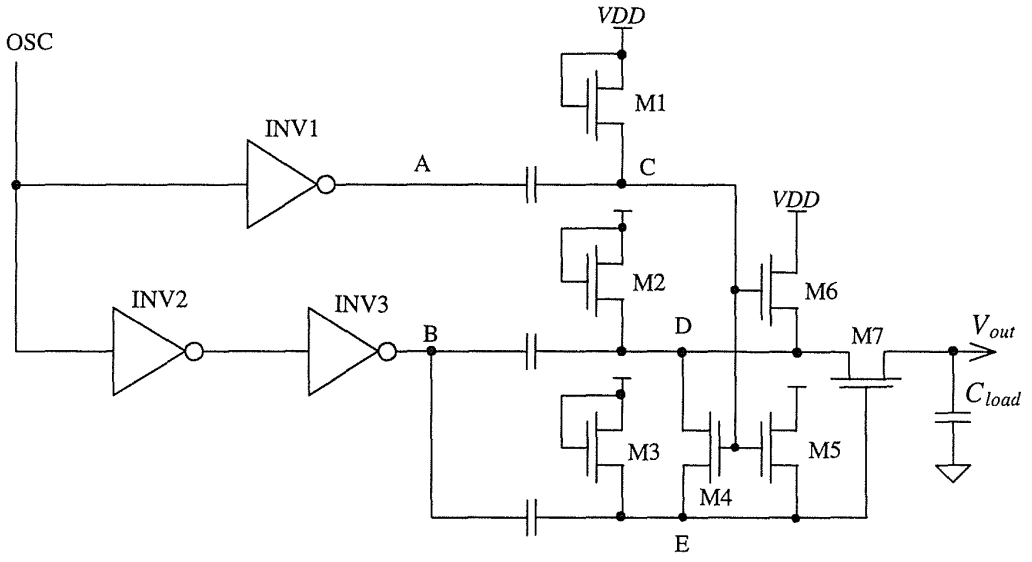



**Figure 18.17** Increased efficiency voltage pump.

To understand the operation of the circuit, let's assume that the voltage at point A is low and the voltage at point C is $VDD - V_{THN}$. When the output of INV1 goes high, point A is $VDD$ and the voltage at point C swings up to $2 \cdot VDD - V_{THN}$. This causes M4, M5, and M6 to turn on and pull points D and E to $VDD$. Now when point B goes high (point A goes back to zero), points D and E swing up to $2 \cdot VDD$ and the output goes to $2 \cdot VDD - V_{THN}$. Note that MOSFETs M2 and M3 are not needed; unless the pump drives a DC load , they never turn on. Also, separating points D and E is unnecessary unless the pump supplies a DC current.

### 18.3.2 Generating Higher Voltages

The voltage pumps of the last section are limited to voltages less than $2 \cdot VDD$ and greater than $2 \cdot VSS$. Figure 18.18 [3] shows a scheme for generating arbitrarily high voltages (limited by the breakdown voltage of the capacitors or the oxide breakdown of the MOSFETs). Again, the MOSFETs are used as diodes in this configuration.

In the following description of circuit operation, we will assume steady-state operation with no DC load present. When CLK is low, point A is pulled, by M1, to $VDD - V_{THN}$. When CLK goes high, point A swings up to $2 \cdot VDD - V_{THN}$. Diode M2 turns on, and point B charges to $2 \cdot VDD - 2 \cdot V_{THN}$. When CLK goes low, $\overline{CLK}$ goes high and point B swings up to $3 \cdot VDD - 2 \cdot V_{THN}$. When $\overline{CLK}$ goes low, point B swings back down to $2 \cdot VDD - 2 \cdot V_{THN}$. This operation proceeds through the circuit to the last stage of the multiplier. The output of the multiplier swings from $(N+1) \cdot VDD - N \cdot V_{THN}$ down to $(N + 1) \cdot VDD - (N + 1) \cdot V_{THN}$. $C_N$ can be made larger than the other capacitors to reduce this ripple.

Microchip Ex. 1055, p. 393
Microchip v. Aptiv
IPR2024-00494

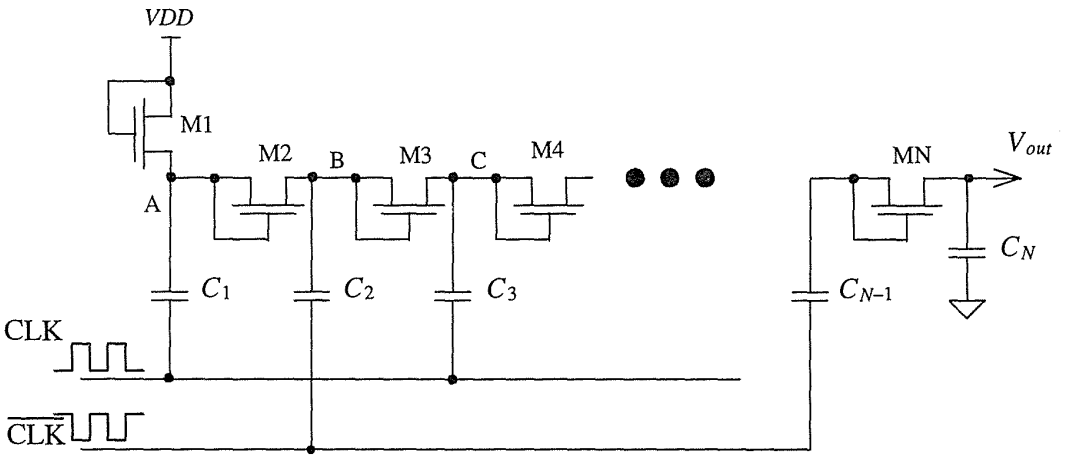



**Figure 18.18** N-stage voltage multiplier.

In practice, several problems exist with this circuit. The threshold voltage, due to the body effect, increases for each MOSFET proceeding from M1 to MN. Stray capacitance at each node to ground lowers the efficiency of the multiplier. When the circuit supplies a current charge, sharing exists between the capacitors in the circuit. In short, the output impedance of the voltage generator can be very high. This limits the circuit to supplying very small currents. Also, in a practical circuit, the output voltage should have some voltage-limiting circuit on the output of the multiplier to limit the output voltage and protect the circuit from breakdown. A simple limiting circuit is the series connection of several long L n-channel MOSFETs from the output of the circuit to ground.

### 18.3.3 Example

A common application of a voltage generator in digital circuits is generating a negative substrate bias [4]; that is, instead of tying the substrate to ground, the substrate is held at some negative voltage. Typically, this negative voltage is between –1 and –2 V. "Pumping" the substrate negative is common in virtually all DRAMs. A negative substrate bias has several benefits. It (1) stabilizes n-channel threshold voltages, (2) increases latch-up immunity after power up, (3) prevents forward biasing n+ to p-substrate pn junction, (4) allows chip inputs to go negative without forward biasing a pn junction, (5) prevents substrate from going locally above ground, (6) reduces depletion capacitances associated with the n+ to p-substrate junction and (7) reduces subthreshold leakage current.

A simple substrate pump is shown in Fig. 18.19. In this circuit we can use n-channel MOSFETs to generate a negative potential, unlike Fig. 18.15, since the negative voltage is connected to the substrate. In this situation, the drain/source implants of the n-channel MOSFETs cannot become forward biased. Note the absence of the load capacitance. It turns out that the capacitance of the substrate to everything

Microchip Ex. 1055, p. 394
Microchip v. Aptiv
IPR2024-00494

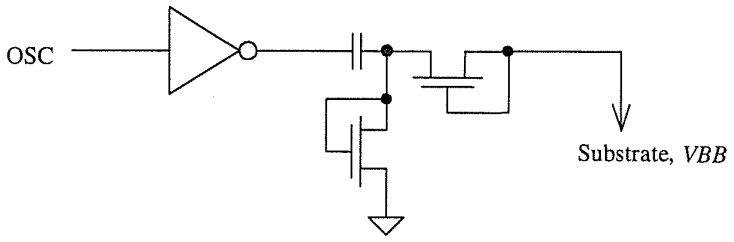



**Figure 18.19** Simple substrate pump.

else in the circuit presents a very large capacitance to the pump. In other words, the substrate itself is a very large capacitor.

An example oscillator circuit used to drive the substrate pump is shown in Fig. 18.20. This is a standard ring oscillator with a NAND gate used to enable/disable the oscillator. Capacitors are added at a few points in the middle of the oscillator to increase the delay and lower the frequency of the oscillator in order to allow the pump's capacitors to fully charge.

The final component in a substrate pump is the regulator, a circuit that senses the voltage on the substrate and enables or disables the substrate pump. Using a comparator with hysteresis, a precision voltage reference, and a level shifting circuit, the enable signal can be generated with the circuit of Fig. 18.21. The hysteresis of the comparator determines the amount of ripple on the substrate voltage. Forcing a constant current through the two MOSFETs, M1 and M2, compels their source-gate voltages to remain constant (note how the body effect is eliminated by using p-channel MOSFETs), causing the MOSFETs to behave as if they were batteries. The battery action of M1 and M2 shifts the substrate voltage up so that it lies in the common mode range of the comparator. The number of MOSFETs (in this case two) and the magnitude of the current $I$ determine the substrate voltage.

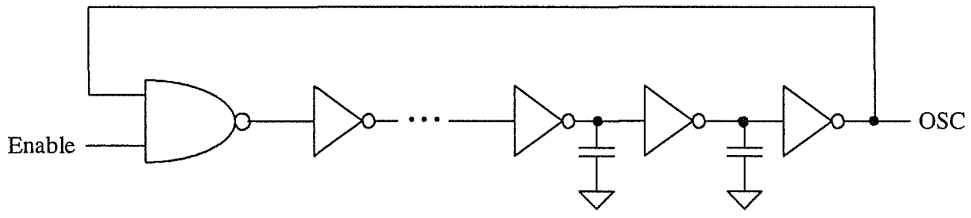



**Figure 18.20** Ring oscillator with enable.

Microchip Ex. 1055, p. 395
Microchip v. Aptiv
IPR2024-00494

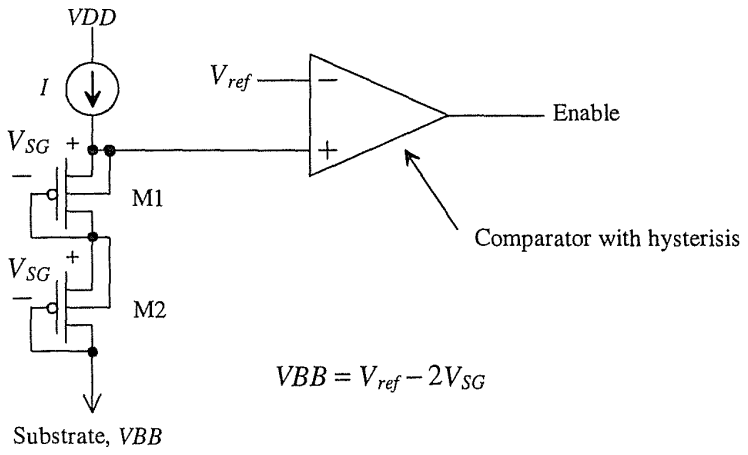



**Figure 18.21** Regulator circuit used in substrate pump.

A simpler and less accurate implementation of the regulator is shown in Fig. 18.22. The comparator and voltage reference are implemented with an inverter and M1-M3. MOSFET M3 causes the inverter to have hysteresis, that is, behave like a Schmitt trigger. MOSFETs M2 and M3 form an inverter with a switching point voltage of approximately $V_{THN}$. The current source of Fig. 18.22 is implemented with the long L MOSFET M4. The level shifting is performed with the n-channel MOSFETs M5 and M6. When point A gets above a potential of $V_{THN}$, the Enable output goes high, causing the substrate pump to turn on and drive the substrate voltage negative. When point A

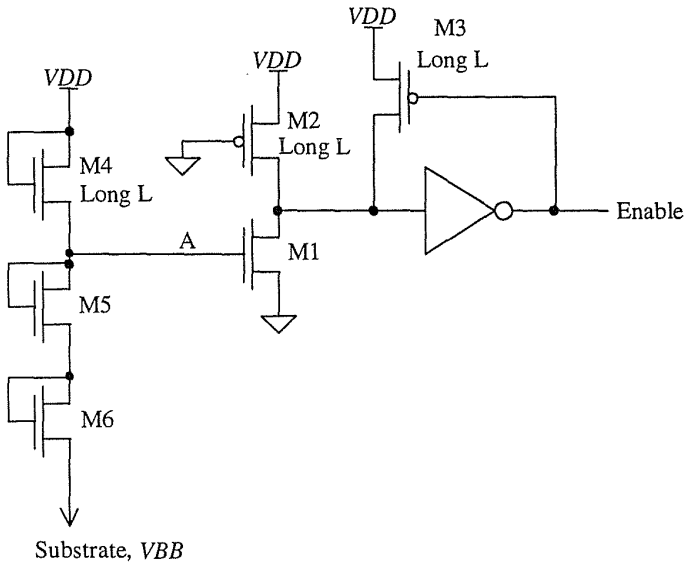



**Figure 18.22** Simpler substrate regulator.

Microchip Ex. 1055, p. 396
Microchip v. Aptiv
IPR2024-00494

gets pulled, via the substrate voltage through M5 and M6, below $V_{THN}$, the Enable output goes low and the pump shuts off. The substrate voltage generated with this circuit is approximately $-V_{THN}$ with body effect, which is approximately –1 V.

## REFERENCES

[1] D. A. Hodges and H. G. Jackson, *Analysis and Design of Digital Integrated Circuits,* McGraw-Hill Publishing Co., 2nd ed., 1988. ISBN 0-07-029158-6.

[2] R. Gregorian, K. W. Martin, and G. C. Temes, "Switched-Capacitor Circuit Design," *Proceedings of the IEEE,* Vol. 71, No. 8, August, pp. 941-966, 1983.

[3] J. F. Dickson, "On-Chip High-Voltage Generation in MNOS (sic) Integrated Circuits Using an Improved Voltage Multiplier Technique," *IEEE Journal of Solid State Circuits*, Vol. SC-11, No. 3, June 1976.

[4] P. Zagar, *4 MEG DRAM Tutorial-Course Notes,* Micron Semiconductor, Boise, Idaho.

[5] J. Wu, Y. Chang, and K. Chang, "1.2V CMOS Switched-Capacitor Circuits," *ISSCC*, 1996.

## PROBLEMS

**18.1** Design a Schmitt trigger with $V_{SPH}$ = 3 and $V_{SPL}$ = 2.5 V. Perform a SPICE DC sweep on your design, labeling the switching point voltages.

**18.2** Estimate $t_{PHL}$ and $t_{PLH}$ for the Schmitt trigger of Ex. 18.1, driving a 100 fF load capacitance.

**18.3** Design and simulate the operation of a Schmitt trigger-based oscillator with an output frequency of 10 MHz.

**18.4** Estimate the total input capacitance on the control voltage input for the VCO shown in Fig. 18.8.

**18.5** Determine, using SPICE simulations, $V_{SPH}$ and $V_{SPL}$ for the Schmitt trigger shown in Fig. 18.9.

**18.6** Design an astable multivibrator with an output oscillation frequency of 20 MHz. Use minimum-size inverters in the design. What is the $V_{SP}$ of the inverters?

**18.7** Design and simulate the operation of a one-shot that has an output pulse width of 100 ns. Comment on the maximum rate of retrigger and how ESD diodes connected to bonding pads will affect the circuit operation if the resistor and capacitor are bonded out.

**18.8** Generate, using SPICE, Fig. 18.14.

**18.9** Design a nominally 7 V voltage generator that can supply at most 1 μA of DC current (7 MEG resistor). Simulate the operation of your design with SPICE.

Microchip Ex. 1055, p. 397
Microchip v. Aptiv
IPR2024-00494

**18.10** Design and simulate the operation of a nominally –1 V substrate pump. Comment on the design tradeoffs.

**18.11** Figure P18.11 shows a voltage generator [5] useful in increasing the efficiency of the N-stage voltage multiplier. Describe the operation of the circuit. Use timing diagrams to illustrate your understanding.

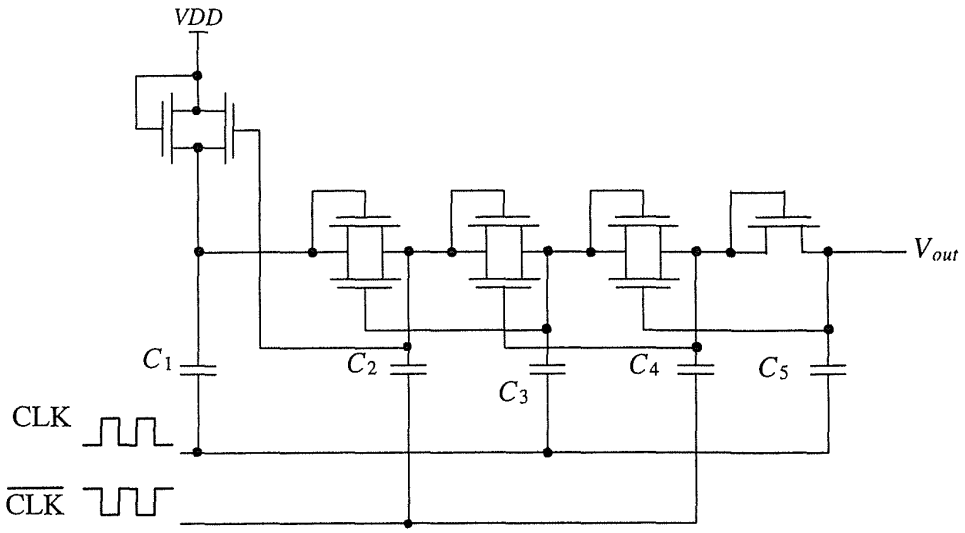



**Figure P18.11**

Microchip Ex. 1055, p. 398
Microchip v. Aptiv
IPR2024-00494

Chapter

19

# Digital Phase-Locked Loops

The digital phase-locked loop, DPLL, is a circuit that is used frequently in modern integrated circuit design [1]. Consider the waveform and block diagram of a communication system shown in Fig. 19.1. Digital data are loaded into the shift register at the transmitting end. The data are shifted out sequentially to the transmitter output driver. At the receiving end, where the data may be analog (i.e., not have well-defined amplitudes) after passing through the communication channel, the receiver amplifies and changes the data back into digital logic levels. The next logical step in this sequence of events is to shift the data back into a shift register at the receiver and process the received data. However, the absence of a clock signal makes this difficult. The DPLL performs the function of generating a clock signal which is locked or in synchronization with the incoming signal. The generated clock signal is used in the receiver to clock the shift register and thus recover the data. This application of a DPLL is often termed *a clock-recovery circuit* or *bit synchronization circuit.*

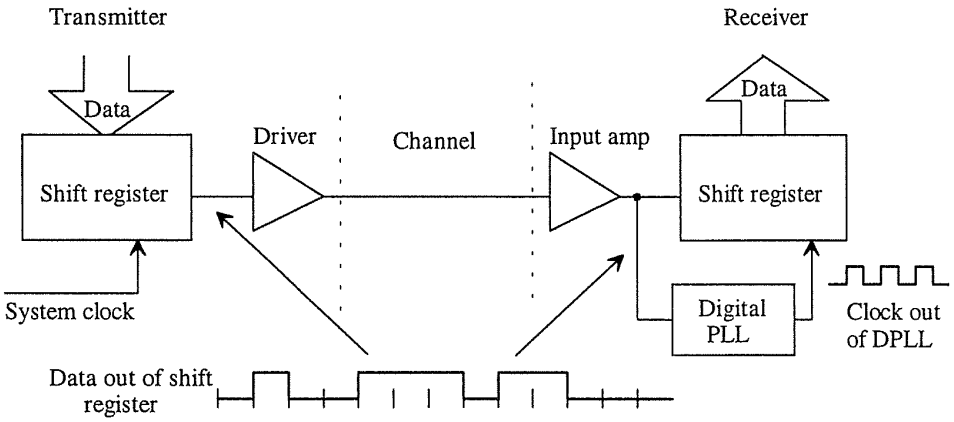



**Figure 19.1** Block diagram of a communication system using a DPLL for the generation of a clock signal.

Microchip Ex. 1055, p. 399
Microchip v. Aptiv
IPR2024-00494

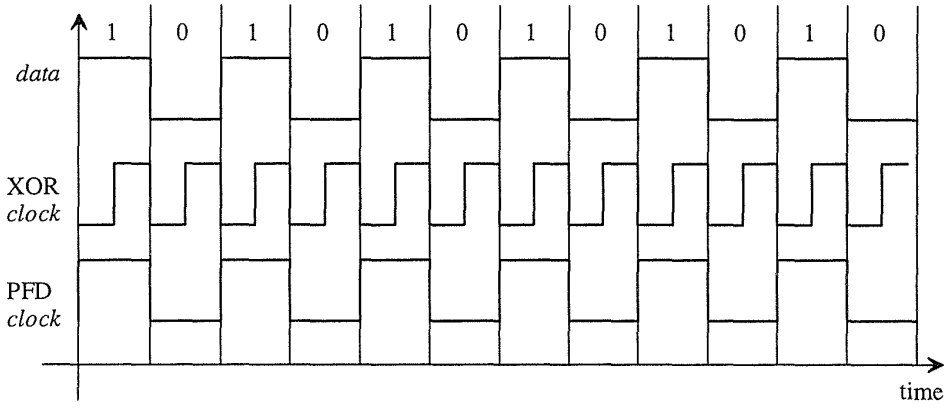



**Figure 19.2** Data input to DPLL in lock and possible clock outputs using the XOR phase detector and PFD.

A more detailed picture of the incoming data and possible clock signals out of the DPLL are shown in Fig. 19.2. The possible clock signals are labeled XOR *clock* and PFD (phase frequency detector) *clock* corresponding to the type of phase[1] detector (PD) used. For the XOR PD, the rising edge of the clock occurs in the center of the data, while for the PFD the rising edge occurs at the beginning of the data. The phase of the clock signal is determined by the PD used[2].

A block diagram of a DPLL is shown in Fig. 19.3. The PD generates an output signal proportional to the time difference between the *data in* and the divided down clock, *dclock*. This signal is filtered by a loop filter. The filtered signal, $V_{inVCO}$, is connected to the input of a voltage controlled oscillator (VCO). Each one of these blocks is discussed in detail in the following sections. Once each block is understood, we will put them together and discuss operation of the DPLL.

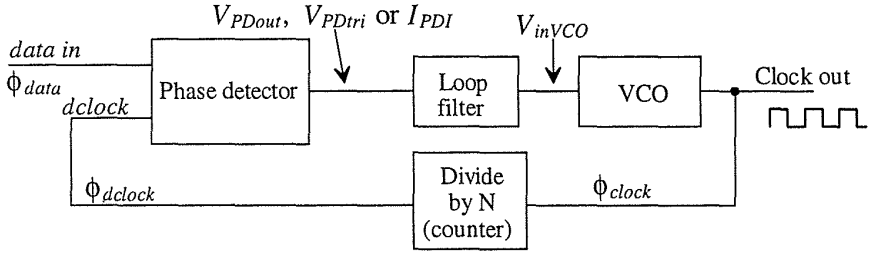



**Figure 19.3** Block diagram of a digital phase locked loop.

[1] A more correct name for digital applications is time difference detector (TDD).

[2] The PFD is rarely used in clock recovery applications. Figure 19.2 is simply used to illustrate the different phase relations resulting from an XOR PD or PFD in a locked DPLL.

Microchip Ex. 1055, p. 400
Microchip v. Aptiv
IPR2024-00494

## 19.1 The Phase Detector

The first component in our DPLL is the phase detector. The two types of phase detectors, an XOR gate and a phase frequency detector (PFD), have significantly different characteristics. This makes understanding the limitations and performance very important. Selection of the type of phase detector is the first step in a DPLL design.

### 19.1.1 The XOR Phase Detector

The XOR PD is simply an exclusive OR gate. When the output of the XOR is a pulse train with a 50 percent duty cycle (square wave), the DPLL is said to be in lock; or in other words, the clock signal out of the DPLL is synchronized to the incoming data, provided the conditions stated below are met. Consider the XOR PD shown in Fig. 19.4. Let's begin by assuming that the incoming data are a string of zeros and that a divide by two is used in the feedback loop. The output of the phase detector is simply a replica of the *dclock* signal. Since the *dclock* signal has a 50 percent duty cycle, it would appear that the DPLL is in lock. If a logic "1" is suddenly applied, there is no way to know if the clock signal is synchronized (the clock rising edge coincides with the center of the data bit) to the data. This leads to the first characteristic of an XOR PD;

1. The incoming data must have a minimum number of transitions over a given time interval.

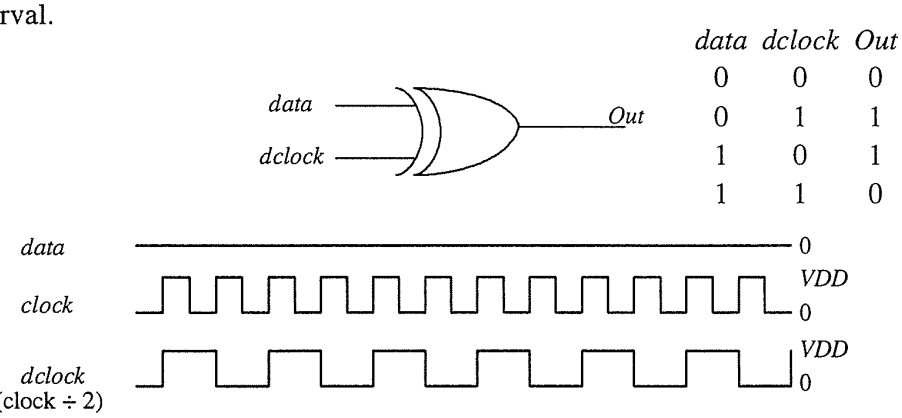



**Figure 19.4** Operation of the XOR phase detector.

Now consider the situation when the output of the phase detector, with the data input being a string of zeros, is applied to a simple RC low-pass filter (Fig. 19.5). If RC >> period of the clock signal, the output of the filter is simply *VDD*/2. This leads to the second characteristic of the XOR phase detector.

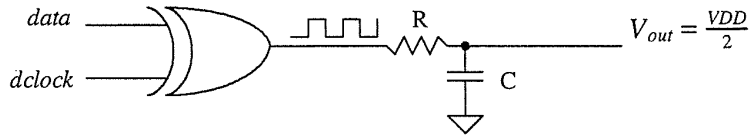



**Figure 19.5** How the filtered output of the phase detector becomes VDD/2.

Microchip Ex. 1055, p. 401
Microchip v. Aptiv
IPR2024-00494

2. With no input data, the filtered output of the phase detector is *VDD*/2.

The voltage out of the loop filter is connected to the input of the VCO. Consider the typical characteristics of a VCO shown in Fig. 19.6. The frequency of the square wave output of the VCO is $f_{center}$ when $V_{in}$ ( = $V_{center}$) is *VDD*/2 (typically). The other two frequencies of interest are the minimum and maximum oscillator frequencies, $f_{min}$ and $f_{max}$ possible, with input voltage $V_{min}$ and $V_{max}$ , respectively. It is important that the VCO continue to oscillate with no input data. Normally, the VCO is designed so that the nominal data input rate and the VCO center frequency are the same. This minimizes the time it takes the DPLL to lock (and is critical for proper operation of the XOR PD).

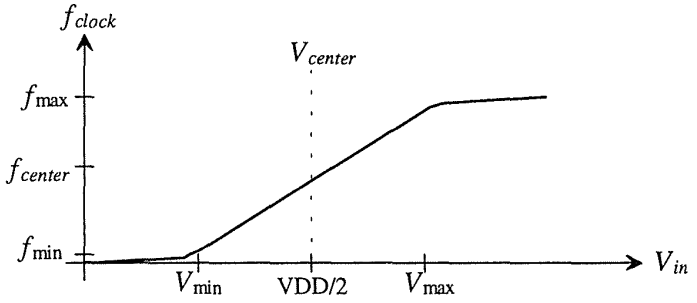



**Figure 19.6** Output frequency of VCO versus input control voltage.

Now let's consider an example input to the phase-detector and corresponding output. The data shown in Fig. 19.7 is leading the *dclock* signal. The corresponding output of the phase detector is also shown. If this output is applied to a low-pass filter, the result is an average voltage less than *VDD*/2. This causes the VCO frequency to decrease until the edge of the *dclock* is centered on the data.

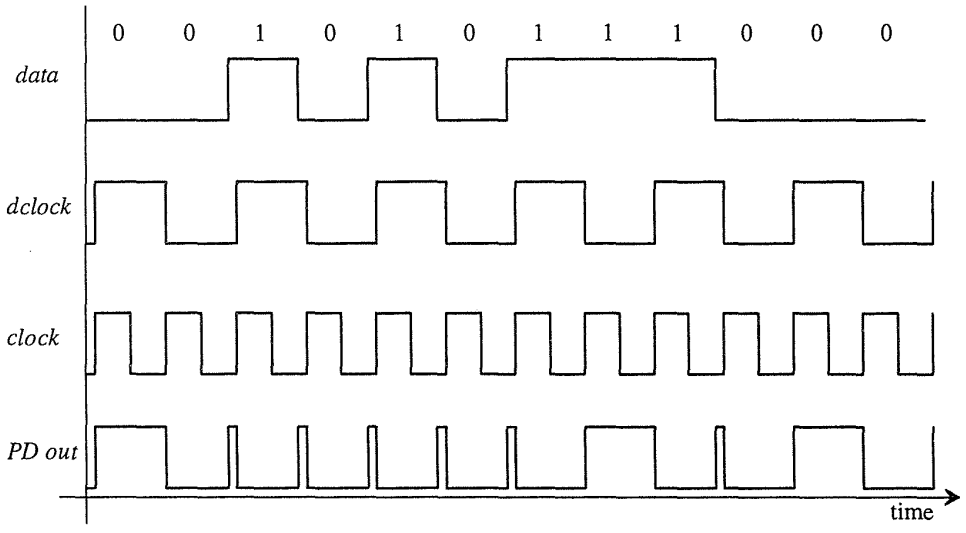



**Figure 19.7** Possible XOR phase-detector inputs and the resulting output.

Microchip Ex. 1055, p. 402
Microchip v. Aptiv
IPR2024-00494

3. The time it takes the loop to lock is dependent on the data pattern input to the DPLL and the loop-filter characteristics.

Since the output of the PD is averaged, or more correctly integrated, noise injected into the data stream (a false bit) can be rejected. A fourth characteristic of this PD is:

4. The XOR DPLL has good noise rejection.

Another important characteristic of a DPLL is whether or not it will lock on a harmonic of the input data. The XOR DPLL will lock on harmonics of the data. To prove that this is indeed the case, consider replacing any of the clock signals of Figs. 19.2, 19.4, or 19.7 with a clock at twice or half the frequency. The average of the waveforms will remain essentially the same. A fifth characteristic of a DPLL using an XOR gate is:

5. The VCO operating frequency range should be limited to frequencies much less than $2f_{clock}$ and much greater than $0.5f_{clock}$ where $f_{clock}$ is the nominal clock frequency for proper lock with a XOR PD.

The loop filter used with this type of PD is a simple RC low-pass filter as shown in Fig. 19.5. Since the output of the PD is oscillating, the output of the filter will show a ripple as well, even when the loop is locked. This modulates the clock frequency, an unwanted characteristic of a DPLL using the XOR PD. This characteristic can be added to our list:

6. A ripple on the output of the loop filter with a frequency equal to the clock frequency will modulate the control voltage of the VCO.

To characterize the phase detector (see Fig. 19.8) we can define the time difference between the rising edge of the *dclock* and the beginning of the *data* as $\Delta t$. The phase difference between the *dclock* and *data*, $\phi_{data} - \phi_{dclock}$, is given by

$$\Delta\phi = \phi_{data} - \phi_{dclock} = \frac{\Delta t}{T_{dclock}} \cdot 2\pi \text{ (radians)} \tag{19.1}$$

or, in terms of the DPLL output clock frequency,

$$\Delta\phi = \frac{\Delta t}{2T_{clock}} \cdot 2\pi \tag{19.2}$$

$$f_{clock} = \frac{1}{T_{clock}} = 2f_{dclock} = \frac{2}{T_{dclock}} \tag{19.3}$$

When the loop is locked, the *clock* rising edge is centered on the data; the time difference, $\Delta t$, between the *dclock* rising edge and the beginning of the data is simply $T_{clock}/2$ or $T_{dclock}/4$ (see Fig. 19.8c). Therefore, the phase difference between *dclock* and the *data*, under locked conditions, may be written as

$$\Delta\phi = \frac{\pi}{2} \tag{19.4}$$

Microchip Ex. 1055, p. 403
Microchip v. Aptiv
IPR2024-00494

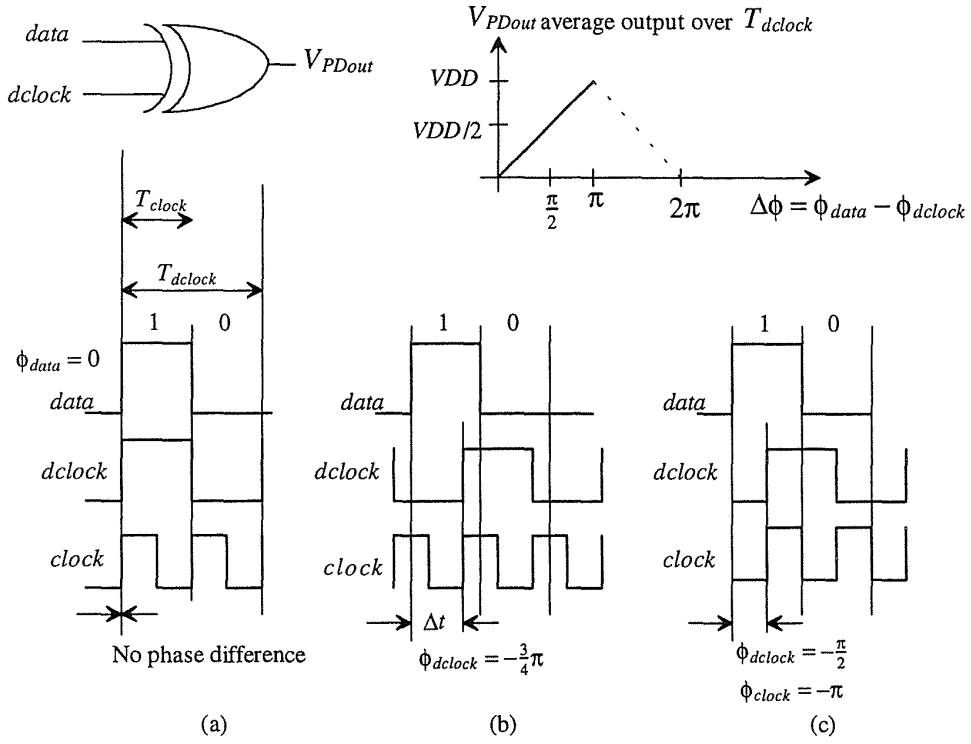



**Figure 19.8** XOR PD output for various inputs (assuming input data are a string of alternating ones and zeros).

Note that the phase difference between *clock* and *data* when in lock is $\pi$.

The average voltage out of the phase detector (Fig. 19.8) may be expressed by

$$V_{PDout} = VDD \cdot \frac{\Delta\phi}{\pi} = K_{PD} \cdot \overbrace{\Delta\phi}^{input} \tag{19.5}$$

where the gain of the PD may be written as

$$K_{PD} = \frac{VDD}{\pi} \text{ (V/radians)} \tag{19.6}$$

To aid in the understanding of these equations, consider the diagrams shown in Fig. 19.8. If the edges of the clock and data are coincident in time (Fig. 19.8a), the XOR output, $V_{PDout}$, is 0 V and the phase difference is 0. The loop filter averages the output of the PD and causes the VCO to lower its output frequency. This causes $\Delta\phi$ to increase, thus increasing $V_{PDout}$. Depending on the selection of the loop filter, this increase could cause the rising edges to increase beyond, or *overshoot*, the desirable center point as shown in Fig. 19.8b. In this case, the phase difference is $-\frac{3}{4}\pi$, and $V_{PDout}$ is $\frac{3}{4}VDD$. Figure 19.8c shows the condition when the loop is in lock and the phase difference is $\pi/2$.

Microchip Ex. 1055, p. 404
Microchip v. Aptiv
IPR2024-00494

Here we should point out the importance of having the VCO center frequency, $f_{center}$, equal to the desired clock frequency. If $f_{center} = f_{clock}$, the VCO control voltage, depending on the loop filter, will look similar to Fig. 19.9a during acquisition (the loop trying to lock). If these frequencies are not equal, the control voltage will oscillate, causing the clock to move, in time, around some other point than the center of the data bit (Fig. 19.9b). The actual control voltages will look different from those portrayed in Fig. 19.9 because the VCO control voltage is dependent on the input data pattern as discussed earlier.

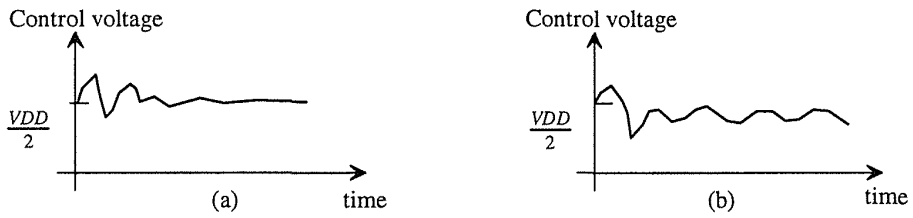



**Figure 19.9** Average output voltage of phase detector during acquisition.

To summarize the design criteria of the VCO with the XOR PD, we desire that;

1. The center frequency, $f_{center}$, should be equal to the clock frequency when the VCO control voltage is *VDD*/2.

2. The maximum and minimum oscillation frequencies, $f_{max}$ and $f_{min}$, of the VCO should be selected to avoid locking on harmonics of the input data.

3. It is important that the VCO duty cycle be 50 percent. If this is not the case, the DPLL will have problems locking, or once locked the clock will jitter (move around in time).

### 19.1.2 Phase Frequency Detector

A schematic diagram of the phase frequency detector is shown in Fig. 19.10 [1]. The output of the PFD depends on both the phase and frequency of the inputs. This type of phase detector is also termed a sequential phase detector. It compares the leading edges of the *data* and *dclock*. A *dclock* rising edge cannot be present without a data rising edge. To aid in understanding the PFD, consider the examples shown in Fig. 19.11. The first thing we notice is that the *data* pulse width and the *dclock* pulse width do not matter. If the rising edge of the *data* leads the *dclock* rising edge (Fig. 19.11a), the "up" output of the phase detector goes high while the *Down* output remains low. This causes the *dclock* frequency to increase, having the effect of moving the edges closer together. When the *dclock* signal leads the data (Fig. 19.11b) *Up* remains low while the *Down* goes high a time equal to the phase difference between *dclock* and *data*. Figure 19.11c shows the condition of the loop being locked. Notice that, unlike the XOR PD, the outputs remain low when the loop is locked. Again, several characteristics of the PFD can be described:

Microchip Ex. 1055, p. 405
Microchip v. Aptiv
IPR2024-00494

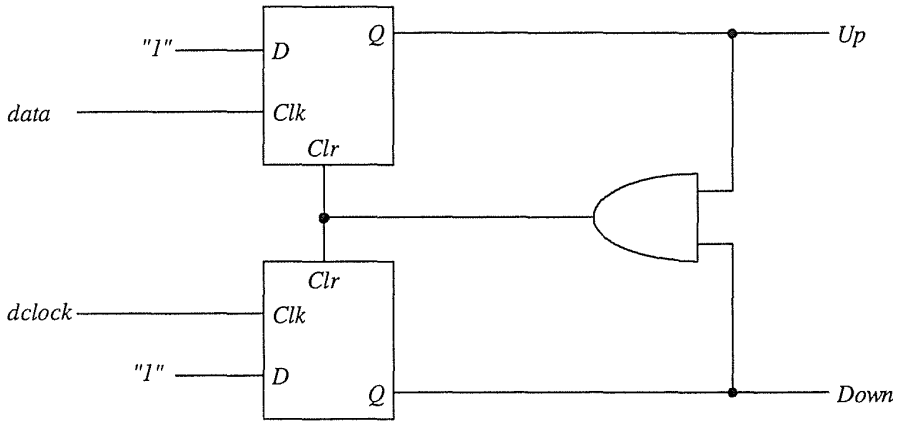



**Figure 19.10** Phase frequency detector (PFD).

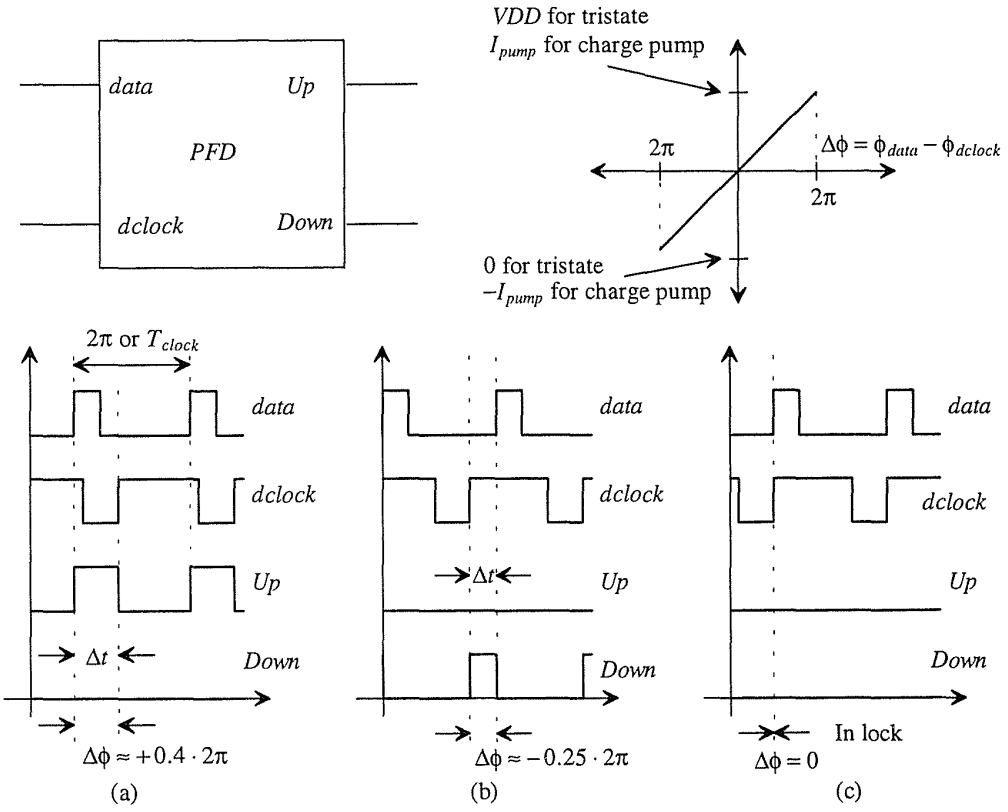



**Figure 19.11** PFD phase-detector inputs and outputs.

Microchip Ex. 1055, p. 406
Microchip v. Aptiv
IPR2024-00494

1. A rising edge from the *dclock* and *data* must be present when doing a phase comparison.

2. The width of the *dclock* and the *data* is irrelevant.

3. The PFD will not lock on a harmonic of the data.

4. The outputs (*Up* and *Down*) of the PFD are both logic low when the loop is in lock, eliminating ripple on the output of the loop filter.

5. This PFD has poor noise rejection; a false edge on either the *data* or the *dclock* inputs will drastically affect the output of the PFD.

The output of the PFD should be combined into a single output for driving the loop filter. There are two methods of doing this, both of which are shown in Fig. 19.12. The first method is called a *tri-state* output. When both signals, *Up* and *Down*, are low, both MOSFETs are off and the output is in a high-impedance state. If the *Up* signal goes high, M2 turns on and pulls the output up to *VDD* while if the *Down* signal is high the output is pulled low through M1. The main problem that exists with this configuration is that the power supply variations can significantly affect the output voltage when M2 is on. The effect is to modulate the VCO control voltage. This wasn't as big a problem when the XOR PFD was used because of the averaging taking place. The second configuration shown in this figure is the so-called *charge pump* [2]. MOS current sources are placed in series with M1 and M2. When the PFD *Up* signal goes high, M2 turns on, connecting the current source to the loop filter. Since the current source can be made insensitive to supply variations, modulation of the VCO control voltage is absent.

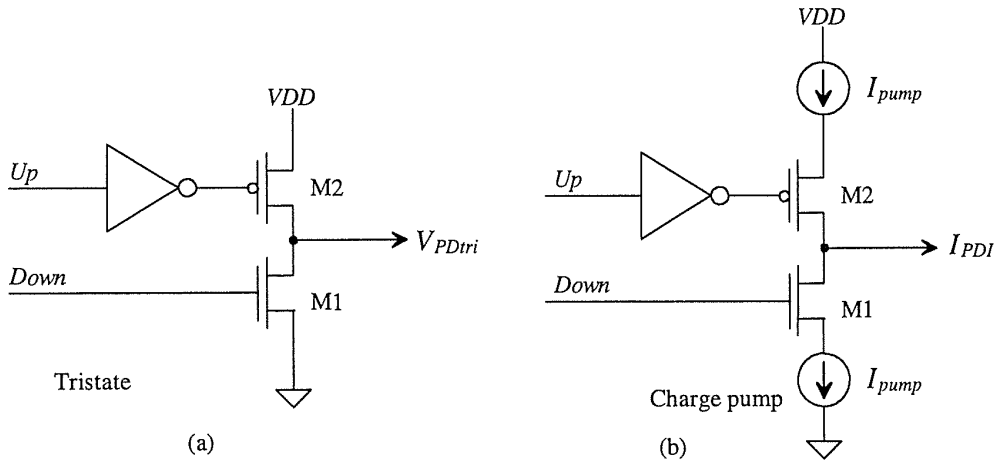



**Figure 19.12** (a) Tri-state and (b) charge pump outputs of the PFD.

Microchip Ex. 1055, p. 407
Microchip v. Aptiv
IPR2024-00494

We can characterize the PFD in much the same manner as we did the XOR PD. We will assume that $f_{clock}$ and $f_{dclock}$ are equal in this analysis. Therefore, the feedback loop, Fig. 19.3, divides by one ($N = 1$). If we again assume that the time difference between the rising edges of the *data* and *dclock* is labeled $\Delta t$ and the time between leading edges of the *clock* (or the time differences between the leading edges of the *data* since we must have both leading edges present to do a phase comparison) is labeled $T_{clock}$ then we can write the phase ($T_{clock} = T_{dclock}$) as

$$\Delta\phi = \frac{\Delta t}{T_{clock}} \cdot 2\pi \text{ (radians)} \tag{19.7}$$

The phase difference, $\Delta\phi$, is zero when the loop is in lock. The output voltage of the PFD using the tri-state output configuration (see Fig. 19.11) is

$$V_{PDtri} = \frac{VDD - 0}{4\pi} \cdot \Delta\phi = K_{PDtri} \cdot \Delta\phi \tag{19.8}$$

where the gain is,

$$K_{PDtri} = \frac{VDD}{4\pi} \text{ (volts/radian)} \tag{19.9}$$

If the output of the PFD uses the charge-pump configuration, the output current can be written (again see Fig. 19.11) as

$$I_{PDI} = \frac{I_{pump} - (-I_{pump})}{4\pi} \cdot \Delta\phi = K_{PDI} \cdot \Delta\phi \tag{19.10}$$

where

$$K_{PDI} = \frac{I_{pump}}{2\pi} \text{ (amps/radian)} \tag{19.11}$$

The loop filters used with tri-state and the charge-pump outputs are shown in Fig. 19.13. The first loop filter has a transfer function given by

$$V_{inVCO} = \frac{1 + j\omega R_2 C}{1 + j\omega(R_1 + R_2)C} \cdot V_{PDtri} = K_F \cdot V_{PDtri} \tag{19.12}$$

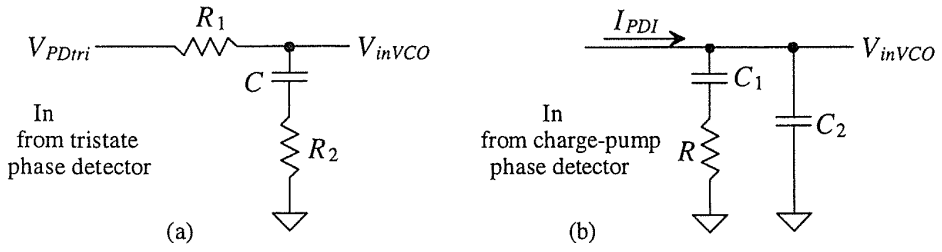



**Figure 19.13** Loop filters for (a) tristate output and (b) charge-pump output.

Microchip Ex. 1055, p. 408
Microchip v. Aptiv
IPR2024-00494

The charge-pump loop-filter transfer function is given (noting the input variable is a current while the output is a voltage) by

$$V_{inVCO} = I_{PDI} \cdot \frac{1 + j\omega RC_1}{j\omega(C_1 + C_2) \cdot \left[1 + j\omega R\frac{C_1 C_2}{C_1 + C_2}\right]} = K_F \cdot I_{PDI} \qquad (19.13)$$

To qualitatively understand how these loop filters work, let's begin by considering the loop filter for use with the tri-state output. For slow variations in the phase difference, the filter acts like an integrator averaging the output of the PD. For fast variations, however, the filter looks like a resistive divider without any integration. This allows the loop filter to track fast variations in the time difference between the rising edges. A similar discussion can be made for the loop filter used with the charge pump. For slow variations in the phase, the current, $I_{pump}$, linearly charges $C_1$ and $C_2$. This gives an averaging effect. For fast variations, the charge pump simply drives the resistor $R$ (assuming $C_2$ is small), eliminating the averaging and allowing the VCO to track quickly moving variations in the input data.

The design requirements of the VCO used with the PFD can be much more relaxed than those for the XOR PD; therefore, it is preferred over the XOR PD. Since the output of the PD can be a voltage from 0 to *VDD,* the requirement that $f_{center} = f_{clock}$ is not present, although it is still a good idea to design the VCO so that this is true. The oscillation range, $f_{max} - f_{min}$, is not limited by the harmonics of $f_{clock}$ since the PFD will not lock on a harmonic of the clock frequency. The VCO clock duty cycle is irrelevant with the PFD since the PD looks only at rising edges. In high-speed or data communications applications, however, one may be forced to use the XOR PD.

## 19.2 The Voltage-Controlled Oscillator

The gain of the voltage-controlled oscillator is simply the slope of the curves given in Fig. 19.6. This gain can be written as

$$K_{VCO} = 2\pi \cdot \frac{f_{\max} - f_{\min}}{V_{\max} - V_{\min}} \text{ (radians/s} \cdot \text{V)} \qquad (19.14)$$

The VCO output frequency, $f_{clock}$, is related to the VCO input voltage (see Fig. 19.6) by

$$\omega_{clock} = 2\pi \cdot f_{clock} = K_{VCO} \cdot V_{inVCO} + \omega_o \text{ (radians/s)} \qquad (19.15)$$

where $\omega_0$ is a constant. However, the variable we are feeding back is not frequency but phase. The phase of the VCO clock output is related to $f_{clock}$ by

$$\phi_{clock} = \int \omega_{clock} \cdot dt = \frac{K_{VCO}}{j\omega} \cdot V_{inVCO} \text{ (radians)} \qquad (19.16)$$

where this signal can be related to the $\phi_{dclock}$ by

$$\phi_{dclock} = \frac{1}{N} \cdot \phi_{clock} = \beta \cdot \phi_{clock} \qquad (19.17)$$

where $N$ is the divide by count and $\beta$ is the feedback factor.

Microchip Ex. 1055, p. 409
Microchip v. Aptiv
IPR2024-00494

### 19.2.1 The Current-Starved VCO

The current-starved VCO is shown schematically in Fig. 19.14. Its operation is similar to the ring oscillator discussed in Ch. 11. MOSFETs M2 and M3 operate as an inverter, while MOSFETs M1 and M4 operate as current sources. The current sources, M1 and M4, limit the current available to the inverter, M2 and M3; in other words, the inverter is starved for current. The MOSFETs M5 and M6 drain currents are the same and are set by the input control voltage. The currents in M5 and M6 are mirrored in each inverter/current source stage. An important property of the VCO used in any of the CMOS DPLLs discussed in this chapter is the input impedance. The filter configurations we have discussed rely on the fact that the input resistance of the VCO is practically infinite and the input capacitance is small compared to the capacitances present in the loop filter. Attaining infinite input resistance is usually an easy part of the design. For the charge-pump configuration, the input capacitance of the VCO can be added to $C_2$.

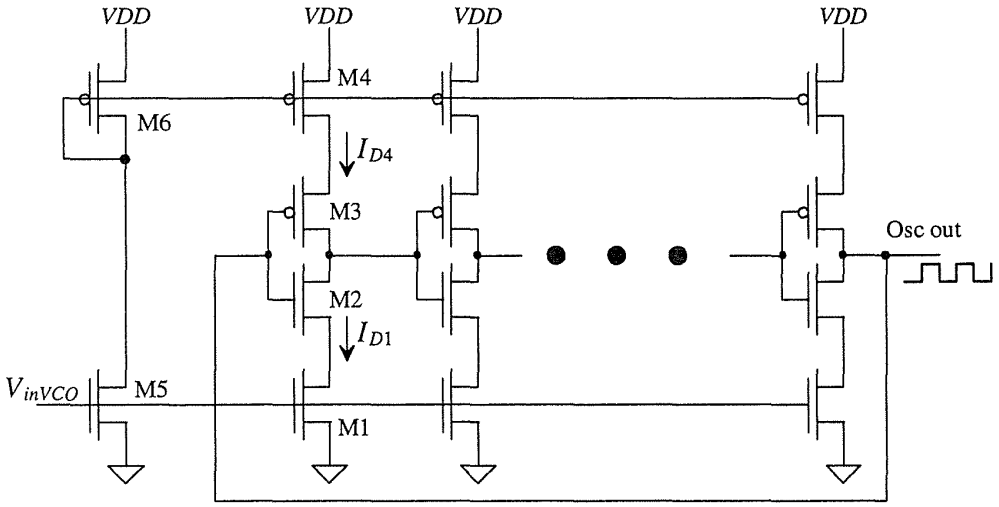



**Figure 19.14** Current-starved VCO.

To determine the design equations for use with the current-starved VCO, consider the simplified schematic of one stage of the VCO shown in Fig. 19.15. The total capacitance on the drains of M2 and M3 is given by

$$C_{tot} = C_{out} + C_{in} = \overbrace{C'_{ox}(W_pL_p + W_nL_n)}^{C_{out}} + \overbrace{\frac{3}{2}C'_{ox}(W_pL_p + W_nL_n)}^{C_{in}} \quad (19.18)$$

which is simply the output and input capacitances of the inverter as discussed in Ch. 11. This equation can be written in a more useful form as

Microchip Ex. 1055, p. 410
Microchip v. Aptiv
IPR2024-00494

$$C_{tot} = \frac{5}{2} C'_{ox}(W_p L_p + W_n L_n) \tag{19.19}$$

The time it takes to charge $C_{tot}$ from zero to $V_{SP}$ with the constant-current $I_{D4}$ is given by

$$t_1 = C_{tot} \cdot \frac{V_{SP}}{I_{D4}} \tag{19.20}$$

while the time it takes to discharge $C_{tot}$ from $VDD$ to $V_{SP}$ is given by

$$t_2 = C_{tot} \cdot \frac{VDD - V_{SP}}{I_{D1}} \tag{19.21}$$

If we set $I_{D4} = I_{D1} = I_D$ (which we will label $I_{Dcenter}$ when $V_{inVCO} = VDD/2$), then the sum of $t_1$ and $t_2$ is simply

$$t_1 + t_2 = \frac{C_{tot} \cdot VDD}{I_D} \tag{19.22}$$

The oscillation frequency of the current-starved VCO for $N$ (an odd number $\geq 5$) of stages is

$$f_{osc} = \frac{1}{N(t_1 + t_2)} = \frac{I_D}{N \cdot C_{tot} \cdot VDD} \tag{19.23}$$

which is equal to

$$= f_{center}(@\, V_{inVCO} = VDD/2 \text{ and } I_D = I_{Dcenter})$$

Equation (19.23) gives the center frequency of the VCO when $I_D = I_{Dcenter}$. The VCO stops oscillating, neglecting subthreshold currents, when $V_{inVCO} < V_{THN}$. Therefore we can define

$$V_{\min} = V_{THN} \text{ and } f_{\min} = 0 \tag{19.24}$$

The maximum VCO oscillation frequency, $f_{max}$, is determined by finding $I_D$ when $V_{inVCO} = VDD$. At the maximum frequency then, $V_{max} = VDD$.

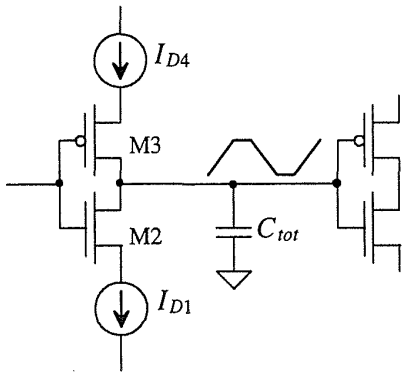



**Figure 19.15** Simplified view of a single stage of the current-starved VCO.

Microchip Ex. 1055, p. 411
Microchip v. Aptiv
IPR2024-00494

The output of the current-starved VCO shown in Fig. 19.14 normally has its output buffered through one or two inverters. Attaching a large load capacitance on the output of the VCO can significantly affect the oscillation frequency or lower the gain of the oscillator enough to kill oscillations altogether.

The average current drawn by the VCO (see Ch. 11) is

$$I_{avg} = N \cdot \frac{VDD \cdot C_{tot}}{T} = N \cdot VDD \cdot C_{tot} \cdot f_{osc} \tag{19.25}$$

or

$$I_{avg} = I_D \tag{19.26}$$

The average power dissipated by the VCO is

$$P_{avg} = VDD \cdot I_{avg} = VDD \cdot I_D \tag{19.27}$$

If we include the power dissipated by the mirror MOSFETs, M5 and M6, the power is doubled from that given by Eq. (19.27), assuming that $I_D = I_{D5} = I_{D6}$. For low-power dissipation we must keep $I_D$ low, which is equivalent to stating that for low-power dissipation we must use a low-oscillation frequency.

**Example 19.1**

Design a current-starved VCO with $f_{center}$ = 100 MHz using the CN20 process parameters. Estimate the power dissipation. Simulate the design using SPICE.

We begin by calculating the total capacitance, $C_{tot}$. Using Eq. (19.19) and assuming the inverters, M2 and M3, are sized for equal drive, that is, $L_n = L_p = 2$ μm, $W_n = 3$ μm, and $W_p = 9$ μm, the capacitance is

$$C_{tot} = \frac{5}{2} \cdot 800 \frac{\text{aF}}{\mu\text{m}^2} \cdot (9 \cdot 2 + 3 \cdot 2)\mu\text{m}^2 = 48 \text{ fF}$$

Since the oscillation frequency is relatively large, compare this with the five-stage ring oscillator described in Ch. 11. We will use the minimum number of stages, that is, 5. The center drain current, using Eq. (19.23), is given by

$$I_{Dcenter} = 100 \text{ MHz} \cdot 5 \cdot 48 \text{ fF} \cdot 5 \text{ V} = 120 \text{ μA}$$

To determine the size of M5 and M1 of Fig. 19.14, we solve for *W/L* in

$$I_{Dcenter} = \frac{\beta_5}{2}(V_{GS} - V_{THN})^2 = \frac{50 \frac{\mu A}{V^2}}{2} \cdot \frac{W_n}{L_n}(2.5 - 0.83)^2 = 120 \text{ μA}$$

giving $W_n/L_n$ = 1.72. We will use $W_n$ = 5 μm and $L_n$ = 3 μm as a close approximation. We use these values to set the size of M5 and M1 (of each stage). Sizing the p-channel we require, $\beta_6 = \beta_5$; this causes M5 to go into the triode region when $V_{inVCO}$ is approximately 3.5 and above. This requirement sets the size of M6 and M4 at $W_p$ = 15 μm and $L_p$ = 3 μm. The power dissipated, using Eq. (19.27), is 1.2 mW. The simulation results are shown in Fig. 19.16.

Microchip Ex. 1055, p. 412
Microchip v. Aptiv
IPR2024-00494

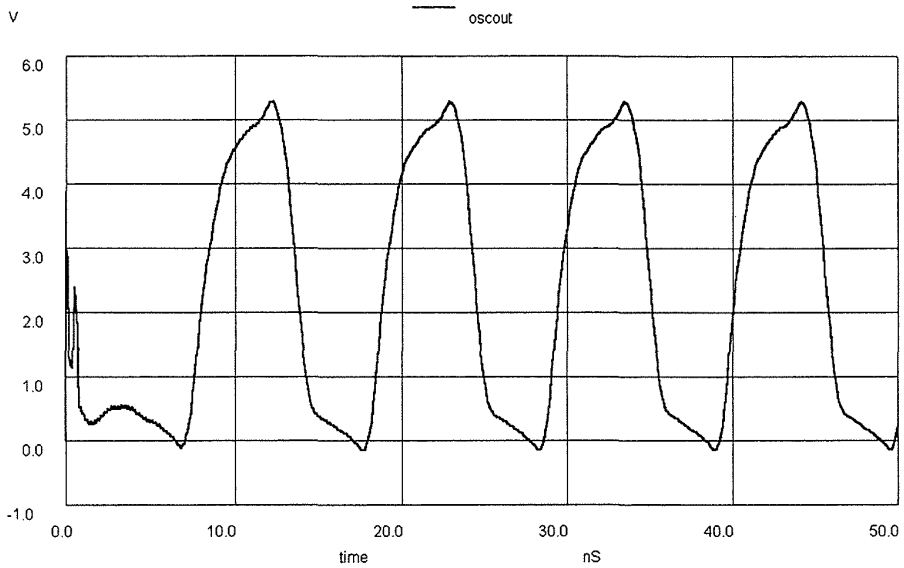



**Figure 19.16** Simulation results for the oscillator of Ex. 19.1 (unbuffered output). Simulations did not include drain and source depletion capacitances, that is, the perimeters and areas of the drain and source where not specified.

The simulation results did not include a specification of drain and source areas. Including these capacitances causes the $f_{center}$ to drop to 80 MHz. ■

### 19.2.2 Source Coupled VCOs

Another variety of VCO is shown in Fig. 19.17 [3]. These circuits can be designed to dissipate less power than the current-starved VCO of the last section for a given frequency. The major disadvantage of these configurations is the need for a capacitor, something that may not be available in a single-poly pure digital process without using parasitics for example, a metal1 to metal2 capacitor and the reduced output voltage swing. However, this configuration is useful when the VCO center frequency is set by an external capacitor; that is, the capacitor shown in the figure is bonded out.

To understand the operation, let's consider the NMOS source coupled VCO of Fig. 19.17a. The operation of the CMOS source coupled VCO of Fig. 19.17b is identical to the NMOS VCO except for the fact that the load MOSFETs M3 and M4 pull the outputs to $VDD - V_{THN}$ for the NMOS VCO and $VDD$ for the CMOS VCO. The MOSFETs M5 and M6 behave as constant-current sources sinking a current $I_D$. MOSFETs M1 and M2 operate as switches. If M1 is off and M2 is on, the drain of M1 is pulled to $VDD - V_{THN}$ by M3. It is very important to remember the body effect in this discussion. For example, using the CN20 process, the threshold voltage of M3 is approximately 1.5 V so that the maximum output swing is limited to 3.5 V when the supply voltage is 5 V. Since the gate of M2 is at $VDD - V_{THN}$, the source and drain (the *Output*) of M2 are approximately $VDD - 2 \cdot V_{THN}$. This is the minimum output voltage. The output voltage swing is limited to $V_{THN}$, or approximately 1.5 V, a swing from

Microchip Ex. 1055, p. 413
Microchip v. Aptiv
IPR2024-00494

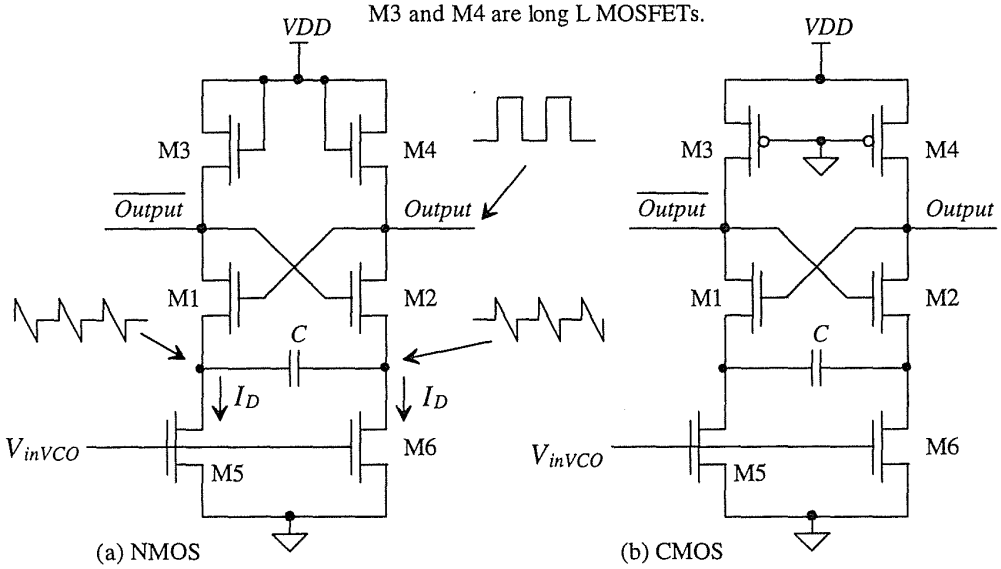



**Figure 19.17** Source coupled voltage-controlled oscillators (also known as source coupled multivibrators).

approximately 3.5 V to 2 V. The oscillator would require a buffer, possibly an inverter or self-biased diff-amp circuit to restore CMOS logic levels.

The simplified schematic shown in Fig. 19.18 with M1 off and M2 on will be helpful in determining the oscillator frequency. The *Output*, gate of M1 is approximately $VDD - 2 \cdot V_{THN}$ and is held at this voltage through M2 until M1 turns on and M2 turns off. Initially, at the moment when M1 turns off and M2 turns on, point X is $VDD - V_{THN}$. The current through $C$, $I_D$, causes point X to discharge down toward ground. When point X gets down to $VDD - 3 \cdot V_{THN}$, M1 turns on and M2 turns off. In

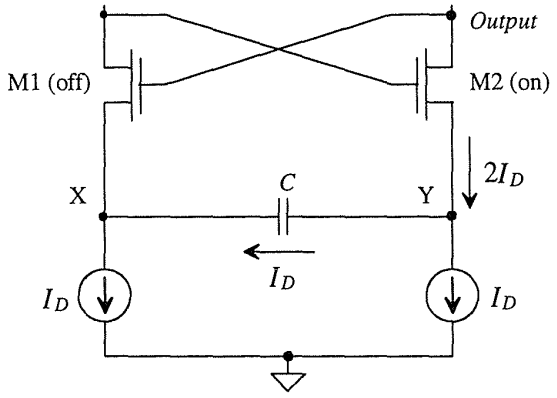



**Figure 19.18** Simplified schematic of source coupled oscillator, M1 is on and M2 is off.

Microchip Ex. 1055, p. 414
Microchip v. Aptiv
IPR2024-00494

other words, the voltage at point X changed a total of $2 \cdot V_{THN}$ before switching took place. The time it takes point X to change $2 \cdot V_{THN}$ is given by

$$\Delta t = C \cdot \frac{2 \cdot V_{THN}}{I_D} \tag{19.28}$$

Since the circuit is symmetrical, two of these discharge times are needed for each cycle of the oscillator. The frequency of oscillation is given by

$$f_{osc} = \frac{1}{2\Delta t} = \frac{I_D}{4 \cdot C \cdot V_{THN}} \tag{19.29}$$

The waveforms at the points X, Y, and *Output* are shown in Fig. 19.19 for continuous time operation.

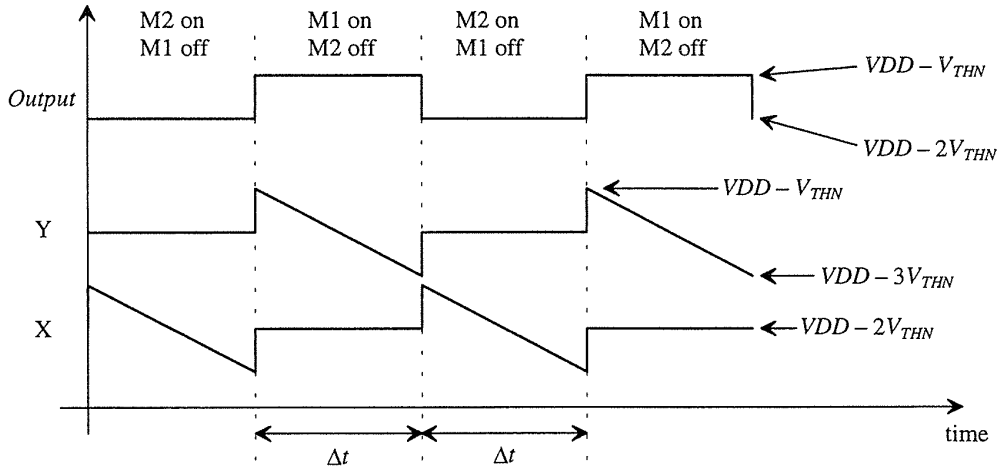



**Figure 19.19** Voltage waveforms for the NMOS source coupled VCO.

**Example 19.2**

Design an NMOS source coupled VCO with a center frequency of 1 MHz. Assume $I_D$ = 10 μA at $V_{inVCO}$ of 2.5 V. (Design of current sources will be discussed in the next chapter.) Simulate the design using SPICE, showing the output of the VCO and points X and Y.

Assuming $V_{THN}$ is approximately 1 V (this approximation is based on M1 and M2's threshold voltage), the value of the capacitor is determined using Eq. (19.29), or

$$C = \frac{10\ \mu A}{4 \cdot 1\ MHz} = 2.5\ pF$$

Since M1 and M2 are used as switches, we want to make their βs large. For the present design we picked *W/L* = 30/2. MOSFETs M3 and M4 were sized for a

Microchip Ex. 1055, p. 415
Microchip v. Aptiv
IPR2024-00494

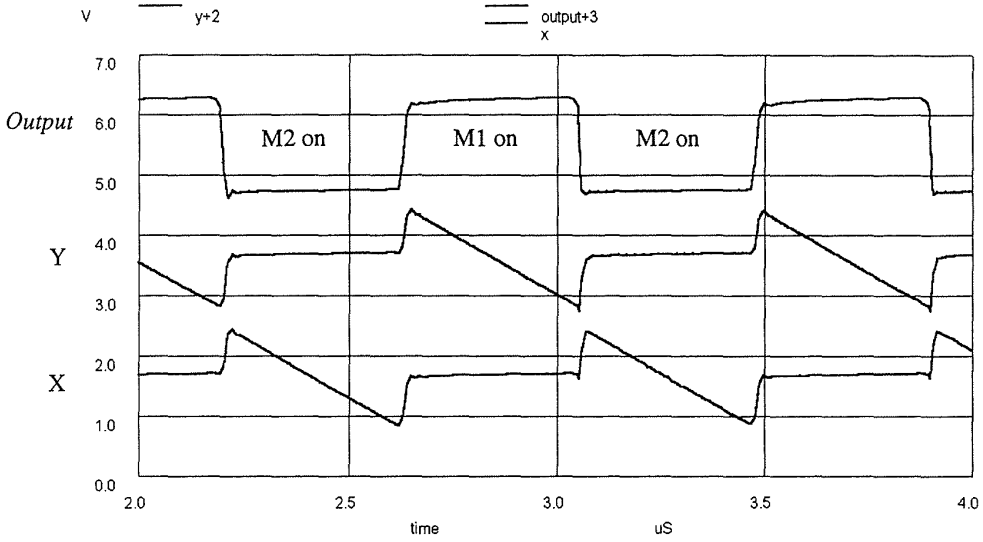



**Figure 19.20** SPICE simulation results for Ex. 19.2.

relatively large resistance by setting $W/L$ = 3/8. The simulation results are shown in Fig. 19.20. A practical limit on the oscillator frequency using the CN20 process is in the tens of MHz. ■

Another example of a VCO that is particularly well suited to both high and low frequency operation is shown in Fig 19.21 [4]. The magnitudes of the current sources

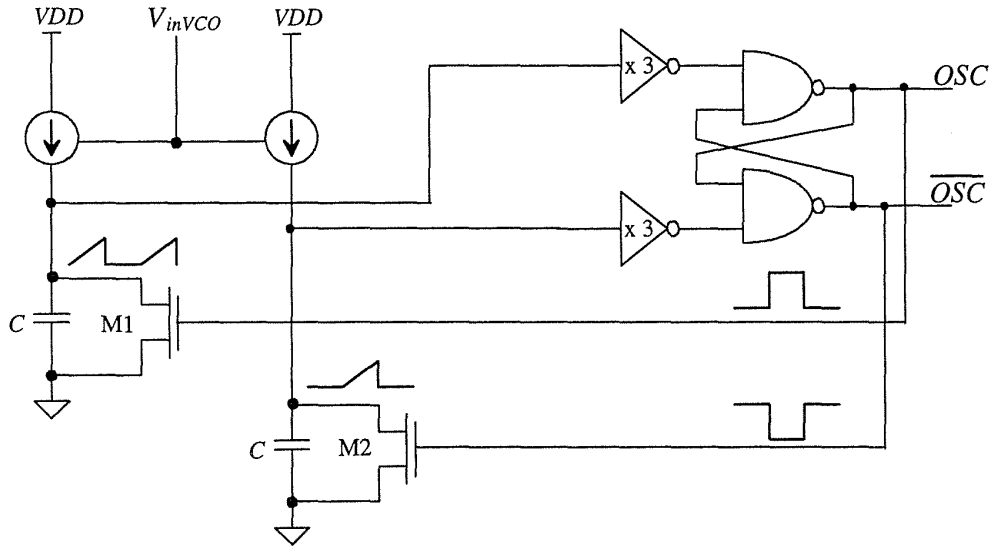



**Figure 19.21** VCO with wide operating range (requires two capacitors).

Microchip Ex. 1055, p. 416
Microchip v. Aptiv
IPR2024-00494

are controlled by an external voltage. The positive feedback around the loop is used to ensure that only one of M1 and M2 are on at a time. The switching points of the inverters combined with the current source determine the oscillator frequency.

## 19.3 The Loop Filter

The loop filter is the brain of the DPLL. In this section, we discuss how to select the loop-filter values in order to keep the DPLL from oscillating (i.e., keep the $V_{inVCO}$ voltage from oscillating, causing the frequency out of the VCO to wander). If the loop-filter values are not selected correctly, it may take the loop too long to lock, or once locked small variations in the input data may cause the loop to unlock.

In the following discussion, we will be concerned with the *pull-in range* and the *lock range*. The pull-in range, $\pm\Delta\omega_P$, is defined as the range of input frequencies which the DPLL will lock to. The time it takes the loop to lock is labeled $T_P$ and may be a very long time. If the center frequency of the DPLL is 10 MHz and the pull-in range is 1 MHz, the DPLL will lock on an input frequency from 9 to 11 MHz in a time $T_P$ (assuming $N = 1$). The lock range, $\pm\omega_L$, is the range of frequencies in which the DPLL locks within one single beat note between of the divided down output (*dclock*) and input (*data*) of the DPLL. *The operating frequency of the DPLL should be limited to the lock range for normal operation.* Once the DPLL is locked, it will remain locked as long as abrupt frequency changes, $\Delta\omega$, in the input frequency (input frequency steps) over a time interval $t$ are much smaller than the natural frequency of the system squared, that is, $\Delta\omega/t < \omega_n^2$.

### 19.3.1 XOR DPLL

Consider the block diagram of the DPLL using the XOR DPLL shown in Fig. 19.22. The phase transfer function is given by

$$H(s) = \frac{\phi_{clock}}{\phi_{data}} = \frac{K_{PD}K_FK_{VCO}}{s + \beta \cdot K_{PD}K_FK_{VCO}} \tag{19.30}$$

with $s = j\omega$ and the feedback factor, $\beta$, is

$$\beta = \frac{1}{N} \tag{19.31}$$

The transfer function of the loop filter is given by

$$K_F = \frac{1}{1 + j\omega RC} = \frac{1}{1 + sRC} \tag{19.32}$$

Substituting Eqs. (19.31) and (19.32) into Eq. 19.30 yields

$$H(s) = \frac{\phi_{clock}}{\phi_{data}} = \frac{K_{PD}K_{VCO} \cdot \frac{1}{1+sRC}}{s + \frac{1}{N} \cdot K_{PD}K_{VCO} \cdot \frac{1}{1+sRC}} \tag{19.33}$$

This is a second-order system. $H(s)$ can be rewritten as

Microchip Ex. 1055, p. 417
Microchip v. Aptiv
IPR2024-00494

$$H(s) = \frac{\frac{K_{PD}K_{VCO}}{RC}}{s^2 + \frac{s}{RC} + \frac{1}{N} \cdot \frac{K_{PD}K_{VCO}}{RC}} = \frac{f_{clock}}{f_{data}} = \frac{(K_{VCO}V_{inVCO})/2\pi + f_o}{f_{data}} \tag{19.34}$$

since $j\omega \cdot f = \phi$. The natural frequency, $\omega_n$, of this system is given by

$$\omega_n = \sqrt{\frac{K_{PD}K_{VCO}}{N \cdot RC}} \tag{19.35}$$

and the damping ratio, $\zeta$, is given by

$$\zeta = \frac{1}{2RC\omega_n} = \frac{1}{2} \cdot \sqrt{\frac{N}{K_{PD}K_{VCO} \cdot RC}} \tag{19.36}$$

The pull-in range is given by

$$\Delta\omega_P = \frac{\pi}{2} \cdot \sqrt{2\zeta\omega_n K_{VCO}K_{PD} - \omega_n^2} \tag{19.37}$$

while the pull-in time is given by

$$T_P = \frac{4}{\pi^2} \cdot \frac{\Delta\omega_{center}^2}{\zeta\omega_n^3} \tag{19.38}$$

The lock range of the loop is given by

$$\Delta\omega_L = \pi\zeta\omega_n = \frac{\pi}{2} \cdot \frac{1}{RC} \tag{19.39}$$

while the lock time is

$$T_L = \frac{2\pi}{\omega_n} \tag{19.40}$$

Probably the best way to understand how these equations affect the performance of a DPLL is through an example.

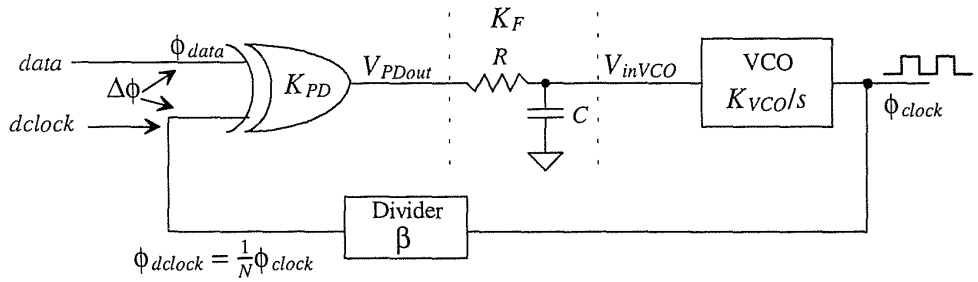



**Figure 19.22** Block diagram of a DPLL using a XOR phase detector

**Example 19.3**

The DPLL of Fig. 19.22 is used to generate a clock for a 9 Mbits/s data stream with data format (non-return to zero level, or simply NRZ-level) shown in Fig.

Microchip Ex. 1055, p. 418
Microchip v. Aptiv
IPR2024-00494

19.7. Assume that there is a transition at least once every nine bits (an odd-parity bit), that the code for the eight-bit word 255 (11111111) is not allowed, and that some synchronization words are present in the data to help the DPLL lock[3]. Determine $R$ and C of the loop filter, and comment on your selection of VCO.

The width of one bit of *data* is 111 ns (= 1/9 MHz). The frequency of the DPLL output *clock* is 9 MHz. A divide by 2 ($N = 2$) stage is used in the feedback loop to make *dclock* = 4.5 MHz. If the *data* input is an alternating series of ones and zeros as shown in Fig. 19.23, the frequency of the resulting square wave is 4.5 MHz, or one-half of the data rate. This is the major advantage of using NRZ code; that is, the data rate can be twice the channel bandwidth.

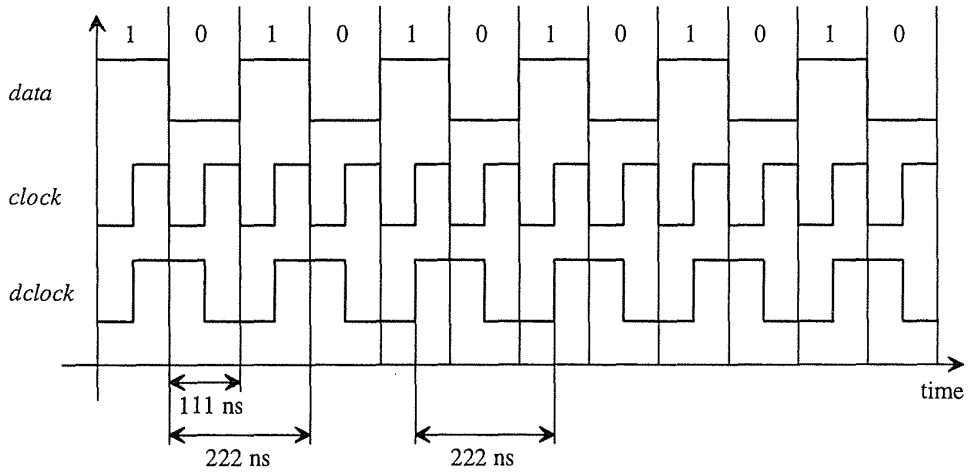



**Figure 19.23** Example waveforms for the DPLL of Example 19.3 in lock.

Next let's discuss the VCO. From the previous discussion of Sec. 19.1.1, the maximum $f_{clock}$ can be is 18 MHz and the minimum is 4.5 MHz. When the control voltage is *VDD*/2, the $f_{center}$ of the VCO is 9 MHz. (This is important!) If we try to use the basic current-starved VCO of the previous section, we run into the problem of locking on harmonics of the desired clock frequency. A simple method of setting the maximum current in this VCO, and thus the maximum frequency, is to place a current limiting resistor in series with the drains of M5 and M6 (see Fig. 19.24). The current that flows in M5 and M6 is mirrored by M8 and M9 and sets the variable current in the VCO. To set the minimum current, we add MOSFET M7 to provide a minimum current through M6 when $V_{inVCO} < V_{THN}$. (Note that this current is always present and added to the voltage-variable current through M6.) In all practical situations, the VCO

[3] System considerations concerning selection of a data pattern as well as regenerating the clock from NRZ data will be discussed in greater detail later in the chapter.

Microchip Ex. 1055, p. 419
Microchip v. Aptiv
IPR2024-00494

must be hand tailored to operate over the desired frequency range. Here in itself is the main problem using the XOR-based DPLL in a pure monolithic form. Bringing the frequency setting components off chip can help solve this problem. Precision components can be used to set the operating frequencies. One last concern when designing the VCO, or any CMOS circuit for that matter, is its temperature behavior. This subject is covered in more detail in the coming chapters.

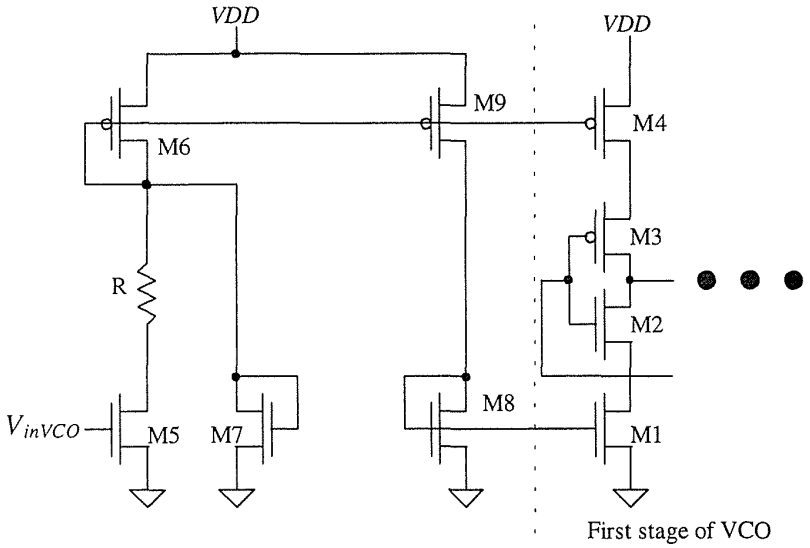



**Figure 19.24** Modifications to the current-starved VCO to set minimum and and maximum frequencies.

If we design the VCO so that when $V_{inVCO} = 4\text{V} = V_{max}$, the output oscillator frequency, $f_{max}$ is 9.1 MHz; when $V_{inVCO} = 1\text{V} = V_{min}$, the output frequency, $f_{min}$ is 8.9 MHz then the gain of the oscillator, $K_{VCO}$, is $2\pi \cdot 67$ kHz/V, or

$$K_{VCO} = 418 \times 10^3 \text{ radians/V·s}$$

The only component left in the first-cut design of the loop is selection of $RC$. Let's begin by setting the damping ratio $\zeta$ to 0.7. Using Eq. (19.36) results in

$$0.7 = \frac{1}{2} \cdot \sqrt{\frac{2}{\frac{5}{\pi} \cdot (418 \times 10^3) RC}} \rightarrow RC = 1.5 \text{ μs}$$

The natural frequency of the second-order system is, from Eq. (19.36),

$$\omega_n = \frac{1}{2 \cdot RC \cdot \zeta} = 476 \times 10^3 \text{ radians/s}$$

The lock range is

$$\Delta\omega_L = \pi\zeta\omega_n = 1 \times 10^6 \text{ radians/s or } \Delta f_L = 167 \text{ kHz}$$

Microchip Ex. 1055, p. 420
Microchip v. Aptiv
IPR2024-00494

which is outside the VCO operating range. Therefore, the DPLL will lock over the entire VCO operating range, that is, from 8.9 to 9.1 Mbits/s. We do not need to calculate the pull-in range since it is greater than the lock-in range. Again, the design of the oscillator is important in this DPLL. The lock time, $T_L$, of the DPLL is approximately 13 μs. A simulation of this DPLL using the software provided with [1] is shown in Fig. 19.25. Initially, at $t < 0$, the data rate input is 9.0 Mbits/s, and both the average phase-detector output and the output of the filter are 2.5 V ($VDD/2$ and $\Delta\phi = \pi/2$), point A in Fig. 19.25. At $t = 0$, the data rate jumps to 9.04 Mbits/s. Both the average phase-detector and filter outputs increase in voltage to 3.1 V. The *dclock* is now aligned slightly off center of the *data*, (3.1 $V/K_{PD} = \Delta\phi$ or $\Delta\phi = 0.62 \cdot \pi$), keeping the loop locked. ■

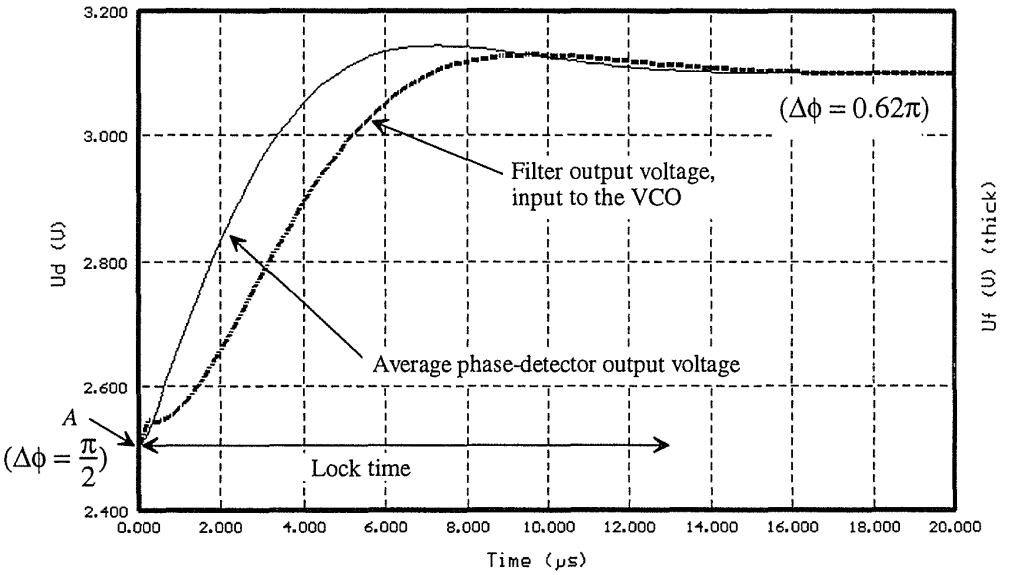



**Figure 19.25** Simulation of the DPLL of Ex. 19.3 with a 40 kbit/s data rate step. This corresponds to a frequency step of 20 kHz.

The stringent requirements of the VCO used in a DPLL with a XOR phase detector can be greatly relaxed by using the active proportional + integral (active PI) loop filter shown in Fig. 19.26. The transfer function of this filter is given by

$$K_F = \frac{1 + sR_2C}{sR_1C} \tag{19.41}$$

The natural frequency of the resulting second-order system is given by

$$\omega_n = \sqrt{\frac{K_{PD}K_{VCO}}{NR_1C}} \tag{19.42}$$

and the damping ratio is given by

Microchip Ex. 1055, p. 421
Microchip v. Aptiv
IPR2024-00494

$$\zeta = \frac{\omega_n R_2 C}{2} \tag{19.43}$$

The lock range is extended to

$$\Delta\omega_L = 4\pi\zeta\omega_n \tag{19.44}$$

while the lock time remains $2\pi/\omega_n$. The pull-in range, using the active PI loop filter, is limited by the VCO oscillator frequency. Consider the following example.

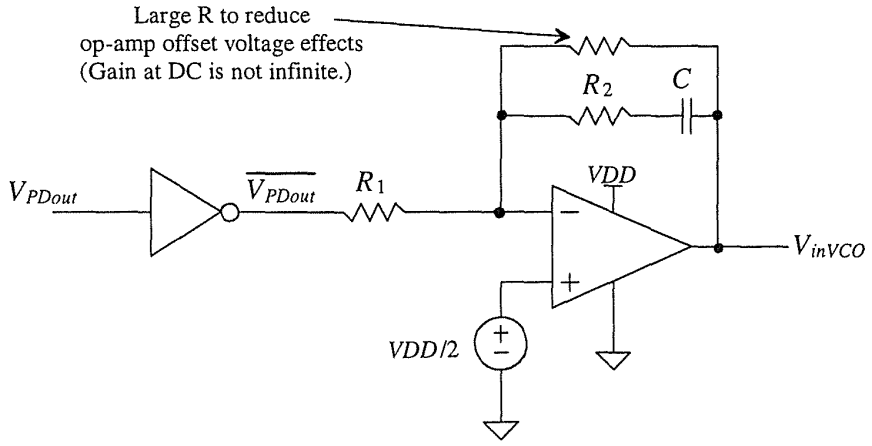



**Figure 19.26** Active PI loop filter.

**Example 19.4**
Repeat Ex. 19.3 using the active PI loop filter.

The gain of the VCO can be increased using the active PI loop filter. If the output of the VCO is 10 MHz ($=f_{max}$), when the input voltage is 4 V and 8 MHz ($=f_{min}$) when the input to the VCO is 1 V, then the gain of the VCO is

$$K_{VCO} = 2\pi \cdot \frac{10-8}{4-1} = 4.2\times 10^6 \text{ radians/V·s}$$

which is much easier to realize in CMOS. If the lock range between *data* and *dclock* is set so that the VCO limits the operating frequency range, say $\Delta f_L$ = 1 MHz, then the natural frequency using Eq. (19.44) and assuming $\zeta$ = 0.7 is

$$\omega_n = \frac{2\pi \cdot 1\times 10^6}{4\pi \cdot 0.7} = 713\times 10^3 \text{ radians/s}$$

Using Eqs. (19.42) and (19.43) with $N$ = 2, we can solve for $R_1C$ and $R_2C$ as approximately 6.6 μs and 2 μs, respectively. The lock time of this loop is now 8.8 μs. Simulation results showing a 40 kbit/s data rate step are shown in Fig. 19.27. Figure 19.28 shows how a 500 kbit/s abrupt change in the data rate is

Microchip Ex. 1055, p. 422
Microchip v. Aptiv
IPR2024-00494

tracked by the DPLL. This corresponds to the data changing from 9 Mbits/s to 9.5 Mbits/s. The DPLL of Ex. 19.3 cannot lock to data input at 9.5 Mbits/s. In summary, the benefits of using an active PI loop filter over the simple RC are: relaxed requirements for the VCO (the gain of the VCO can be larger), wider operating frequency range, and faster lock time. ■

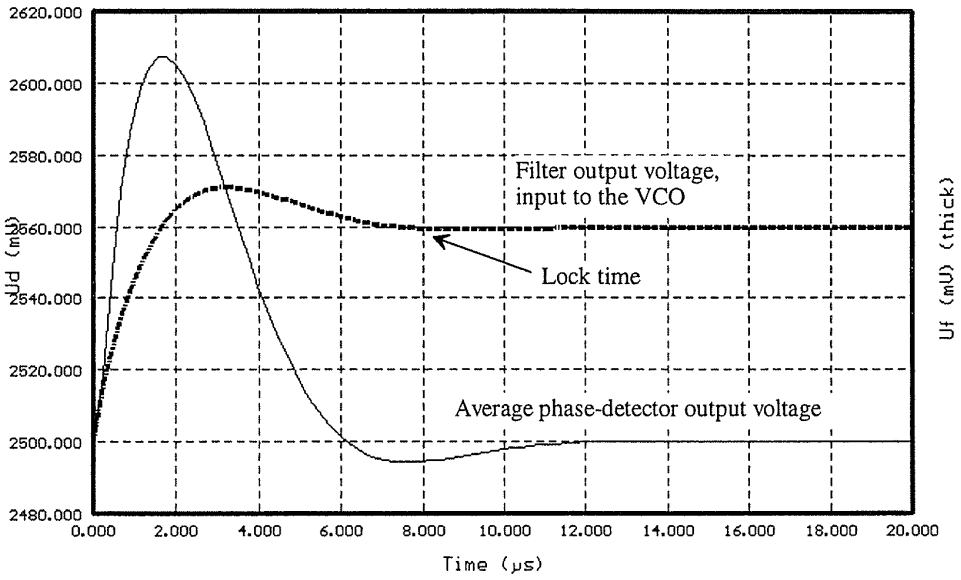



**Figure 19.27** Simulation of the DPLL of Ex. 19.4 with a 40 kbit/s data rate step. Note that a 40 kbit/s step corresponds to an input frequency step of 20 kHz.

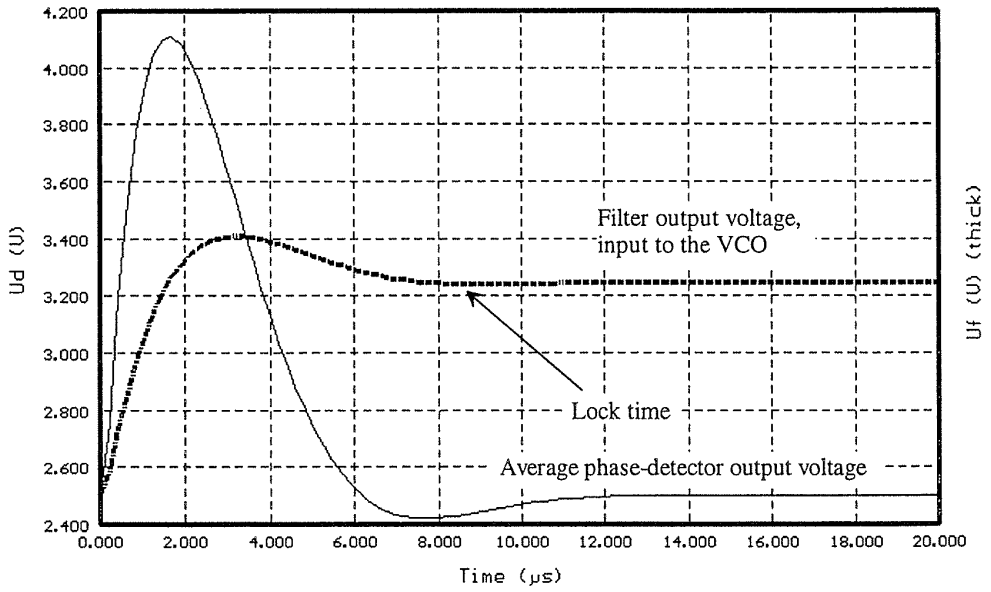



**Figure 19.28** Simulation of the DPLL of Ex. 19.4 with a 500 kbit/s input data rate step.

Microchip Ex. 1055, p. 423
Microchip v. Aptiv
IPR2024-00494

### 19.3.2 PFD DPLL

A block diagram of a DPLL using the PFD with tri-state output is shown in Fig. 19.29. The phase transfer function is the same as Eq. (19.30)

$$H(s) = \frac{\phi_{clock}}{\phi_{data}} = \frac{K_{PDtri} K_F K_{VCO}}{s + \beta \cdot K_{PDtri} K_F K_{VCO}} \tag{19.45}$$

where

$$K_F = \frac{1 + sR_2C}{1 + s(R_1 + R_2)C} \tag{19.46}$$

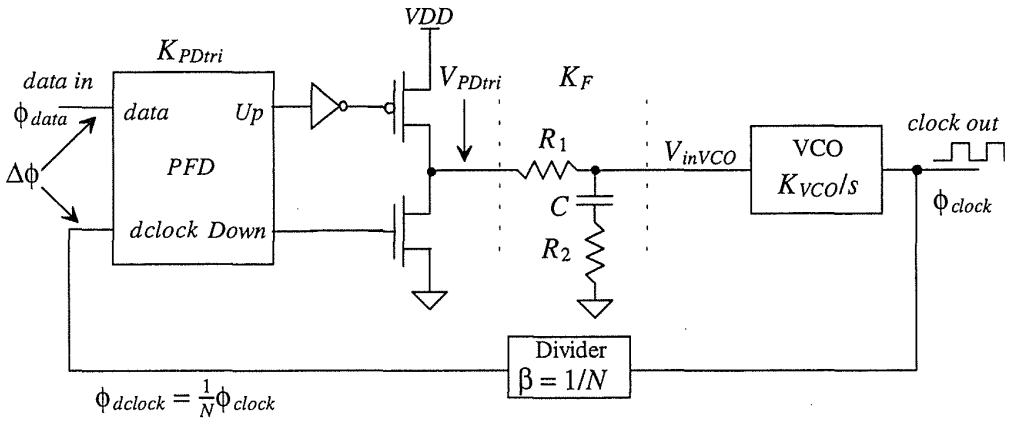



**Figure 19.29** Block diagram of a DPLL using a sequential phase detector (PFD).

When this filter is driven with the tri-state output, no current flows in $R_1$ or $R_2$ when the output is in the high-impedance state. The voltage across the capacitor remains unchanged. We can think of the filter, tri-state output as an ideal integrator with a transfer function

$$K_F' = \frac{1 + sR_2C}{s(R_1 + R_2)C} \tag{19.47}$$

Substituting this equation into Eq. 19.45 and rearranging results in

$$H(s) = \frac{K_{PDtri} K_{VCO} \frac{1+sR_2C}{(R_1+R_2)C}}{s^2 + s\frac{K_{PDtri}K_{VCO}R_2C}{N(R_1+R_2)C} + \frac{K_{PDtri}K_{VCO}}{N(R_1+R_2)C}} = \frac{\phi_{clock}}{\phi_{data}} = \frac{f_{clock}}{f_{data}} \tag{19.48}$$

From this equation the natural frequency is

$$\omega_n = \sqrt{\frac{K_{PDtri} K_{VCO}}{N(R_1 + R_2)C}} \tag{19.49}$$

and the damping factor is determined by solving

Microchip Ex. 1055, p. 424
Microchip v. Aptiv
IPR2024-00494

$$2\zeta\omega_n = \frac{K_{PDtri}K_{VCO}R_2C}{N(R_1+R_2)C} \tag{19.50}$$

which results in a damping factor given by

$$\zeta = \frac{\omega_n}{2}\cdot R_2C \tag{19.51}$$

The lock range is given by

$$\Delta\omega_L = 4\pi\zeta\omega_n \tag{19.52}$$

while the lock time, $T_L$, remains $4\pi\omega_n$. The pull-in range is limited by the VCO operating frequency. The pull-in time is given by

$$T_P = 2R_1C\cdot\ln\frac{(K_{VCO}/N)\cdot(VDD/2)}{(K_{VCO}/N)(VDD/2)-\Delta\omega} \tag{19.53}$$

where $\Delta\omega$ is the magnitude of the *input* frequency step.

**Example 19.5**

Design a DPLL that generates a clock signal at a frequency of 256 kHz from a 1-kHz square wave input. This application of the DPLL, is called *frequency synthesis* (Fig. Ex19.5).

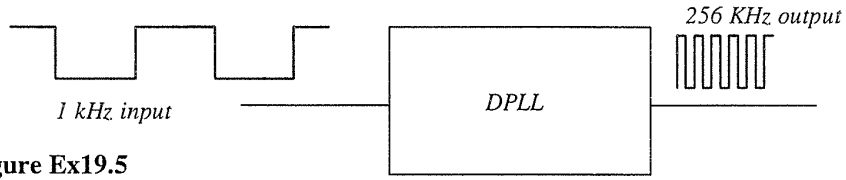



**Figure Ex19.5**

We will use the DPLL of Fig. 19.29 with a divide by 256 ($N$ = 256) in the feedback loop. Furthermore, assume that the VCO oscillates from 500 kHz down to DC for input voltages ($V_{inVCO}$) from 0 to 5 V. The gain of the VCO is then

$$K_{VCO} = 2\pi\cdot\frac{(500-0)\ KHz}{5-0} = 628\times10^3 \text{ radians/V·s}$$

The lock range, $\Delta f_L$ will be set to 200 Hz. This means the loop will lock up on an input frequency from 800 to 1200 Hz and generate an output frequency 256 times the input frequency in a time $T_L$. For example, if 900 Hz is input into the loop, 230 kHz will be output by the DPLL. The natural frequency of the DPLL, again assuming $\zeta$ = 0.7, is

$$\omega_n = \frac{2\pi\cdot200}{4\pi\cdot0.7} = 143 \text{ radians/s}$$

Using Eqs. (19.51) and (19.49) gives the loop-filter time constants of approximately, $R_2C$ = 9.8 ms and $R_1C$ = 38 ms ($K_{PDtri} = VDD/4\pi$). In most cases,

Microchip Ex. 1055, p. 425
Microchip v. Aptiv
IPR2024-00494

the loop filter would be implemented off-chip. The lock time, $T_L$, for this DPLL is 44 ms. Figure 19.30 shows how the average PD and loop-filter outputs change with a 200 Hz input step in frequency. Initially, at $t < 0$, the input voltage to the VCO is 2.5 V and the input frequency is 1,000 Hz, while the output of the VCO is 256,000 Hz ($N$·1,000 Hz). For times greater than zero, the input frequency jumps to 1,200 Hz, causing the VCO output frequency to jump to 307,000 Hz. Approximately 44 ms after the initial frequency jump, the input voltage to the VCO, $V_{inVCO}$, settles at approximately 3 V.

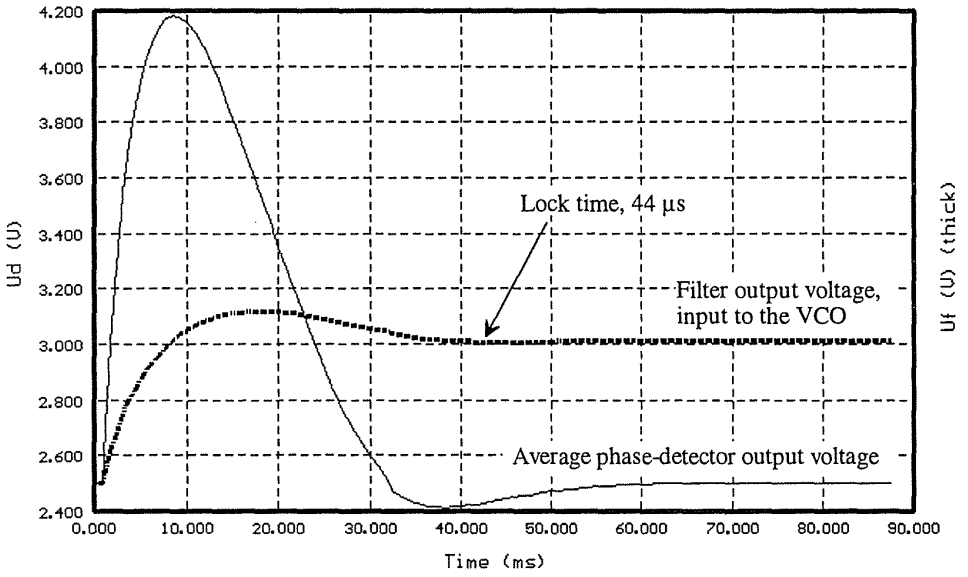



**Figure 19.30** Simulation of the DPLL of Ex. 19.5 with an input frequency step of 200 Hz

The maximum VCO output frequency is 500 kHz. This corresponds to an input frequency of 1.95 kHz (divide the VCO output frequency by 256). An 800 Hz frequency step in the input to the DPLL (change the input frequency from 1,000 to 1,800 Hz) results in the waveforms shown in Fig. 19.31. Since the frequency step lies outside the lock range of the DPLL, the VCO input voltage will oscillate before finally locking. The time it takes the loop to pull-in the VCO control voltage, and relock is approximately $T_P$, or

$$T_P = 2R_1C \cdot \ln \frac{(628 \times 10^3/256) \cdot (2.5)}{(628 \times 10^3/256)(2.5) - 2\pi \cdot 800} = 130 \text{ ms}$$

Once the loop is locked, it will not lose lock unless the change in the input frequency, $\Delta\omega$, over a time $t$ is greater than the $\omega^2_n$. The relaxed requirements on the VCO operating range make the PFD the phase detector of choice when designing CMOS DPLLs. ■

Microchip Ex. 1055, p. 426
Microchip v. Aptiv
IPR2024-00494

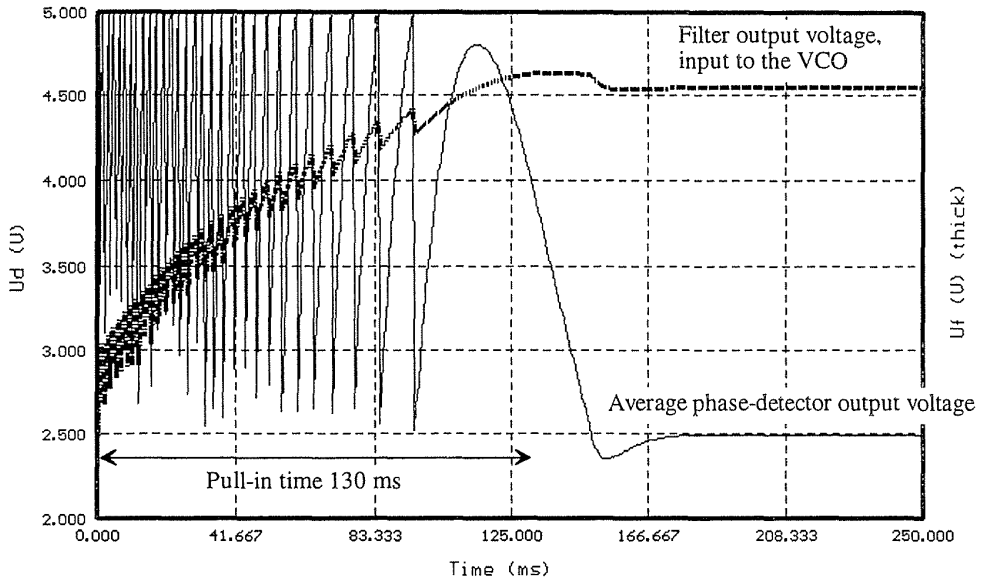



**Figure 19.31** Simulation of the DPLL of Ex. 19.5 with an input frequency step of 800 Hz.

The PFD with a charge-pump output is shown in Fig. 19.32. A CMOS implementation of a DPLL using this configuration is, in general, preferred over the tri-state output because of the better immunity to power supply variations. The capacitor $C_2$ is used to keep $I_{pump} \cdot R$ from causing voltage jumps on the input of the VCO and thus frequency jumps in the DPLL output from occurring. In general, $C_2$ is set at

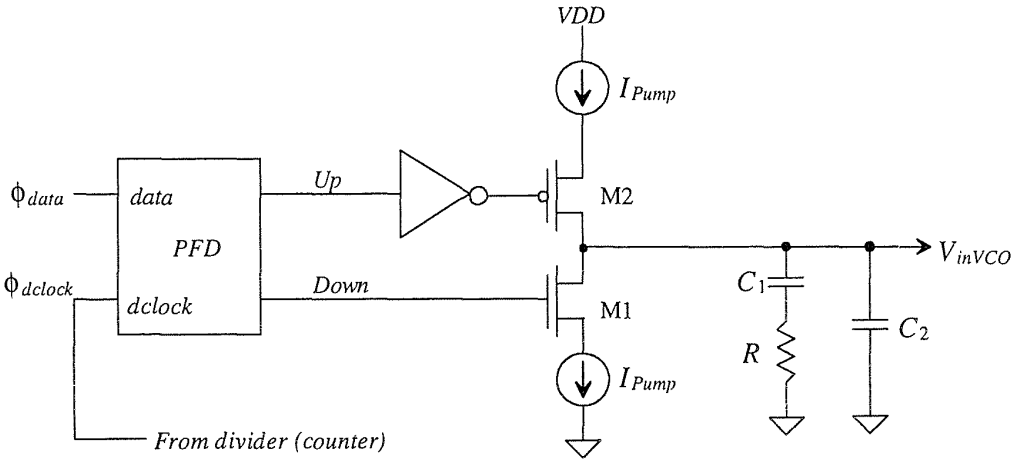



**Figure 19.32** PFD using the charge pump.

Microchip Ex. 1055, p. 427
Microchip v. Aptiv
IPR2024-00494

about one-tenth (or less) of $C_1$. The loop-filter transfer function neglecting $C_2$ is given by

$$K_F = \frac{1 + sRC_1}{sC_1} \tag{19.54}$$

The feedback loop transfer function is given by

$$H(s) = \frac{\phi_{clock}}{\phi_{data}} = \frac{K_{PDI}K_{VCO}(1 + sRC_1)}{s^2 + s(\frac{K_{PDI}K_{VCO} \cdot R}{N}) + \frac{K_{PDI}K_{VCO}}{NC_1}} \tag{19.55}$$

From the transfer function the natural frequency is given by

$$\omega_n = \sqrt{\frac{K_{PDI}K_{VCO}}{NC_1}} \tag{19.56}$$

and the damping factor is

$$\zeta = \frac{\omega_n}{2} \cdot RC_1 \tag{19.57}$$

The lock range and lock time remain the same (using the different values for the natural frequency and damping ratio) as the PFD with the tri-state output. Again, the pull-in range is set by the VCO oscillator frequency range. The pull-in time is given by

$$T_P = 2RC_1 \ln\left[\frac{(K_{VCO}/N) \cdot (I_{pump})}{(K_{VCO}/N) \cdot (I_{pump}) - \Delta\omega}\right] \tag{19.58}$$

*Discussion*

When selecting values for the loop filter, we assumed that the output resistance of the phase detector was small (for the XOR and tri-state PD) compared to the impedances used in the loop filter. We also assumed that the input resistance of the VCO was infinite and that the input capacitance of the VCO was small compared to the capacitance used in the loop filter. Considering the parasitics present in the DPLL is an important part of the design.

Examples of CMOS VCOs were given in Sec. 19.2. An example of a CMOS implementation of a PFD is shown in Fig. 19.33[5]. Implementation of the loop filter can use MOSFETs as capacitors and n-well or MOSFETs as resistors.

The center frequency of the VCO was critical for good DPLL performance when using the XOR gate with *RC* loop filter. If the center frequency, $f_{center}$, of the VCO (i.e, $V_{inVCO}$ = *VDD*/2) does not match twice the input data rate (when using NRZ code), the DPLL will lock up at a phase different from $\pi/2$. The need for a precision center frequency is eliminated by using the XOR PD with an active PI loop filter or by using the PFD.

Microchip Ex. 1055, p. 428
Microchip v. Aptiv
IPR2024-00494

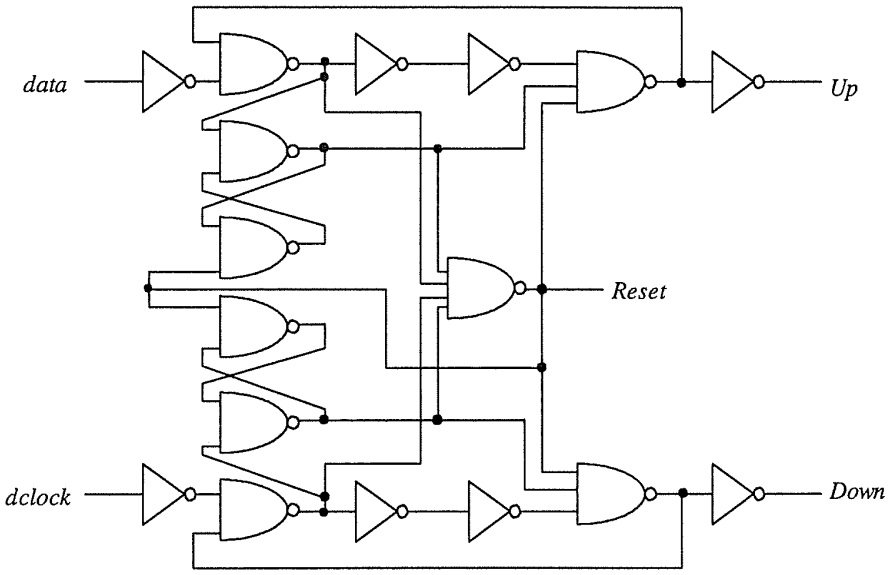



**Figure 19.33** CMOS implementation of a PFD.

## 19.4 System Considerations

System concerns are often the driving force behind the design of a DPLL. Referring to Fig. 19.1, we observe that the data transmitted through the channel should ideally arrive at the receiver with the same shape as it was transmitted with [6]. In reality, the data become distorted. Distortion arises from nonlinearities in the receiver input amplifier and the finite bandwidth of the channel. To understand the conditions for distortionless transmission, consider the block diagram shown in Fig. 19.34. The system has a transfer function in the frequency domain of $H(f)$ and in the time domain $h(t)$. For distortionless transmission, we can relate the input and output of the system by

$$y(t) = K \cdot x(t - t_o) \tag{19.59}$$

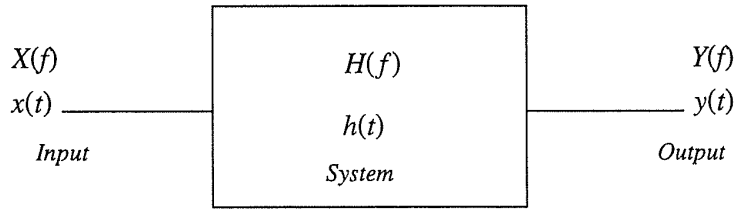



**Figure 19.34** Representation of a system with input and output.

Microchip Ex. 1055, p. 429
Microchip v. Aptiv
IPR2024-00494

where $t_o$ is the time delay through the system and $K$ is a constant. This equation shows that for distortionless transmission through a system the output is simply a scaled, time-delayed version of the input. An interesting observation can be made by taking the Fourier Transform of both sides of this equation,

$$Y(f) = K \cdot X(f)e^{-j2\pi ft_o} \tag{19.60}$$

The transfer function of a distortionless system can then be written as

$$H(f) = \frac{Y(f)}{X(f)} = Ke^{-j2\pi ft_o} \tag{19.61}$$

Figure 19.35 shows the magnitude and phase responses of a distortionless system. A system is distortionless when its amplitude response, $|H(f)|$, is a constant, $K$, and its phase response, $\angle H(f)$, is linear with a slope of $-2\pi t_0$ over all frequencies of interest.

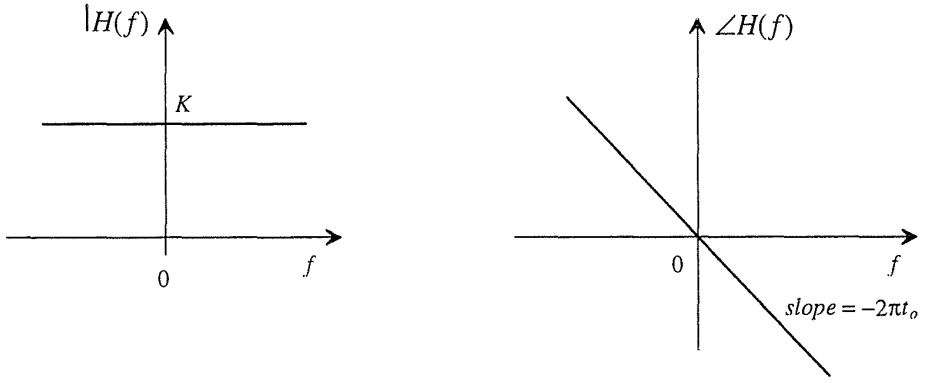



**Figure 19.35** Magnitude and phase response of a distortionless system.

The responses shown in Fig. 19.35 are ideal. In practice, the magnitude response of a system may look similar to Fig. 19.36a. At higher frequencies, the magnitude rolls off. To compensate for this roll-off, or other imperfections, a circuit called an equalizer is added in series with the system (Fig. 19.36b). The equalizer has a transfer function in which its magnitude response increases with increasing frequency beyond a point (Fig. 19.36c). If the low-frequency gain of the equalizer is $A/K$ and the low-frequency gain of the system is $K$, then the resulting gain of the system/equalizer combination is $A$.

Another source of potential distortion occurs when the receiver input data are regenerated into digital levels. Figure 19.37 shows the basic problem. The analog data generated by the receiver are connected to a comparator. The comparator slices the data, ideally through the middle, and regenerates the digital data. Timing errors occur when the comparator decision level is not correctly set at the middle of the data. Figure 19.37a shows how the analog data should be sliced. While Fig. 19.37b shows the resulting digital data with the decision level too high, the resulting ones are too narrow and the zeros are too wide. What makes slicing the data correctly even more difficult is

Microchip Ex. 1055, p. 430
Microchip v. Aptiv
IPR2024-00494

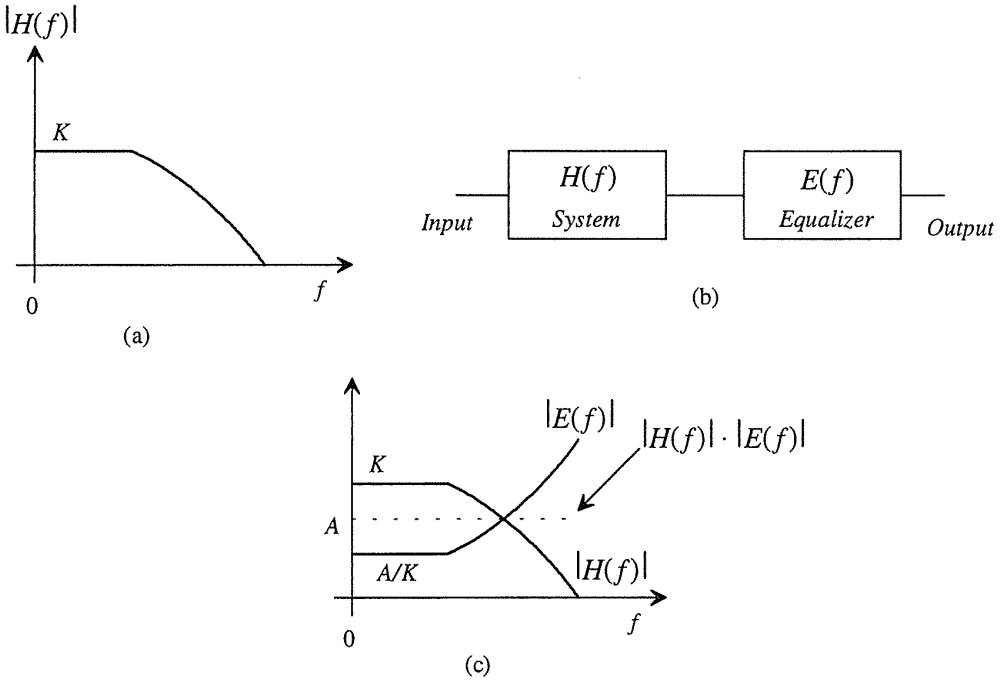



**Figure 19.36** Using an equalizer to lower distortion in a system.

the fact that the amplitude response of the channel can change with time and the data pattern can affect the average level of the data. Consider the data and decision-making circuit of Fig. 19.38. The comparator slices the input data around ground. The result, for the waveforms shown in Fig. 19.38, is output logic ones that are too wide (in time). A long string of ones causes the output zeros to be too wide. In both cases, timing errors are possible when clocking the data into the shift register (see Fig. 19.1).

There are two solutions to this problem. The first uses a circuit that will determine the peak positive and negative input analog amplitudes, average the values, and feed back the result to the comparator in the decision-making circuit. The second

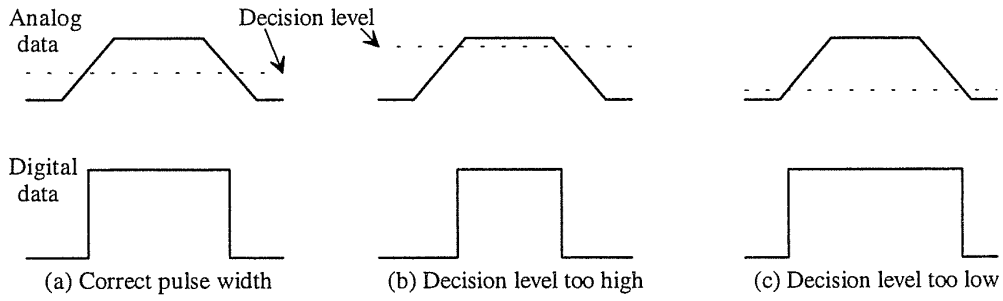



**Figure 19.37** Timing errors in regenerating digital data.

Microchip Ex. 1055, p. 431
Microchip v. Aptiv
IPR2024-00494

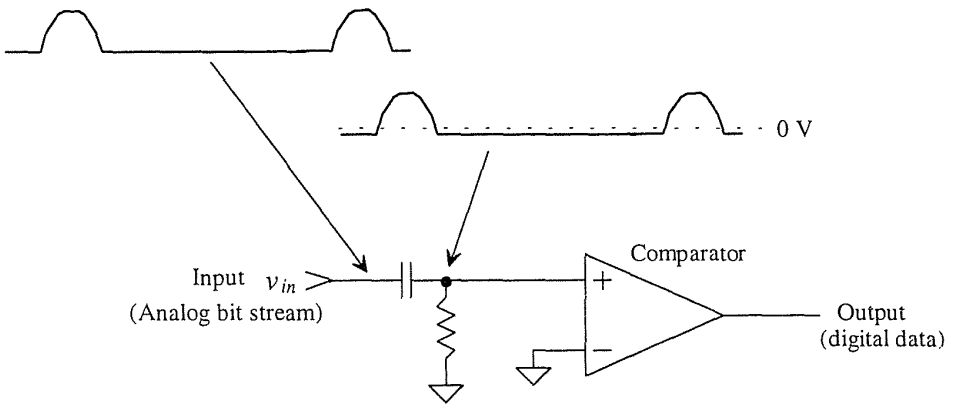



**Figure 19.38** The effect of data pattern on the decision-making circuit.

method encodes the digital data so that the duty cycle of the resulting encoded data is 50 percent. The encoding increases the channel bandwidth for a constant data rate.

Consider the peak detector circuit shown in Fig. 19.39. When the input, $v_{in}$, is greater than the output, $v_{peak}$, the comparator output goes high, turning the MOSFET on. This causes the capacitor to charge toward *VDD*. When $v_{peak}$ gets slightly larger than the input voltage, the comparator turns off. This in turn shuts the MOSFET off, leaving the capacitor essentially charged to the peak value of $v_{in}$.

The peak detector is used in the digital decision circuit shown in Fig. 19.40. Current sources, of a small value, are added to the basic decision circuit to make the peak detectors lossy. This may be necessary so that the decision circuit can follow the slow amplitude and data variations. The buffers shown in the circuit isolate the peak detectors from one another. These buffers can be implemented in some cases using the MOSFET source follower configuration (see Ch. 22). The two resistors perform the averaging function. The average of the analog input is feedback to the inverting input of the comparator. This scheme forces the data to be sliced in the middle, eliminating timing errors.

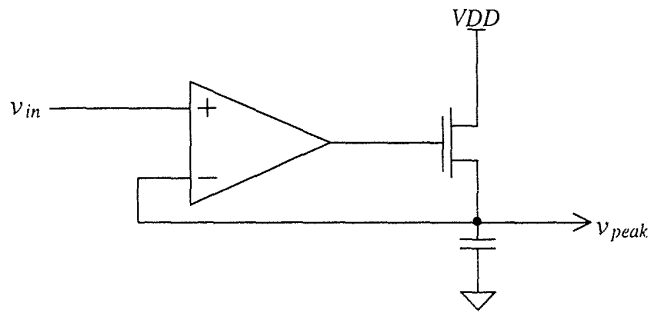



**Figure 19.39** CMOS peak detector.

Microchip Ex. 1055, p. 432
Microchip v. Aptiv
IPR2024-00494

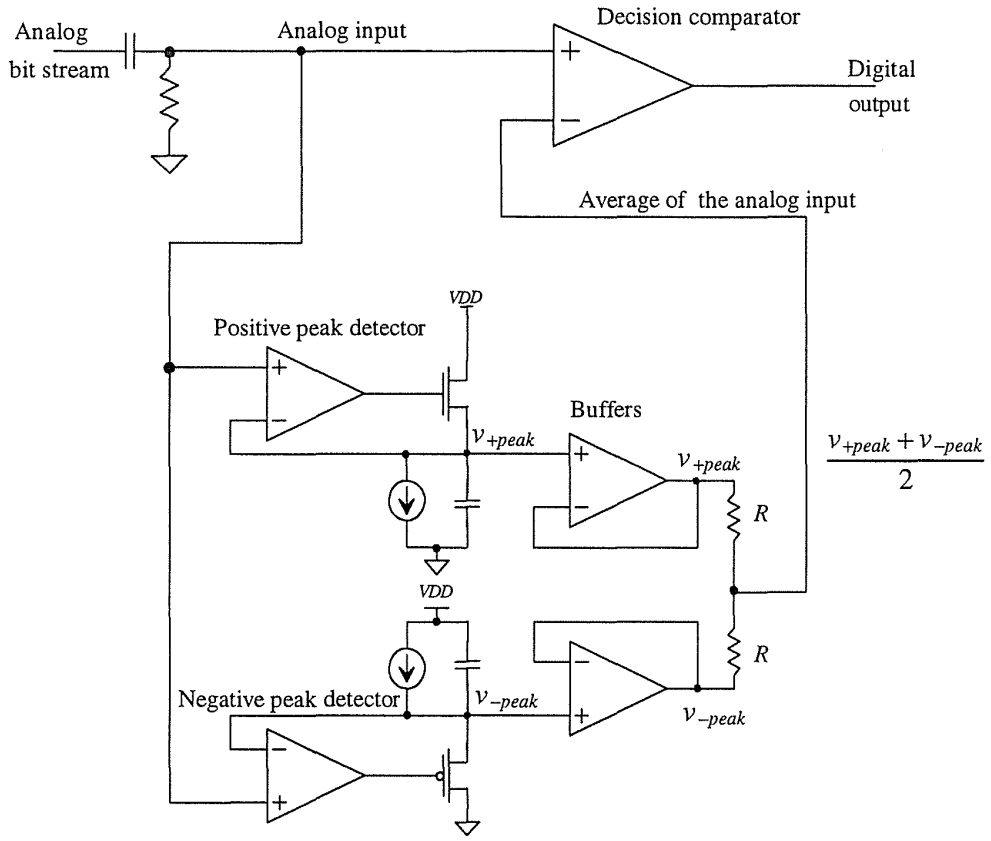



**Figure 19.40** Decision circuit.

Encoding the data can eliminate the need for a decision circuit. If the resulting encoded data has a 50 percent duty cycle, it can be passed through a capacitor in the receiver (Fig. 19.38), resulting in an analog signal centered around ground. The comparator can then have its noninverting input connected to ground and slice the data at the correct moments in time (in the middle of the data bit). An example of an encoding scheme is shown in Fig. 19.41. Encoding occurs in the transmitter prior to transmission over the channel. This particular encoding scheme is referred to as the bi-phase format, or more precisely, the bi-phase-level (sometimes called bi-phase-L or Manchester NRZ) format. The cost of using this scheme over the NRZ data format is increased channel bandwidth. Other encoding schemes are shown in Fig. 19.42 [1].

### 19.4.1 Clock Recovery from NRZ Data

One of the most important steps in the design of a communication system is selection of the transmission format, that is, NRZ, bi-phase, or some other format, together with use of parity, cyclic redundancy code, or some other encoding format.

Microchip Ex. 1055, p. 433
Microchip v. Aptiv
IPR2024-00494

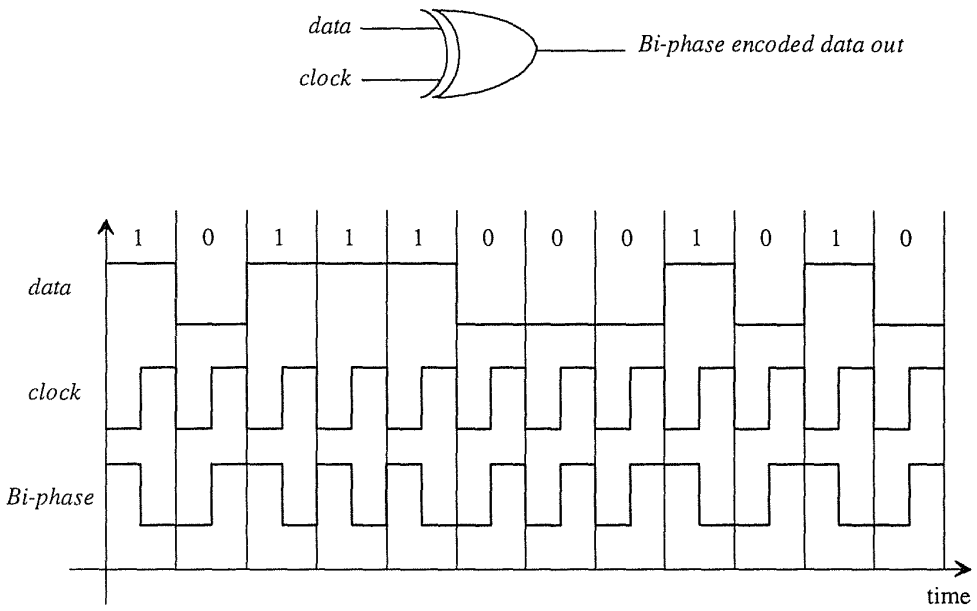



**Figure 19.41** Bi-phase data encoding.

In this section, we discuss some of the considerations that go into the design of a clock-recovery DPLL in a system that uses NRZ. The reasons for the selection of the formats used in Ex. 19.3 and the need for a divide by 2 in the feedback loop will be explained.

Let's begin this discussion by considering the NRZ *data* and *clock* shown in Fig. 19.43. Let's further assume that these signals are the inputs to an XOR PD in a DPLL, which is not in lock since the clock is not aligned properly to the data. The resulting output of the XOR PD is shown in this figure as well. If we were to average this output using a loop filter, we would get *VDD*/2. In fact, it is easy to show that shifting the *clock* signal in time has no effect on the average output of the PD. Why? To answer this question, let's use some numbers. Assume that a bit width of *data* is 10 ns (which is also the period of the *clock*). The frequency of the square wave resulting from the alternating strings of ones and zeros is 50 MHz. We know that if we take the Fourier Transform of a square-wave, only the odd harmonics (i.e., 50, 150, and 250 MHz) are present. Since the *clock* signal is at 100 MHz, there is no energy or information common between the *clock* and *data* signals. To remedy this, we divide the clock down in frequency, *dclock*, so that it is at the same frequency as the alternating ones and zeros of the data (divide by 2).

The next problem we encounter, if we use the divide by two in the feedback loop, occurs when we get *data* that is a repeating string of two-ones followed by two-zeroes, Fig. 19.44 (the *dclock* is not locked to the data). Again, there is no common information between the two inputs, and the resulting XOR PD output will always have

Microchip Ex. 1055, p. 434
Microchip v. Aptiv
IPR2024-00494

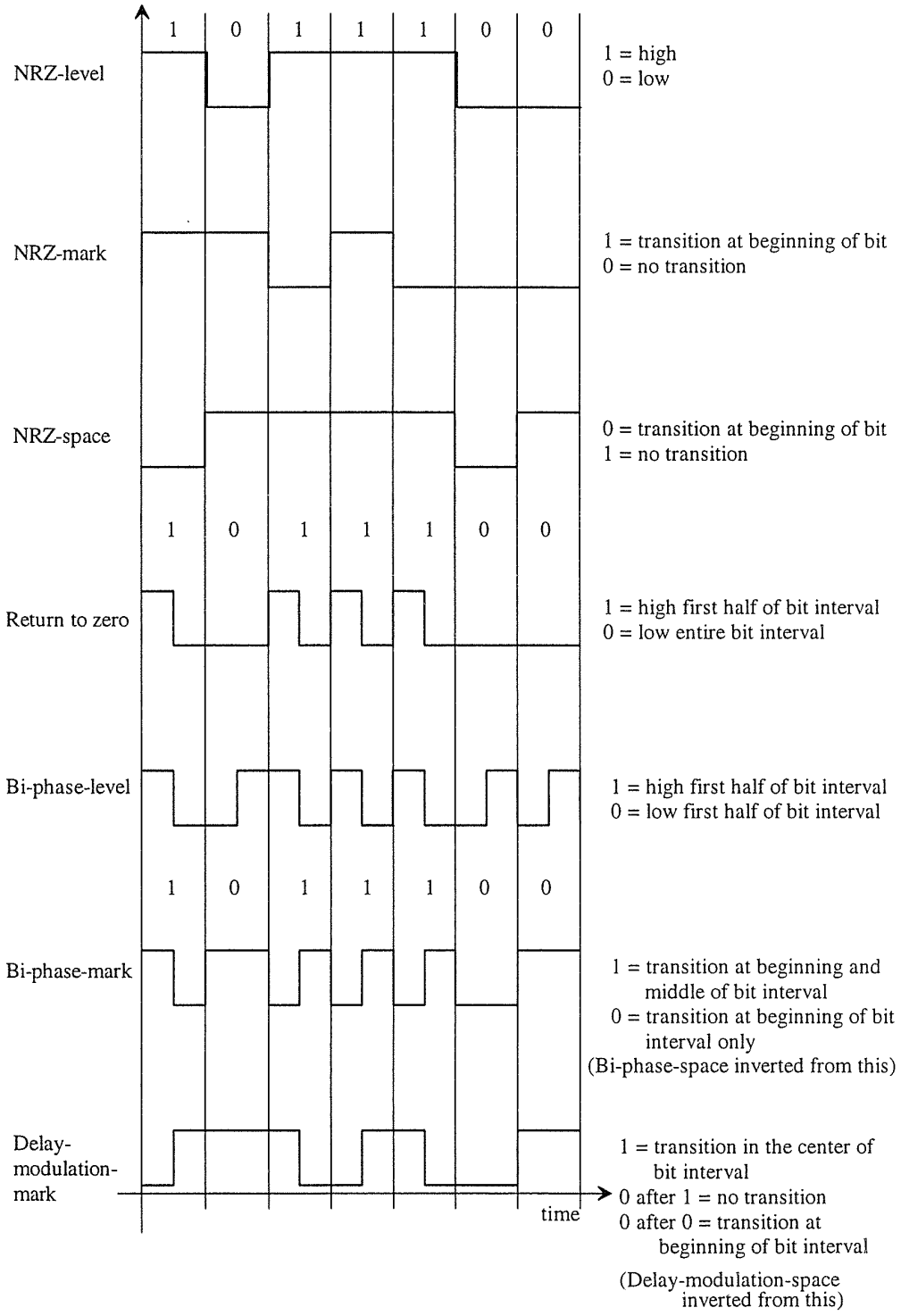



**Figure 19.42** Data transmission formats [1].

Microchip Ex. 1055, p. 435
Microchip v. Aptiv
IPR2024-00494

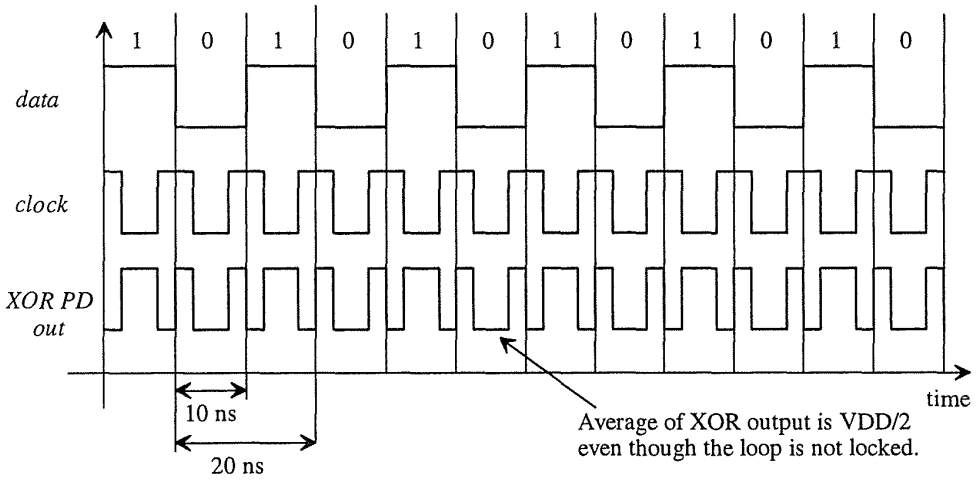



**Figure 19.43** The problems of using clock without the divide by 2 to lock on data.

an average of *VDD*/2. In this case, however, the *dclock* is running at 50 MHz and the *data* is a square-wave of 25 MHz. Should we divide the *clock* down further to avoid this situation? The answer is no. However many times we divide the clock down, we can still come up with a data string that will not allow the loop to lock. Also, it is the actual edge transitions (the frequency of the *data* and *dclock*) that is used when the inputs are not pure square-waves. Increasing the width of *dclock* has the effect of removing information and making it more difficult to lock up to the data. One solution to this problem, for an eight-bit word, was given in Ex. 19.3. If we use odd-parity with an eight-bit word (nine bits total) and eliminate, at the transmitting end, the possibility of all eight bits being high, that is, 11111111 or 255, then it is impossible to generate a square-wave.

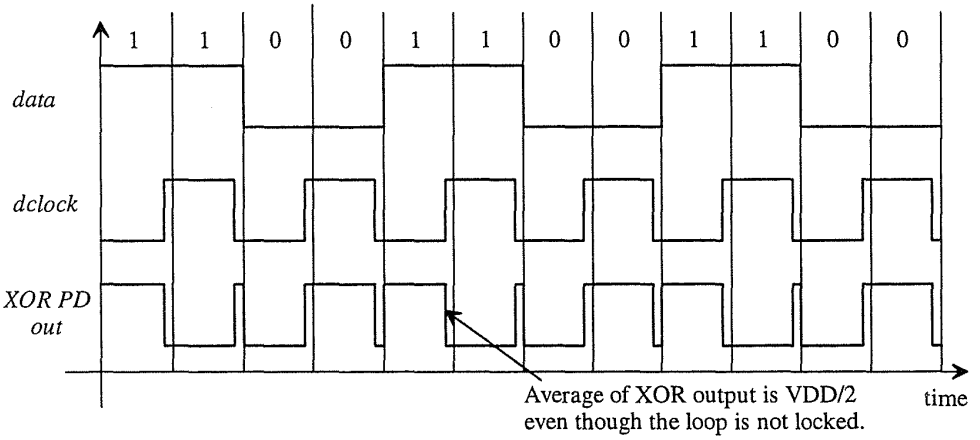



**Figure 19.44** Problems trying to lock on a data stream that is one-half the dclock frequency.

Microchip Ex. 1055, p. 436
Microchip v. Aptiv
IPR2024-00494

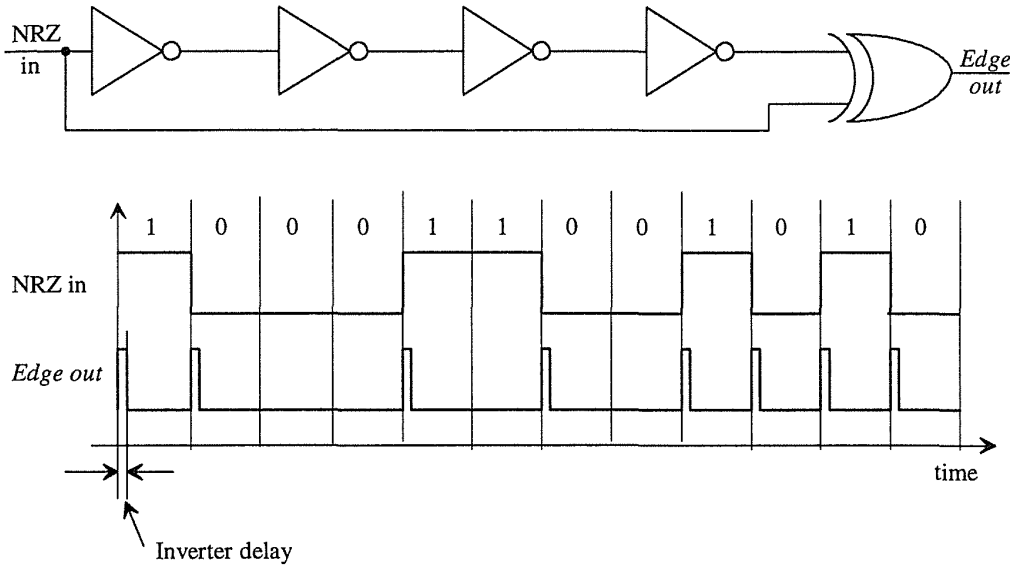



**Figure 19.45** Detecting the edges in NRZ data.

The restrictions on the data pattern in a communications system using NRZ data can be reduced by detecting the edges of the input data using an edge detector circuit (Fig. 19.45) [1]. The delay through the inverters sets the width of the output pulses, labeled "*Edge out*" in this circuit. The frequency content of the output pulses will always contain energy at the *clock* frequency and thus the loop can lock up on the data.

As an example, consider the block diagram and data shown in Fig. 19.46. The *Edge output* is connected as the *input* of an XOR based DPLL. The output of the VCO, *clock*, will lock up on the center of the *Edge output*, that is, the rising edge of the *clock* signal will become aligned with the center of *Edge out*. Averaging *PD out*, in this figure, results in *VDD*/2. If the clock is shifted to the left or right in time, the average value of *PD out* will shift down or up, causing the VCO frequency to change, keeping the *clock* aligned to the center of *Edge out*.

Several practical problems exist with this configuration. The delay through the inverters should be constant whether a high-to-low or a low-to-high transition is propagating through the inverters. Also, the delay of the inverters, or whatever element is used for the delay (one common element for high-speed applications is a microstrip line), should be close to one-half of the bit-interval time for best performance. For example, if the bit interval, for the data shown in Fig. 19.46, is 10 ns, then the delay used in the edge detector should be 5 ns. This delay is important since it directly affects the gain of the phase detector and therefore the transient properties of the DPLL.

Not having the *clock* aligned to the center of the data bit can cause problems in high-speed clock-recovery circuits. Simply adding a delay in series with the *clock*

Microchip Ex. 1055, p. 437
Microchip v. Aptiv
IPR2024-00494

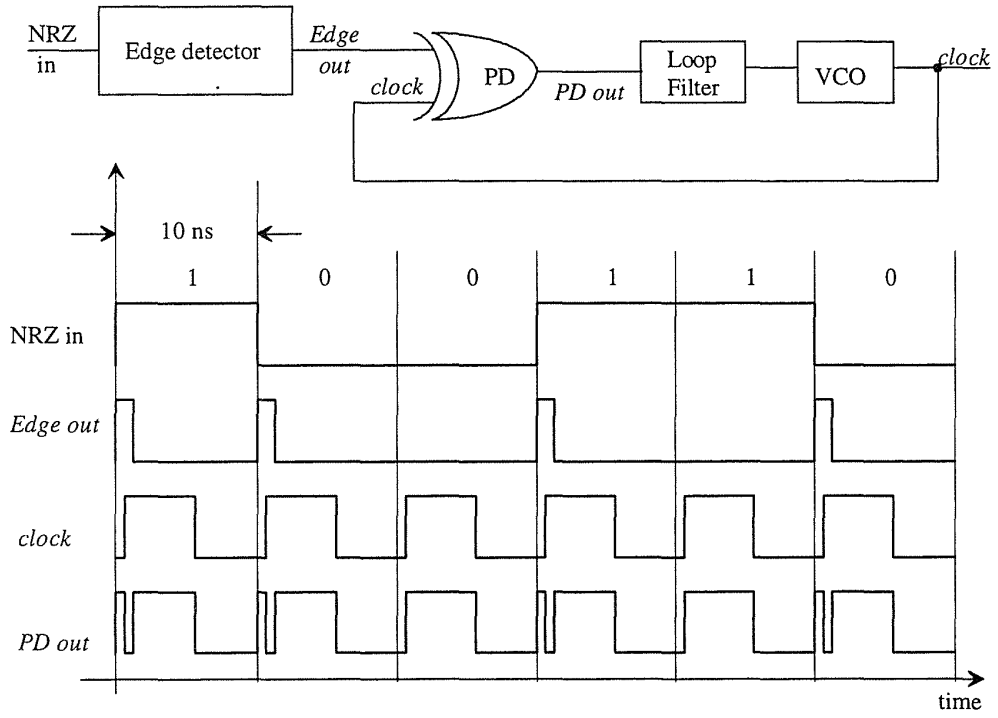



**Figure 19.46** Clock-recovery circuit for NRZ using an edge detector. Note that the DPLL is in lock, when the rising edge of clock is centered on the edge output pulse.

signal will not solve this problem since the temperature dependence and process variations of the associated circuit would not guarantee proper alignment. What is needed is a circuit that is *self-correcting*, causing the clock signal to align to the center of the data bit independent of the data-rate, temperature, or process variations.

The phase-detector portion of a self-correcting clock-recovery circuit is shown in Fig. 19.47 [7] along with associated waveforms for a locked DPLL. Nodes A and B are simply the input NRZ data shifted in time by one-half bit-interval and one bit-interval, respectively. The outputs of the phase detector are labeled *Increase* and *Decrease*. If *Increase* is low more often than *Decrease* the average voltage out of the loop filter and thus the frequency out of the VCO will decrease. A loop filter that can be used in a self-correcting DPLL is shown in Fig. 19.48a. This filter is the active-PI loop filter discussed in Sec. 19.3.1, with an added input to accommodate both outputs of the PD. Figure 19.48b shows the resulting waveforms in a DPLL where the clock is leading the center of the NRZ data, and thus *Increase* is high less often than *Decrease*. If the NRZ data were lagging the center of the data bit *Decrease* would be high less often than *Increase,* resulting in an increase in the loop-filter output voltage. Note that in this discussion we have neglected the propagation delays present in the circuit. For a

Microchip Ex. 1055, p. 438
Microchip v. Aptiv
IPR2024-00494

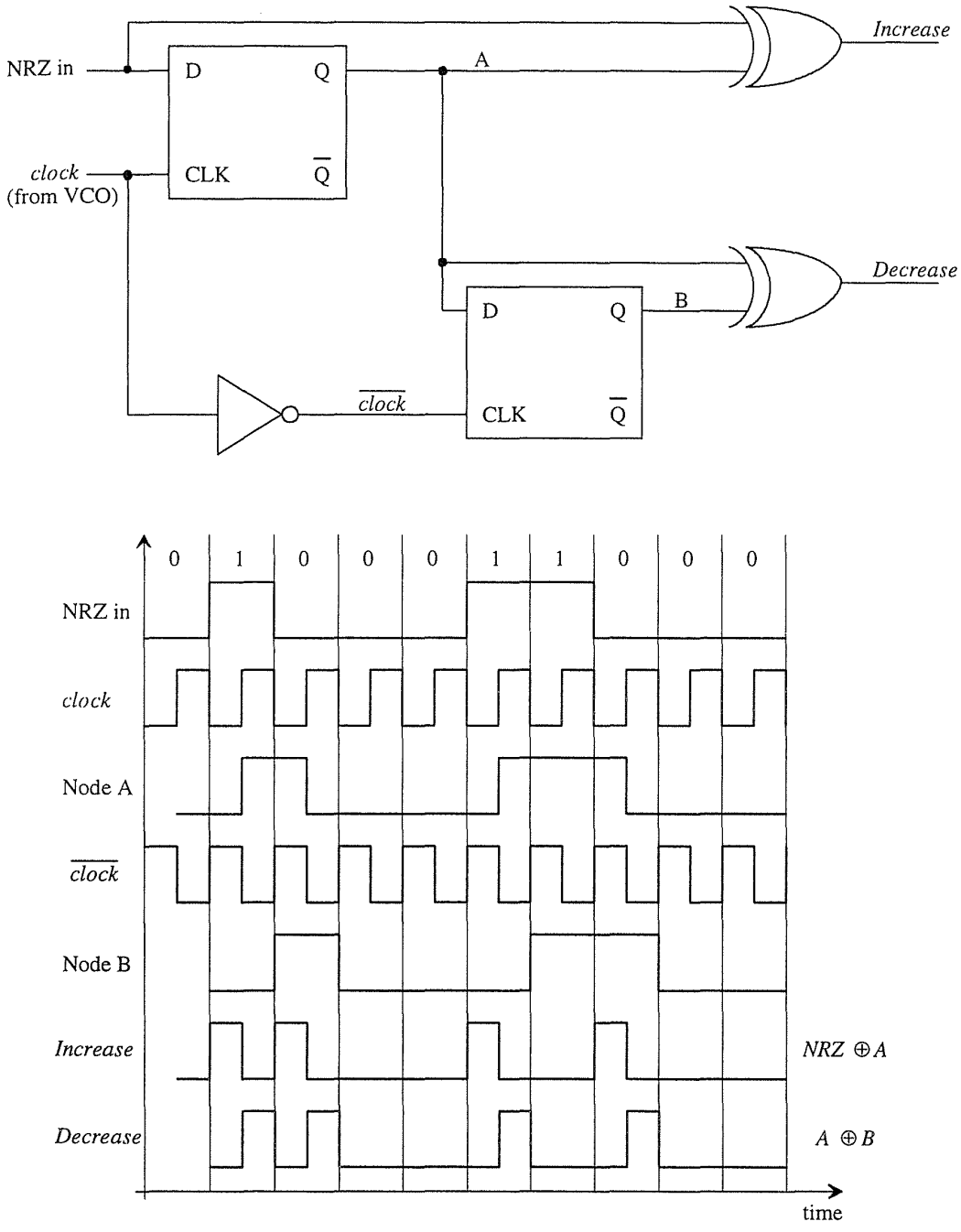
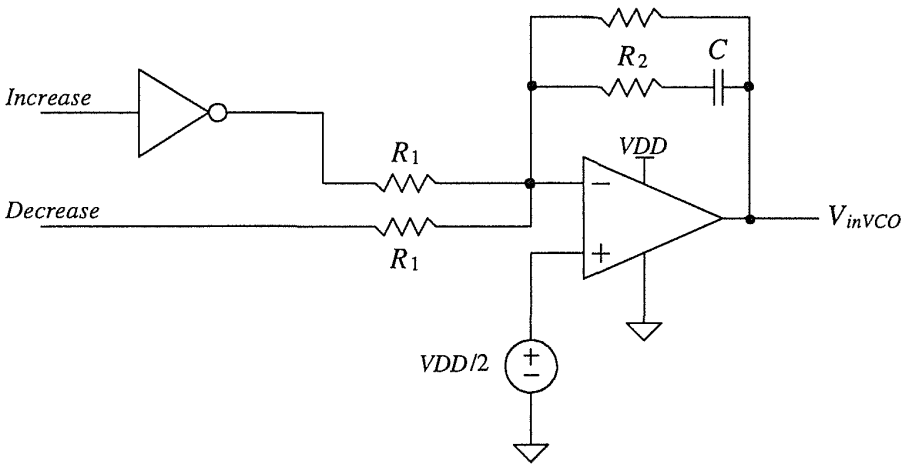
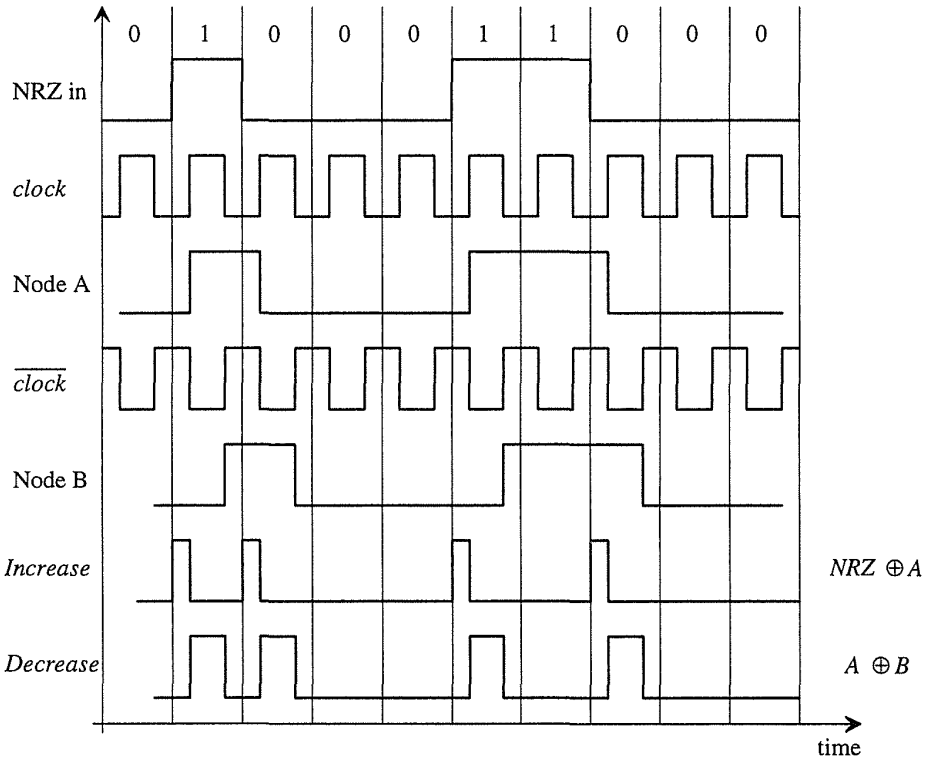



**Figure 19.47** The PD portion of a self-correcting clock-recovery circuit in lock [7].

Microchip Ex. 1055, p. 439
Microchip v. Aptiv
IPR2024-00494




(a)




(b)

**Figure 19.48** (a) Possible loop filter used in a self-correcting DPLL and (b) waveforms when the loop is not in lock and the clock leads the center of the data.

Microchip Ex. 1055, p. 440
Microchip v. Aptiv
IPR2024-00494

high-speed self-correcting PD design, we would have to analyze each delay in the PD to determine their effect on the perfomance of the DPLL.

*Jitter*

Jitter, in the most general sense, for clock-recovery and synchronization circuits, can be defined as the amount of time the regenerated clock varies once the loop is locked. Figure 19.49a shows the idealized case when the *clock* doesn't jitter, while Fig. 19.49b shows the actual situation where the clock-rising edge moves in time (jitters). In these figures, the oscilloscope is triggered by the rising edge of the *data*.

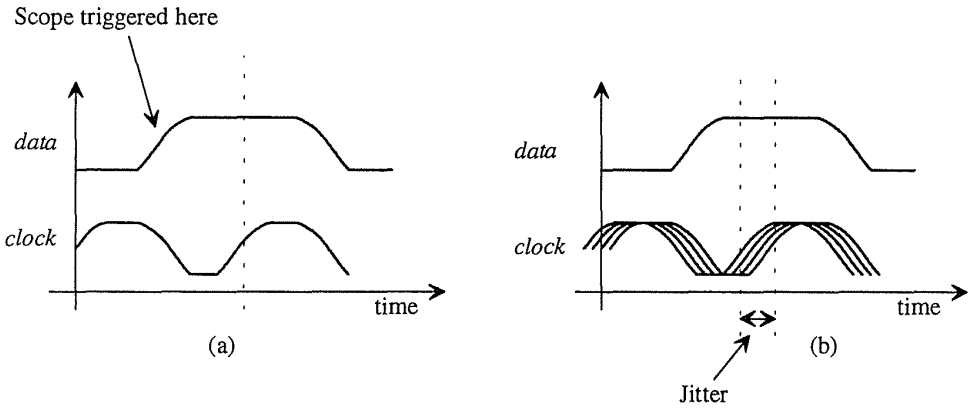



**Figure 19.49** (a) Idealized view of clock and data without jitter and (b) with jitter.

In the following discussion, we will neglect power supply and oscillator noise, that is, we will assume that the oscillator frequency is an exact number that is directly related to the VCO input voltage. In the section following this one, we will cover delay-locked loops and will further discuss the limitations of the VCO.

Consider using the charge pump with the self-correcting PD shown in Fig. 19.50. When the loop is locked (from Fig. 19.47), both increase and decrease occur (with the same width) for every transition in the incoming data. Note that unlike a PFD/charge pump combination where the output of the charge pump remains unchanged when the loop is locked, the self-correcting PD/charge-pump combination will generate a voltage ripple on the input of the VCO (similar to the XOR PD with RC or Active PI filter)[4]. Let's assume that this ripple is 10 mV and use the values for VCO gain and frequencies given in Ex. 19.5 to illustrate the resulting jitter introduced into the output clock. The change in output frequency resulting from this ripple is $10\text{mV}\cdot(628 \times 10^3 \text{ radians/V}\cdot\text{s})\cdot(1/2\pi)$ or 1 kHz. This means that the output of the

[4] Of course, the self-correcting PD should not be used in most frequency synthesis applications. Similarly, the PFD should not be used in most data-recovery applications.

Microchip Ex. 1055, p. 441
Microchip v. Aptiv
IPR2024-00494

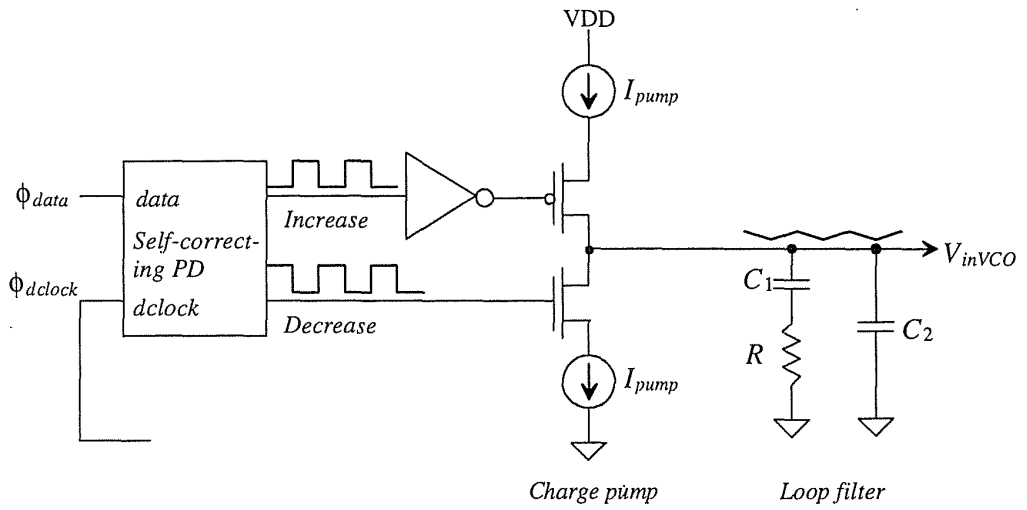



**Figure 19.50** Self-correcting PD with charge pump output.

DPLL will vary from 255.5 to 260.5 kHz or, in terms of a jitter specification the clock jitter is 75 ns.

From this example, jitter can be reduced by

1. *Reducing the gain of the VCO.* Ripple on the input of the VCO has less of an effect on the output frequency. The main disadvantage, in the most general sense, of reducing the gain of the VCO is that the range of frequencies the DPLL can lock up on is reduced. Also, the VCO gain strongly affects the ability to fabricate the VCO without postproduction tuning.

2. *Reducing the bandwidth of the loop filter.* This has the effect of reducing the actual ripple on the input of the VCO. The main drawback here is the increase in chip real estate needed to realize the larger components used in the filter.

3. *Reducing the gain of the PD.* This also has the effect of reducing the ripple on the input of the VCO. This method is, in general, the easiest when using the charge pump since reducing the gain is a simple matter of reducing $I_{pump}$. Substrate noise can become more of a factor in the design.

Another way of stating the above methods of reducing jitter is simply to say that the forward loop gain of the DPLL, that is, $K_{PD}K_FK_{VCO}$, should be made small. The main problems with using small forward loop gain are the reductions in lock range and pull-in range coupled with the associated increases in pull-in and lock times.

We have not discussed jitter reduction in DPLLs employed in frequency synthesizers or clock synchronizers where the input is a clock signal rather than a bit stream. In these circuits the VCO performance, in general, is the limiting factor.

Microchip Ex. 1055, p. 442
Microchip v. Aptiv
IPR2024-00494

## 19.5 Delay-Locked Loops

Problems with PLL output jitter resulting from the VCO output frequency changing (often called oscillator or phase noise) with a constant input voltage ($V_{inVCO}$ = constant) has led to the concept of a delay-locked loop (DLL). Figure 19.51 shows the basic block diagram of a DLL. *Assuming* a reference clock is available at exactly the correct frequency, the input data is delayed through a voltage-variable delay line (VVDL) a time $t_o$ until it is synchronized with the reference clock. Jitter is reduced by using an element, the VVDL, that does not generate a signal (like the VCO did). The transfer function $\phi_{clock}/\phi_{out}$ is zero[8] (the phase of the reference clock is taken as the reference for the other signals in the DLL, i.e., $\phi_{clock} = 0$), so that oscillator noise and the resulting jitter are not factors in DLL design. The jitter considerations discussed in the last section, however, are still a concern since any ripple on the output of the loop filter will cause jitter.

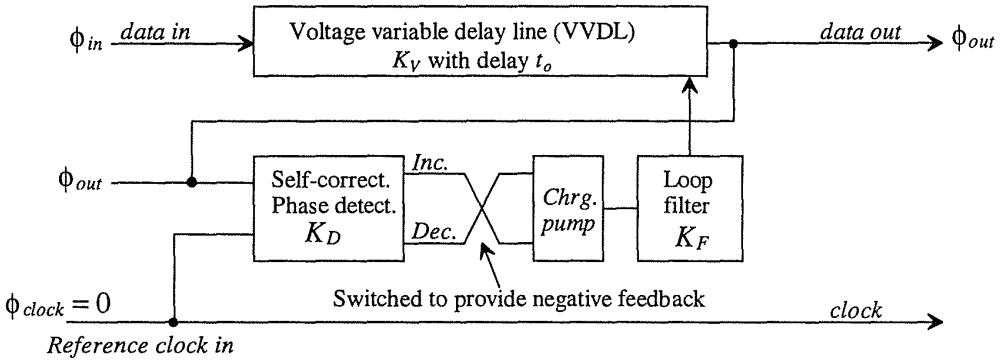



**Figure 19.51** Block diagram of a delay-locked loop.

The phase (in radians) of the input data is related to the phase of the output data by

$$\phi_{out} = \phi_{in} + t_o \cdot \frac{2\pi}{T_{clock}} \tag{19.62}$$

where $T_{clock}$ is the period of the reference clock (or half of the period of the *data in* for a string of alternating ones and zeros). The gain of the VVDL can be written in terms of the delay, $t_o$, by

$$t_o = K_V \cdot V_{outfilter} \tag{19.63}$$

where $K_V$ has units of seconds/V and $V_{outfilter}$ is the voltage input to the VVDL from the loop filter. The minimum and maximum delays of the VVDL should, in general, lie between $T_{clock}/2$ and $1.5T_{clock}$ for proper operation. The output of the loop filter can be written as

$$V_{outfilter} = \phi_{out} \cdot K_D \cdot K_F \tag{19.64}$$

Microchip Ex. 1055, p. 443
Microchip v. Aptiv
IPR2024-00494

The overall transfer function may now be written as

$$\frac{\phi_{out}}{\phi_{in}} = \frac{1}{1 - K_D K_F K_V \cdot \omega_{clk}} \tag{19.65}$$

where $\omega_{clk} = 2\pi/T_{clock}$. The gain of the self-correcting PD with charge-pump output, with the help of Fig. 19.52 and noting that *Increase* and *Decrease* can occur at the same time, is

$$K_D = -\frac{I_{pump}}{\pi} \text{ (amps/radian)} \tag{19.66}$$

The negative sign is the result of switching the *Increase* and *Decrease* outputs of the self-correcting PD before connection to the charge pump in order to provide negative feedback around the loop. Another benefit of the DLL is that the loop filter can be a simple capacitor, which results in a first-order feedback loop, that is,

$$K_F = \frac{1}{sC_1} \tag{19.67}$$

The transfer function of the DLL relating the input data to the time-shifted output is

$$\frac{\phi_{out}}{\phi_{in}} = \frac{1}{1 + \frac{I_{pump}}{\pi} \cdot \frac{1}{sC_1} \cdot K_V \cdot \omega_{clk}} = \frac{s}{s + K_V \cdot \frac{2I_{pump}}{C_1 T_{clock}}} \tag{19.68}$$

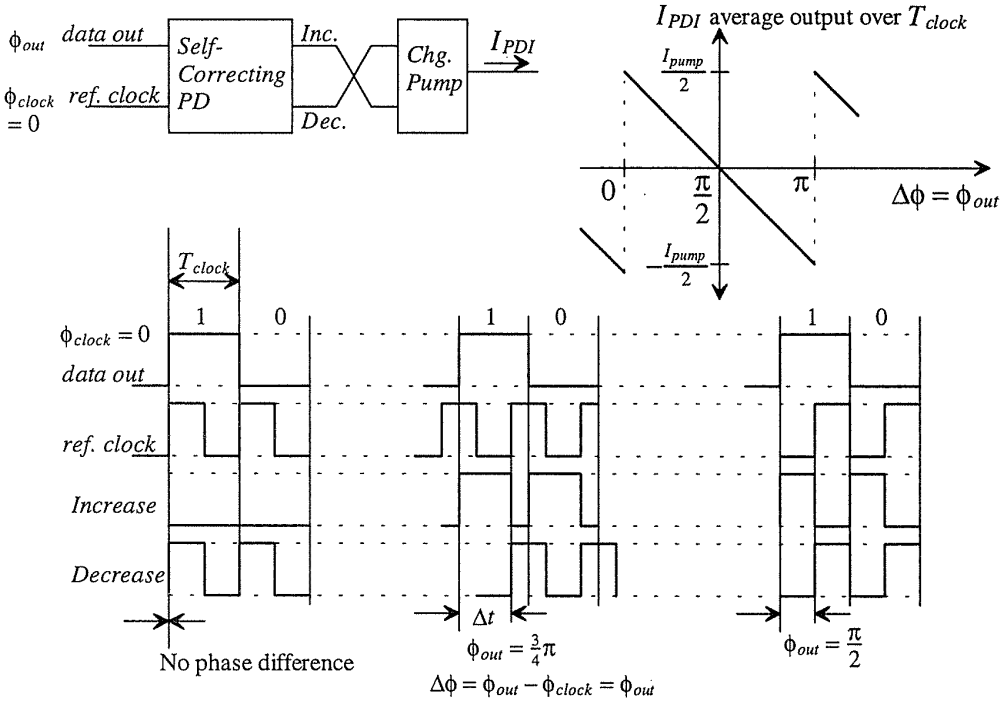



**Figure 19.52** Self-correcting PD output for various inputs (assuming input data is a string of alternating ones and zeros).

Microchip Ex. 1055, p. 444
Microchip v. Aptiv
IPR2024-00494

We know that the frequency of the reference clock must be exactly related to the frequency of the input data. However, there will exist instantaneous changes in the phase of the input data which the output of the DLL should follow. Modeling instantaneous changes in $\phi_{in}$ by $\Delta\phi_{in}/s$ (a step function with an amplitude of $\Delta\phi_{in}$), we get a change in output phase given by

$$\Delta\phi_{out} = \frac{\Delta\phi_{in}}{s + K_V \cdot \frac{2I_{pump}}{C_1 T_{clock}}} \tag{19.69}$$

The time it takes the DLL to respond to an input step in phase is simply

$$T_r = 2.2 \cdot \frac{C_1 T_{clock}}{K_V \cdot 2I_{pump}} \tag{19.70}$$

This time can be decreased by making $C_1/I_{pump}$ small, which from our discussion in the last section, has the result of increasing the output pulse jitter, that jitter that is independent of the oscillator (i.e., dependent on the input data pattern) in a standard PLL. Decreasing $C_1/I_{pump}$ increases the ripple on the control voltage of the VVDL. Similarly, increasing $K_V$ (the time/volt delay of the VVDL) will increase jitter since a given ripple on the control voltage of the VVDL will have a larger effect on the delay. Again, tradeoffs must be made between response to input variations and output jitter.

*Delay Elements*

The VVDL is an important component of the DLL. Figure 19.53a shows the basic implementation of a VVDL using adjustable delay inverters. The last two inverters in the VVDL are used to ensure that clean digital signals are output from the VVDL.

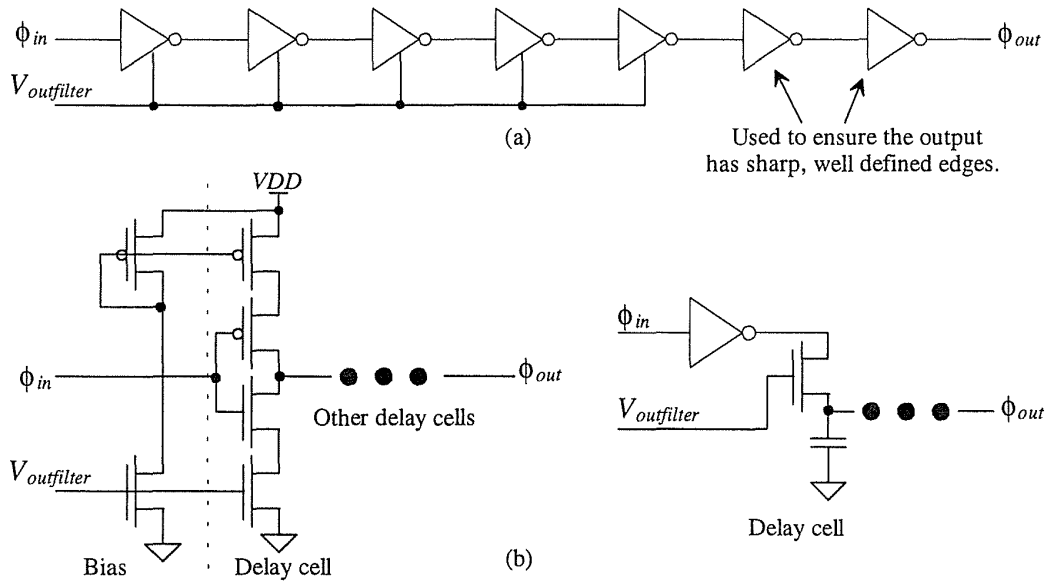



**Figure 19.53** (a) VVDL made using inverter delay cells and (b) possible delay cells.

Microchip Ex. 1055, p. 445
Microchip v. Aptiv
IPR2024-00494

Figure 19.53b shows the circuit schematics for possible delay elements [5, 9-10]. The first delay element should be recognized as a current-starved inverter discussed earlier in the chapter. The second delay element is nothing more than an inverter with a variable load. In practice, these delay elements are rarely used because of the susceptibility to noise and power supply variations but rather fully-differential delay elements are used[5].

Figure 19.54 shows the connection of a fully differential VVDL. Using any number of stages, a fully differential VCO can be implemented with the delay elements (and the proper feedback), while using an even number with the inverting (noninverting) output fed back and connected to the noninverting (inverting) input in-phase (I) and quadrature (Q) signals can be generated [5].

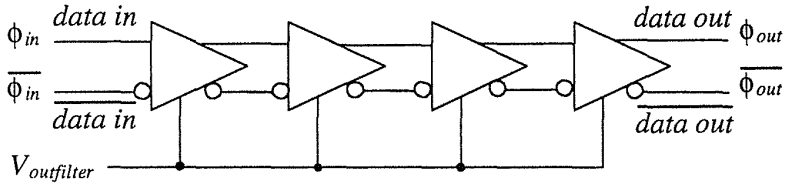



**Figure 19.54** Implementation of a fully differential VVDL.

A simple differential delay element is seen in Fig. 19.55a. This circuit is basically a fully differential inverter. The p-channel loads are long L so that the change in $V_{outfilter}$ can more linearly adjust the delay of the element. The signals out of this element are not truly differential because of the unequal high-to-low and low-to-high times so that good power supply rejection is difficult to achieve. The delay element

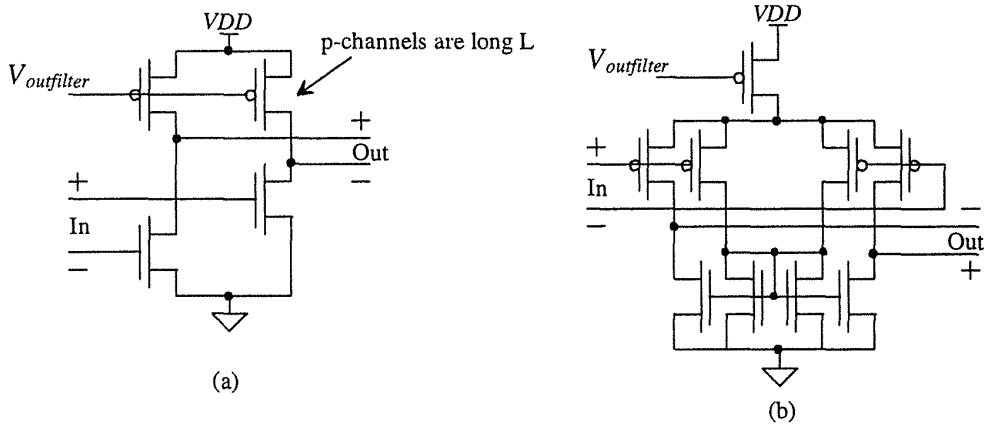



**Figure 19.55** (a) A fully differential inverter and (b) a diff-amp-based delay element. Note that the delay element of (a) cannot be used in a differential VCO.

Microchip Ex. 1055, p. 446
Microchip v. Aptiv
IPR2024-00494

shown in Fig. 19.55b has much better power supply rejection because of the symmetry of the circuit. This circuit will sink and source the same amount of current into a load capacitance (the input capacitance of the next stage) resulting in equal rise and fall times through the delay line. Another example of a delay stage is shown in Fig. 19.56 [10].

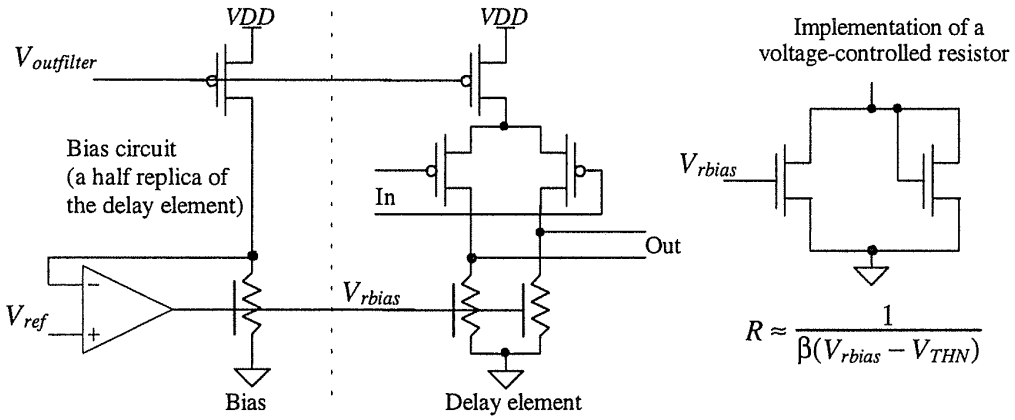



**Figure 19.56** A differential delay element based on a voltage-controlled resistor. The bias circuit adjusts the value of the resistors used in the delay elements to sink the current sourced by the p-channel MOSFETs.

*Clock Synchronization*

Consider the block diagram of the interconnection of a CPU and slave processor [11] shown in Fig. 19.57. The DLL is used in this diagram to synchronize the clock in the master CPU with the clock used in the slave processor. This is very important when common external data/address busses are shared between processors or a processor and memory. Since the "data" input to the DLL is a clock signal that is, it is not data but rather a square-wave, the design of the DLL can be simplified. In particular, we can design the DLL so that it uses only digital components. No analog loop filter or VVDL is needed. Note that simply routing the clock signal around the CPUs (not using a

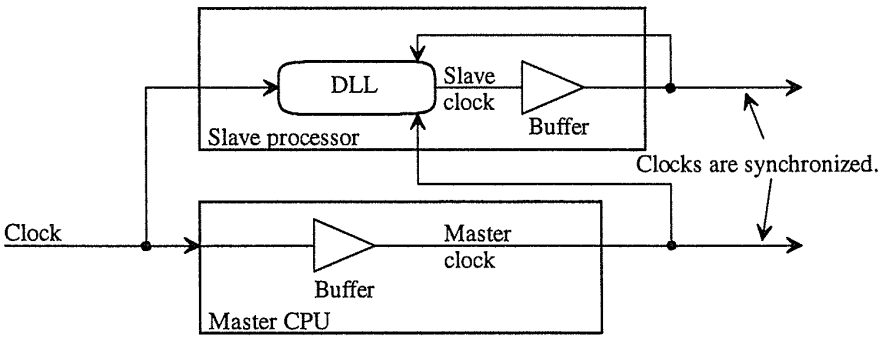



**Figure 19.57** Synchronizing the master CPU clock with the clock in the slave processor.

Microchip Ex. 1055, p. 447
Microchip v. Aptiv
IPR2024-00494

DLL) does not guarantee that the data coming out of the processors are synchronized since the internal delays of the master and slave will not be the same (they are skewed).

The block diagram of a digital-only DLL useful in clock synchronization circuits is shown in Fig. 19.58 [11]. A seven-bit counter is used, in place of an analog loop filter, to select, via the decoder, which tap of the delay line is fed to the output buffers.

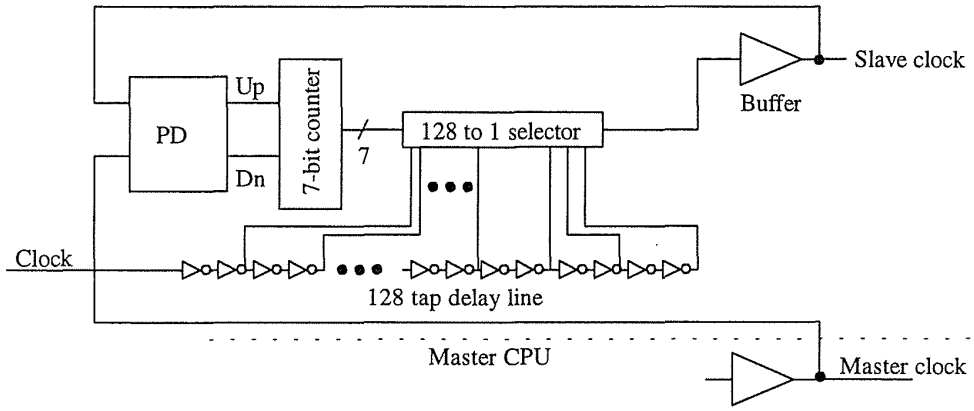



**Figure 19.58** A purely digital DLL.

## REFERENCES

[1] R. E. Best, *Phase-Locked Loops, Theory, Design and Applications,* McGraw-Hill, 2nd ed., 1993. ISBN 0-07-911386-9.

[2] F. M. Gardner, "Charge-Pump Phase-Lock Loops," *IEEE Transactions on Communications,* COM-28, No. 11, pp. 1849-1858, November 1980.

[3] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits,* 3rd ed., John Wiley and Sons, 1993. ISBN 0-471-57495-3.

[4] B. Keeth, R. J. Baker, and H. W. Li, "CMOS Transconductor VCO with Adjustable Operating and Center Frequencies," *Electronics Letters,* Vol. 31, No. 17, pp. 1397-1398, August 1995.

[5] B. Razavi, *Monolithic Phase-Locked-Loops and Clock Recovery Circuits,* IEEE Press, 1996. ISBN 0-7803-1149-3.

[6] S. Haykin, *An Introduction to Analog and Digital Communications,* John Wiley and Sons, 1989. ISBN 0-471-85978-8.

[7] C. R. Hogge, Jr., "A Self Correcting Clock Recovery Circuit," *IEEE Journal of Lightwave Technology*, Vol. LT-3, pp. 1312-1314, December 1985.

[8] T. H. Lee and J. F. Bulzacchelli, "A 155-MHz Clock Recovery Delay- and Phase-Locked Loop," *IEEE Journal of Solid-State Circuits,* Vol. SC-27, pp. 1736-1746, December 1992.

Microchip Ex. 1055, p. 448
Microchip v. Aptiv
IPR2024-00494

[9] M. G. Johnson and E. L. Hudson, "A Variable Delay Line PLL for CPU - Coprocessor Synchronization," *IEEE Journal of Solid-State Circuits,* Vol. SC-23, pp. 1218-1223, October, 1988.

[10] I. A. Young, J. K. Greason, and K. L. Wong, "A PLL Clock Generator with 5 to 110 MHz of Lock Range for Microprocessors," *IEEE Journal of Solid-State Circuits,* Vol. SC-27, pp. 1599-1607, November 1992.

[11] A. Efendovich, Y. Afek, C. Sella, and Z. Bikowsky, "Multifrequency Zero-jitter Delay-Locked Loop," *IEEE Journal of Solid-State Circuits,* Vol. 29, pp. 67-70, January, 1994.

## PROBLEMS

**19.1** Referring to Fig. 19.8, sketch the phase difference between *dclock* and *data* when the width of a data bit is 1 μs and the time between the leading edges of *dclock* and *data* is 0.2 μs. What is the phase difference? What is the average PD output voltage, $V_{PDout}$ for *data* that is an alternating string of ones and zeros.

**19.2** Estimate the phase difference and sketch the PFD outputs (both *Up* and *Down*) for the data shown in Fig. P19.2.

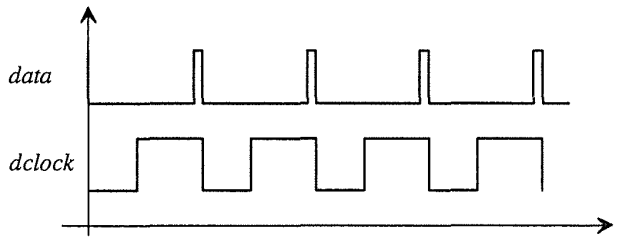



**Figure P19.2**

**19.3** Design and simulate the operation of a 10 MHz VCO (at $V_{inVCO} = VDD/2$) using a current-starved VCO.

**19.4** Repeat Problem 19.3 for a source coupled VCO design.

**19.5** Derive Eq. (19.30).

**19.6** Sketch a simple circuit to implement a divide by 2 in the feedback loop of a DPLL.

**19.7** Sketch the phase relationship, in the time domain, between *data* and *dclock* in Ex. 19.3 when the input data rate is 9.04 Mbits/s. Use the results shown in Fig. 19.25.

**19.8** Derive Eq. (19.41). State all assumptions.

**19.9** Repeat Ex. 19.5 for an input of 1 KHz and an output of 3 kHz. Sketch the logic diagram for the divide by 3 in the feedback loop.

Microchip Ex. 1055, p. 449
Microchip v. Aptiv
IPR2024-00494

**19.10** Verify, in Fig. 19.40, that the resistors average $v_{+peak}$ and $v_{-peak}$.

**19.11** Describe a scheme to implement the return to zero data format shown in Fig. 19.42.

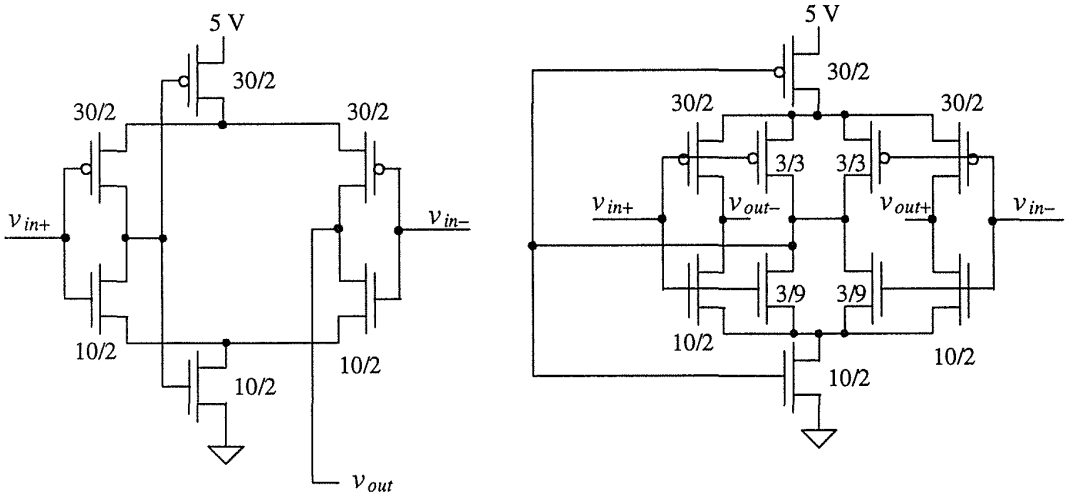
**19.12** Using the CN20 CMOS process, design an edge detector that generates outputs with a width of approximately 5 ns.

**19.13** What is the optimum delay that should be used in an edge detector? Use the waveforms in Fig. 19.46 to illustrate the resulting PD output when this optimum delay is used.

**19.14** Sketch *Increase* and *Decrease*, using the NRZ data in Fig. 19.47, when the rising edge of *clock* is aligned with the beginning of the bit interval.

**19.15** Sketch the signals, similar to Fig. 19.52, when the output phase of the DLL is $\pi/4$.

**19.16** An important circuit in a receiver is the input comparator or decision circuit. This circuit changes the analog data, with possible amplitudes of a few hundreds of mV, into digital data. Figure P19.16 shows both a single-ended output (see also Figure 26.8) and a differential output reciever useful in high-speed digital design. Using SPICE with $v_{in-}$ fixed at 2.5V determine the minimum voltage swing on $v_{in+}$ (centered around 2.5 V) that will cause the outputs to swing from <1 V to > 4 V. What are the unloaded propagation delays of these circuits?




Single-ended output

Differential-ended output

**Figure P19.16**

Microchip Ex. 1055, p. 450
Microchip v. Aptiv
IPR2024-00494

Part

III

# CMOS Analog Circuits

Microchip Ex. 1055, p. 451
Microchip v. Aptiv
IPR2024-00494

Chapter

20

# Current Sources and Sinks

The current source/sink is a basic building block in CMOS IC design and is used extensively in analog integrated circuit design. Ideally, the output impedance of a current source/sink should be infinite and capable of generating or drawing a constant current over a wide range of voltages. However, finite values of $r_o$, and a limited output swing required to keep devices in saturation will ultimately limit the performance of the mirror. The circuits in this chapter will vary in complexity and performance as the design, simulation, and layout issues associated with MOS current sources and sinks are examined.

## 20.1 The Current Mirror

Figure 20.1 shows the basic current mirror [1-3]. A current flows through M1 corresponding to $V_{GS1}$. Since $V_{GS1} = V_{GS2}$, ideally the same current, or a multiple of the current in M1, flows through M2. If the MOSFETs are the same size, the same drain current flows in each MOSFET, provided M2 stays in the saturation region. The current $I_{D1}$ is given by

$$I_{D1} = \frac{\beta_1}{2} \cdot (V_{GS1} - V_{THN})^2 \tag{20.1}$$

while the output current, assuming M2 is in saturation, flowing in M2 is

$$I_{D2} = I_o = \frac{\beta_2}{2}(V_{GS2} - V_{THN})^2 \tag{20.2}$$

Since $V_{GS1} = V_{GS2}$, the ratio of the drain currents is given by

$$\frac{I_{D2}}{I_{D1}} = \frac{\frac{W_2}{L_2}}{\frac{W_1}{L_1}} = \frac{W_2 L_1}{W_1 L_2} = \frac{\beta_2}{\beta_1} \tag{20.3}$$

This equation shows how to adjust the *W/L* ratio of the two devices to achieve the desired output current, $I_{D2}$. This equation, however, does not show how the output

Microchip Ex. 1055, p. 452
Microchip v. Aptiv
IPR2024-00494

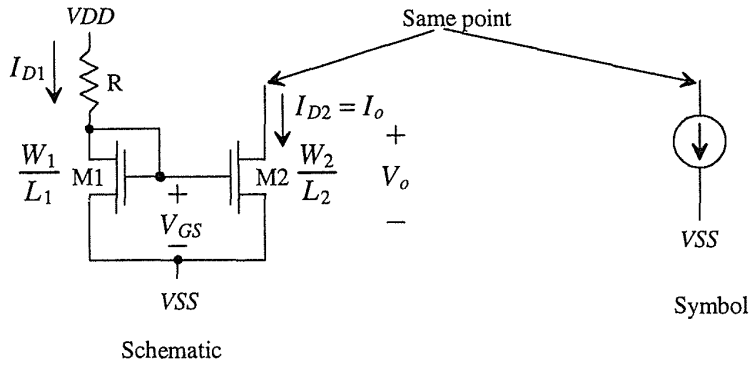



**Figure 20.1** The basic current mirror schematic and symbol.

current will change with the voltage across M2, $V_o$. The reference drain current, $I_{D1}$ in Fig. 20.1, is determined by solving

$$I_{D1} = \frac{VDD - V_{GS} - VSS}{R} = \frac{KP \cdot W_1}{2L_1} \cdot (V_{GS1} - V_{THN})^2 \qquad (20.4)$$

The minimum voltage, $V_{min}$, across the current sink is set by the requirement that M2 remain in saturation, that is, $V_{min} = V_{DS,SAT} = V_{GS} - V_{THN}$. The output resistance of the current source is simply the output resistance of M2, or

$$r_{o2} = \frac{1}{\lambda I_o} = \frac{1}{\lambda I_{D2}} \qquad (20.5)$$

Looking at Eqs. (20.3) and (20.4), the five variables, $L_1$, $L_2$, $W_1$, $W_2$, and $V_{GS}$, are available to the designer to set the currents. Normally, when designing CMOS current mirrors, the values for $V_{GS}$ and $L$ are selected before solving for $W$ to get the desired current. Picking the lengths of all MOSFETs, used in current sources, the same size simplifies Eq. (20.3) to,

$$\frac{I_{D2}}{I_{D1}} = \frac{W_2}{W_1} \qquad (20.6)$$

For digital design we used the minimum length, 2.0 μm for CN20, for our MOSFET switches because we did not care about the output resistance of the MOSFET in the saturation region. For analog design, however, it is extremely important to keep the output resistance as high as possible. It is also desirable to reduce the effects of channel length and mobility modulation discussed in Chs. 5, 6, and 9. These effects are reduced by increasing the channel length of the devices. A general design rule is to set the length of the MOSFETs used in analog applications to two to five times the minimum drawn gate length. Appendix A shows output resistance for different-sized MOSFETs. If we use Eq. (20.5) with this data, we can conclude that a 5 μm device length will give about the same output resistance as a 10 μm device. The sum of the mobility modulation and channel length modulation parameters, λ, for the n- and p-channel MOSFETs is approximately 0.06 $V^{-1}$. As the plots show, the values of λ, and thus the

Microchip Ex. 1055, p. 453
Microchip v. Aptiv
IPR2024-00494

output resistances, are very dependent on the biasing conditions. From these considerations we can make the following statement: *When designing with the CN20 process, we will assume that the length, L, of the MOSFETs used in analog applications is at least 5 μm.*

The biasing of current mirrors, or any other analog circuit, is greatly simplified if we also assume, and design for, a specific gate-source voltage[1], $V_{GS}$. Setting $V_{GS}$ (or $V_{SG}$ for the p-channel) close to the threshold voltage, approximately 0.8 V for CN20, results in very large devices, while setting $V_{GS}$ significantly larger than the threshold voltage causes the transistor to enter the triode region too early. An acceptable difference between $V_{GS}$ and $V_{THN}$, sometimes referred to as the excess gate voltage, $\Delta V$, is several hundred millivolts. Using this general design rule we can make the following assumption: *When designing current sources/sinks with the CN20 process, the gate-source voltage will be initially assumed to be 1.2 V.*

These rules are a starting point in the design. They allow the designer to concentrate on two variables in the basic current mirror (the widths of the devices) rather than five. The $V_{GS}$ is normally adjusted to obtain a desired characteristic (such as minimum voltage across the current source/sink).

**Example 20.1**

Design a current sink using $VDD = -VSS = 2.5$ V to sink a current of 10 μA. Estimate the minimum voltage across the current source and the output resistance. Simulate the operation of your design with SPICE.

The basic design is shown in Fig. Ex20.1. Here we have selected $V_{GS} = 1.2$ V and the length of the devices as 5 μm. The value of $R$, assuming $I_{D1} = I_{D2} = 10$ μA, is determined by solving Eq. (20.4), or

$$R = \frac{2.5 - 1.2 - (-2.5)}{10\ \mu A} = 380\ k\Omega$$

We can solve for the width of both M1 and M2 at the same time,

$$I_{D2} = 10\ \mu A = \frac{KP}{2}\frac{W}{L}(V_{GS} - V_{THN})^2 = \frac{50\frac{\mu A}{V^2}}{2}\frac{W}{5\ \mu m}(1.2 - 0.83)^2$$

which gives $W_1 = W_2 = 14.61$ μm which we round up to 15 μm. The requirement for M2 to stay in the saturation region is

$$V_{DS2} \geq V_{GS2} - V_{THN} = 1.2 - 0.83 = 0.37\ V = \text{excess gate voltage}$$

As long as the drain of M2 is approximately - 2.13 V or greater, M2 will remain in the saturation region. This result is *important*! When the MOSFETs are

[1] This assumption is similar to assuming that the forward voltage drop of a base-emitter junction is 0.7 V. It may be 0.6 or 0.8 V, but we still assume 0.7 V to simplify the biasing of bipolar analog circuits.

Microchip Ex. 1055, p. 454
Microchip v. Aptiv
IPR2024-00494

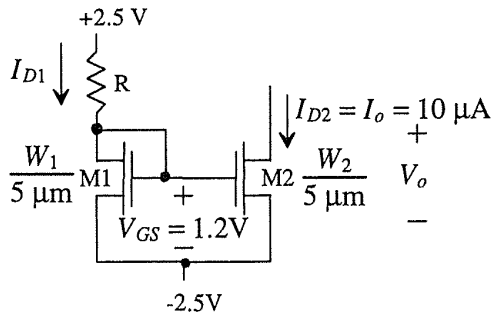



**Figure Ex20.1**

biased with a $V_{GS}$ of 1.2 V, the minimum voltage from drain to source is 0.37 V (the excess gate voltage) for operation in the saturation region.

The small-signal output resistance of the current source is then approximated using Eq. (20.5), by

$$r_o = \frac{1}{\lambda I_o} = \frac{1}{0.06 \cdot 10\ \mu A} = 1.67\ \text{MEG}$$

For AC small-signal analysis, simply replace the current sources with resistors equivalent to the output impedance connected to ground. The output resistance can also be thought of as the resistance looking into the drain of M2. SPICE simulation results are shown in Fig. 20.2. Notice the effect of the finite output resistance. ■

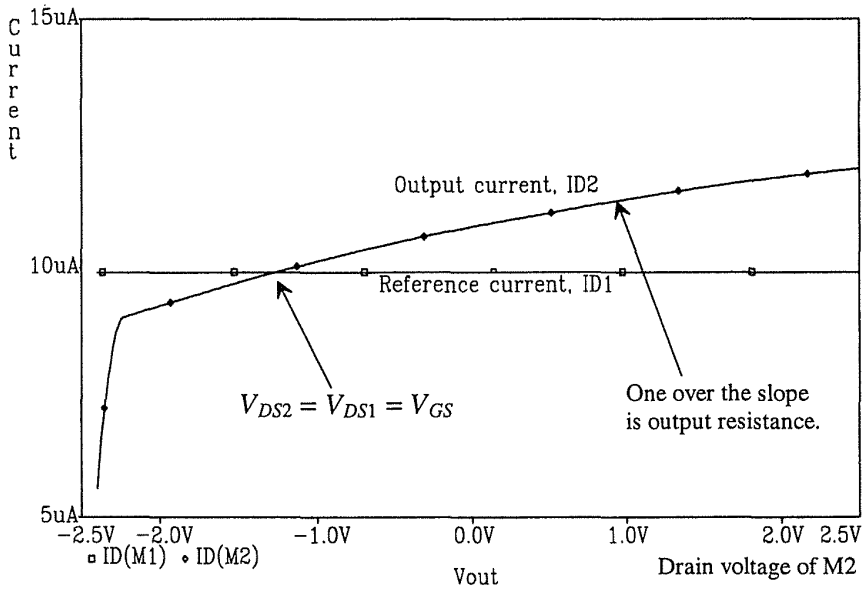



**Figure 20.2** Example 20.1 SPICE simulation results.

Microchip Ex. 1055, p. 455
Microchip v. Aptiv
IPR2024-00494

**Example 20.2**

Design four current sinks with values 20, 30, 50, and 70 μA. What is the minimum voltage across each current sink? Again assume $VDD = -VSS = 2.5$V.

Following the design procedure of the previous example, we see that this problem is simply a matter of re-sizing the MOSFETs used for sinking current to get the 20, 30, 50, and 70 μA required. The schematic of the design with new sizes is shown in Fig. Ex20.2. Layout methods to avoid problems with oxide encroachment and lateral diffusion are discussed in Sec. 20.1.5. Using a single reference with several current sinks is very common in IC design. Also, note that the 1.2 V we assumed for the value of $V_{GS}$ is not exact; it may be 1.1 or 1.3 V. In this configuration, using the reference MOSFET with its gate shorted to its drain, we observe that the small variations in $V_{GS}$ have little effect on the 10 μA flowing in the reference MOSFET. The minimum voltage across any of the current sinks is, from the previous example, 0.37 V corresponding to a minimum voltage on the drains of –2.13 V. ■

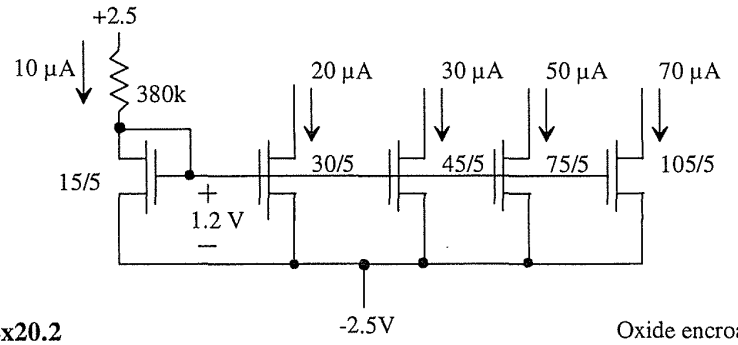



**Figure Ex20.2**

Oxide encroachment has been neglected here.

Consider the two MOSFETs shown in Fig. 20.3a, each biased with the same current $I_1$. Since the gate-source voltages of the devices, M2 and M3, and the drain currents are equal, it follows that the drain-source voltages of the devices must be equal, that is, $V_{GS1} = V_{GS2} = V_{DS1} = V_{DS2}$. In Fig. 20.3b, this fact is used to bias a third MOSFET, M3, that is, $V_{GS3} = V_{GS1} = V_{GS2}$. For biasing purposes, we treat the gate of M3 as if it were connected to the gates of M1 and M2. This configuration is very useful in amplifier design where M2 is used as a current source load and M3 is a common source amplifier.

Current sinks are made using n-channel MOSFETs, while current sources are made using p-channel MOSFETs. We can bias the p-channel MOSFETs in the same way we biased the n-channel MOSFETs to form a current source. A p-channel MOSFET mirror is shown in Fig. 20.4 along with its schematic symbol. Sizing the devices for a particular current follows the general design rules as well, that is, $V_{SG}$ = 1.2 V and $L = 5$ μm. The following example illustrates the design of a current source.

Microchip Ex. 1055, p. 456
Microchip v. Aptiv
IPR2024-00494

Calculating the minimum voltage across the current source begins by defining the gate-source biasing voltage in terms of the excess gate-source voltage $\Delta V$ as

$$V_{GS} = \Delta V + V_{THN} \tag{20.7}$$

Since $V_{GS}$ = 1.2 V, the n-channel excess gate voltage, neglecting the body effect, is $\Delta V = 0.37$ V (see Ex. 20.1) and the p-channel excess gate voltage is 0.290 *V*. The gate voltage of M4 is $2(\Delta V + V_{THN}) = 2.4$ V, while the source voltage of M4 is $\Delta V + V_{THN} = 1.2$ V. The minimum voltage on the drain of M4, and thus the minimum voltage across the current source, are limited by the requirement that M4 remain in the saturation region, $V_{DS4} \geq V_{GS4} - V_{THN}$ or $V_{D4} \geq \Delta V + (\Delta V + V_{THN}) = 1.57$ V. The minimum voltage across the cascode current mirror is significantly larger than the minimum voltage across the basic current mirror ($\Delta V = 0.37$) calculated in Ex. 20.1.

If the voltage on the gate of M4 can be reduced to $2\Delta V + V_{THN}$, then the voltage on the drain of M2 becomes $\Delta V = 0.37$ V, and the minimum voltage across the current source is reduced to $2\Delta V = 0.74$ V. The circuit shown in Fig. 20.6a illustrates this idea [4]. A battery (M6) is used to drop the potential at the gate of M4 down to $2\Delta V + V_{THN}$. This reduces the voltage on the drain of M4 to $2\Delta V$ before M2 and M4 enter the triode region. Implementation of this current source is shown in Fig. 20.6b. The MOSFET M3 is re-sized to generate $3\Delta V + 2V_{THN}$, that is, $V_{GS3} = 2\Delta V + V_{THN} = 1.57$V, on its gate while M6 is used to drop $\Delta V + V_{THN}$ so that the gate voltage of M4 becomes $2\Delta V + V_{THN}$. To accomplish this, the width of M3 is made one-fourth the size of the other MOSFETs. Note that a MOSFET with its gate and drain tied together being fed by a constant current (M1 and M3 in Fig. 20.6) behaves as a constant DC potential (a battery).

The output resistance of the cascode current mirror can be determined using the small-signal models derived in Ch. 9 and shown in Fig. 20.7. A test voltage is applied to the drain of M4. Summing the currents at the drain of M4 and neglecting the body effect result in

$$i_t = g_{m4} v_{gs4} + \frac{v_t - (-v_{gs4})}{r_{o4}} \tag{20.8}$$

where

$$v_{gs4} = -i_t \cdot r_{o2} \tag{20.9}$$

The output resistance of the cascode current source is given by

$$R_o = r_{o4}(1 + g_{m4} r_{o2}) + r_{o2} \approx g_{m4} r_o^2 \tag{20.10}$$

where $r_o = r_{o2} = r_{o4}$. This result can be applied to the more general case shown in Fig. 20.8. The small-signal resistance to ground looking into the drain of the MOSFET is

$$R_o = r_o(1 + g_m R) + R \tag{20.11}$$

which is sometimes approximated by

$$R_o \approx r_o(1 + g_m R) \tag{20.12}$$

Microchip Ex. 1055, p. 457
Microchip v. Aptiv
IPR2024-00494

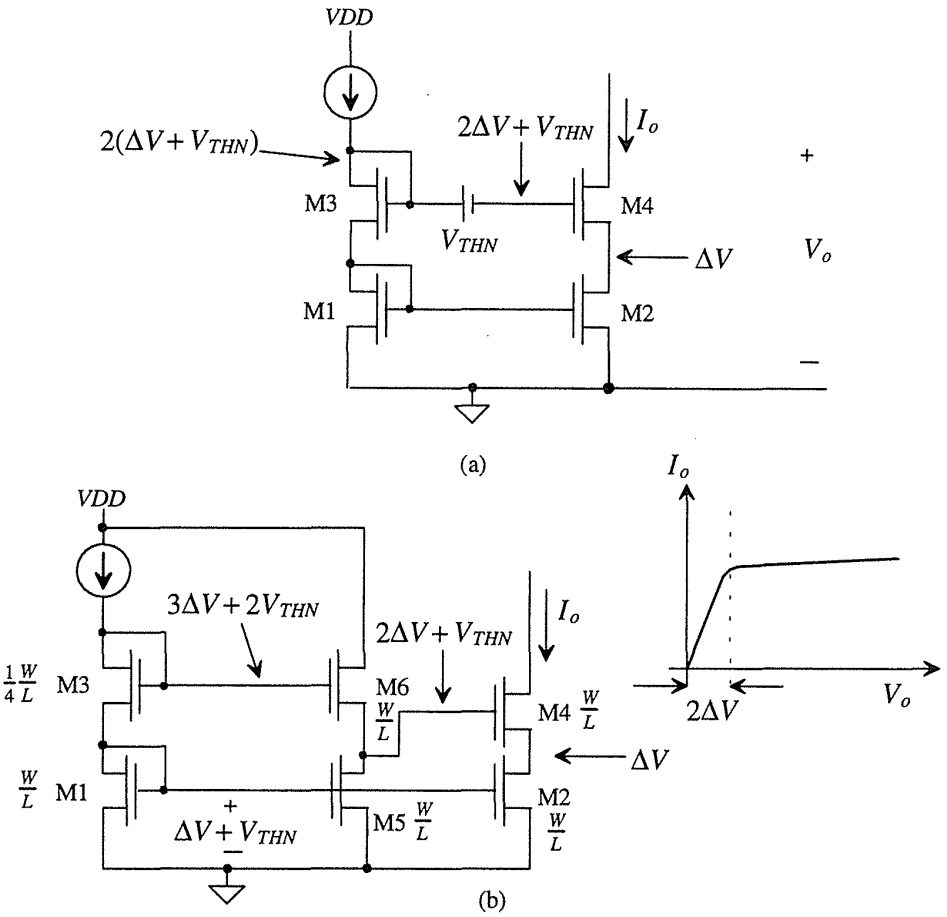



**Figure 20.6** Biasing of the cascode current source for lower minimum voltage across the current source (body effects neglected).

The output resistance of the triple cascode shown in Fig. 20.9 can be written using these results as

$$R_o = r_{o6}\left[1 + g_{m6}\overbrace{(r_{o4}(1+g_{m4}r_{o2})+r_{o2})}^{R \text{ looking into drain of M4}}\right] + \overbrace{r_{o4}(1+g_{m4}r_{o2})+r_{o2}}^{R \text{ looking into drain of M4}} \approx g_{m6}g_{m4}r_o^3 \quad (20.13)$$

When performing an AC small-signal analysis, the cascode connection is simply replaced with a resistor to ground of value given by Eq. (20.10) for the double cascode or Eq. (20.13) for the triple cascode. Increasing the output resistance of a current source or sink is simply a matter of stacking up the MOSFETs until the desired output resistance is reached. The obvious limitation to stacking a large number of cascode devices is the increase in the minimum voltage across the current source/sink needed to keep all devices in the saturation region. Other current mirrors that improve the output impedance using negative feedback will be examined later in the chapter.

Microchip Ex. 1055, p. 458
Microchip v. Aptiv
IPR2024-00494

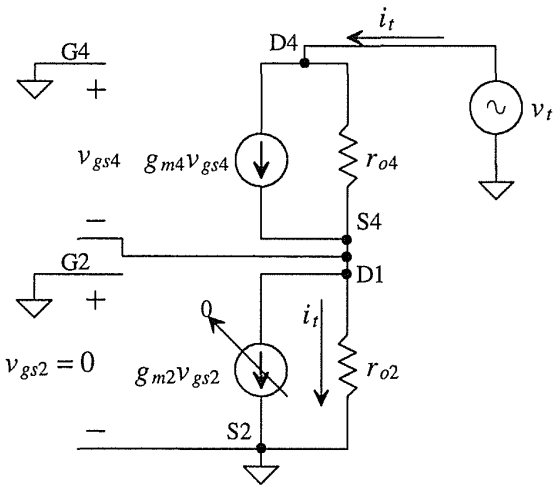



**Figure 20.7** Small-signal model of the cascode current source shown in Fig. 20.5.

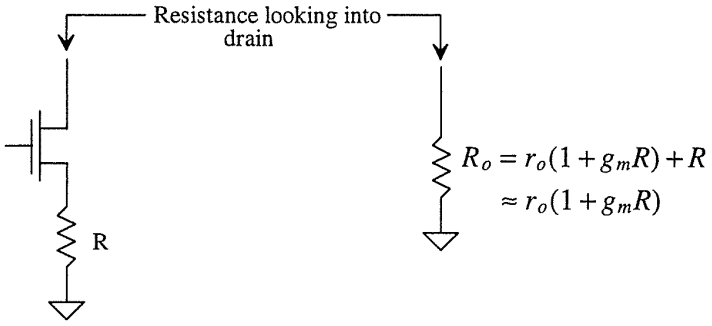



**Figure 20.8** Modeling the resistance looking into the drain of a MOSFET.

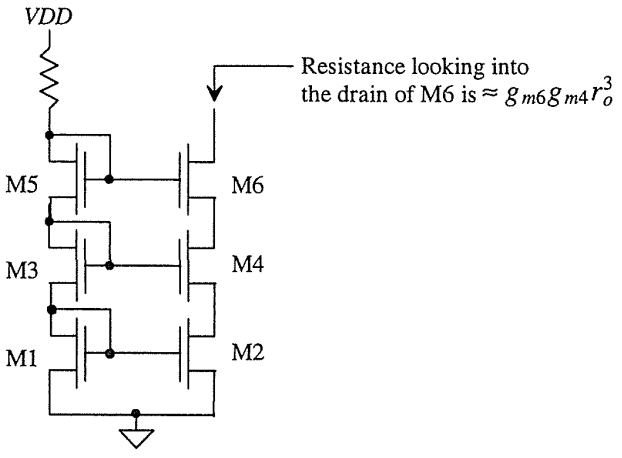



**Figure 20.9** The triple cascode current source.

Microchip Ex. 1055, p. 459
Microchip v. Aptiv
IPR2024-00494

**Example 20.4**

Repeat Ex. 20.1 using a double cascode current mirror.

The resistor value is calculated by

$$R = \frac{VDD - 2V_{GS} - VSS}{I_{D1}} = \frac{2.5 - 2.4 - (-2.5)}{10\ \mu A} = 260\ k\Omega$$

where we have assumed that $I_{D1} = I_{D2} = 10\ \mu A$ and $V_{GS} = 1.2$ V. The sizes of the devices are, from Ex. 20.1, $W = 15\ \mu m$ and $L = 5\ \mu m$. Calculation of the output resistance, $R_o$, begins by calculating the transconductance of the MOSFETs:

$$g_m = \sqrt{2\beta_n I_D} = \sqrt{2 \cdot 50\frac{\mu A}{V^2} \cdot \frac{15}{5} \cdot 10\ \mu A} = 55\ \frac{\mu A}{V}$$

and the output resistance of a single MOSFET is $r_o = 1/\lambda I_D = 1/(10\mu A)(0.06) = 1.67\ M\Omega$. The output resistance of the current source is then

$$R_o \approx g_m r_o^2 = 55\frac{\mu A}{V}(1.67\ M\Omega)^2 = 152\ M\Omega$$

The minimum voltage across the current source is $2\Delta V + V_{THN}$ or 1.57 V. SPICE simulation results are shown in Fig. Ex20.4 along with a schematic. Comparing these simulation results to those of Ex. 20.1, we see the increase in the output resistance. Notice that the body effect (increase in the threshold voltage of M3/M4) causes the currents to be less than the 10 μA we designed for. ■

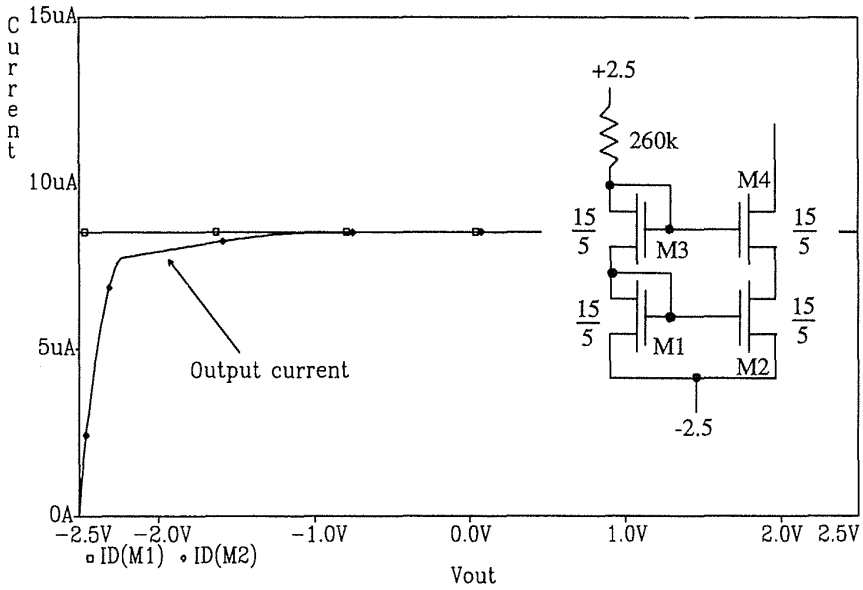



**Figure Ex20.4**

Microchip Ex. 1055, p. 460
Microchip v. Aptiv
IPR2024-00494

### 20.1.2 Sensitivity Analysis

Also of concern when designing current sources and sinks is the sensitivity of the output current to some input, such as the DC power supplies, or the value of a component, such as a resistor. The sensitivity of $I_o$ to *VDD*, for the simple current mirror in Fig. 20.1, can be defined by

$$S_{VDD}^{I_o} = \lim_{\Delta VDD \to 0} \frac{\frac{\Delta I_o}{I_o}}{\frac{\Delta VDD}{VDD}} = \frac{VDD}{I_o} \cdot \frac{\partial I_o}{\partial VDD} \tag{20.14}$$

The sensitivity of the output current to *VDD* is given, using Eq. (20.4), and assuming that $I_o = I_{D1}$ and the change in $V_{GS}$ with *VDD* is small, by

$$S_{VDD}^{I_o} \approx \frac{VDD}{I_o} \cdot \frac{1}{R} \tag{20.15}$$

This equation is useful in answering the question "If *VDD* changes by 10 percent how much will the output current change?" The percentage change in $I_o$ as a function of the percentage change in *VDD* is given by

$$\frac{\Delta I_o}{I_o} = S_{VDD}^{I_o} \cdot \frac{\Delta VDD}{VDD} \tag{20.16}$$

**Example 20.5**

Estimate the variation in $I_o$, for the current mirror of Ex. 20.1, for *VDD* changing from 2.4 to 2.6 V. Simulate the sensitivity using PSPICE. Connect the output of the current source, the drain of M2, to a battery of value -1.3 V (approximately the same potential as the drain of M1).

The sensitivity is given by Eq. (20.15) as

$$S_{VDD}^{I_o} = \frac{2.5}{10\mu A} \cdot \frac{1}{380k} = 0.658$$

The percentage change in *VDD* is 8 percent so that, using Eq. (20.16), the percentage change in the output current is 10.6 percent, that is,

$$\frac{\Delta I_o}{I_o} = 0.658 \cdot \frac{0.2}{2.5} = 0.053 = 5.3 \text{ percent}$$

For an $I_o$ = 10 μA the change in the output current is $\Delta I_o$ = 0.53 μA. In other words, when *VDD* = 2.4V, $I_o$ = 9.73 μA and when *VDD* = 2.6 V, $I_o$ = 10.27 μA. The PSPICE netlist used in this example is as follows.

```
Title
M1      1 1 2 2 CMOSNB  L=5u W=15u
M2      5 1 2 2 CMOSNB  L=5u W=15u
R1      3 1 380k  TC1=0.002
VDD     3 0     DC 2.5
Vdrain2 5 0     DC  -1.3
VSS     2 0     DC -2.5
```

Microchip Ex. 1055, p. 461
Microchip v. Aptiv
IPR2024-00494

```
.MODEL CMOSNB NMOS LEVEL=4
BSIM model parameters
.sens I(vdrain2)
.probe
.DC Vdrain2 -2.5 2.5 .01
.end
```

The output sensitivity results are

DC SENSITIVITIES OF OUTPUT I(vdrain2)

| ELEMENT NAME | ELEMENT VALUE | ELEMENT SENSITIVITY (AMPS/UNIT) | NORMALIZED SENSITIVITY (AMPS/PERCENT) |
|---|---|---|---|
| R1 | 3.800E+05 | 2.441E-11 | 9.277E-08 |
| VDD | 2.500E+00 | -2.465E-06 | -6.161E-08 |
| Vdrain2 | -1.200E+00 | -8.302E-07 | 9.962E-09 |
| VSS | -2.500E+00 | 3.295E-06 | -8.237E-08 |

The sensitivity, defined by PSPICE, is slightly different than previously discussed. For example, the value of the output current I(vdrain2) given above is:

$$\frac{\partial I_o}{\partial VDD} = \frac{\partial I(vdrain2)}{\partial VDD} = -2.465 \times 10^{-6} \ \frac{\text{A}}{\text{V}}$$

which is close to our approximation that

$$\frac{\partial I_o}{\partial VDD} \approx \frac{1}{R} = 2.632 \times 10^{-6} \ \frac{\text{A}}{\text{V}}$$

The sign difference is due to the direction of the current flow defined as positive by PSPICE. The normalized sensitivity of the output current to *VDD*, defined by PSPICE and shown in the fourth column of the PSPICE output file, is given by

$$VDD \cdot \frac{\partial I_o}{\partial VDD} \cdot \frac{1}{100} = \frac{\text{Amps}}{\%}$$

Performing a sensitivity analysis allows the designer to determine which components have the largest effect on the output variable. The sensitivity of the output current is highly dependent on the voltage across the current source, that is, the potential at the drain of M2. Computer simulations are almost a necessity in determining the sensitivity of an output variable to a circuit component in large circuits. ■

### 20.1.3 Temperature Analysis

The temperature coefficient of the simple current mirror is given by

$$TC(I_o) = \frac{1}{I_o} \cdot \frac{\partial I_o}{\partial T} = \frac{1}{T} \cdot S_T^{I_o} \tag{20.17}$$

The output current is given by Eqs. (20.3) and (20.4) as

Microchip Ex. 1055, p. 462
Microchip v. Aptiv
IPR2024-00494

$$I_o = \frac{W_2 L_1}{W_1 L_2} \cdot \frac{VDD - V_{GS} - VSS}{R} \tag{20.18}$$

where we have not assumed that $I_{D1} = I_{D2} = I_o$. If $I_{D1} = I_{D2}$, the ratio of the widths and lengths is unity in this equation (which was assumed for the sensitivity analysis of the previous section). The temperature coefficient is then given by

$$TC(I_o) = -\frac{1}{I_o} \cdot \left[ \frac{W_2 L_1}{W_1 L_2} \cdot \frac{1}{R} \frac{\partial V_{GS}}{\partial T} + \frac{I_o}{R} \cdot \frac{\partial R}{\partial T} \right] \tag{20.19}$$

From Ch. 7 we know that $\frac{1}{R}\frac{\partial R}{\partial T}$ is the temperature coefficient for a resistor, which, for the n-well resistor, is 2,000 ppm/°C. The gate-source voltage of M2 can be expressed by

$$V_{GS} = V_{THN} + \sqrt{\frac{I_o}{\beta_2/2}} = V_{THN} + \sqrt{\frac{W_2 L_1}{W_1 L_2} \cdot \frac{(VDD - V_{GS} - VSS)}{R \cdot KP(T)\frac{W_2}{2 \cdot L_2}}} \tag{20.20}$$

or, if $VDD - VSS >> V_{GS}$, then

$$\frac{\partial V_{GS}}{\partial T} = \frac{\partial V_{THN}}{\partial T} + \sqrt{\frac{2L_1}{W_1} \cdot \frac{(VDD - VSS)}{R \cdot KP(T)}} \cdot \left(-\frac{1}{2}\right) \cdot \left[ \frac{1}{KP(T)} \frac{\partial KP(T)}{\partial T} + \frac{1}{R} \frac{\partial R}{\partial T} \right] \tag{20.21}$$

The temperature dependence of the threshold voltage was given in Ch. 9. The change in threshold voltage with temperature is given by

$$\frac{\partial V_{THN}}{\partial T} = V_{THN} \cdot TCV_{THN} \tag{20.22}$$

where $TCV_{THN} \approx -3,000 \frac{ppm}{°C}$ or for $V_{THN} = 0.83$ V, the change in threshold voltage with temperature is approximately -2.4 mV/°C. The temperature dependence of the transconductance parameter was modeled, again from Ch. 9, by

$$KP(T) = KP(T_0) \cdot \left(\frac{T}{T_0}\right)^{-1.5} \text{ (T is in °K)} \tag{20.23}$$

The change in the transconductance parameter with temperature is given by

$$\frac{\partial KP(T)}{\partial T} = KP(T_0)(-1.5)\left(\frac{T}{T_0}\right)^{-2.5} \cdot \frac{1}{T_0} \rightarrow \frac{1}{KP(T)} \cdot \frac{\partial KP(T)}{\partial T} = \frac{-1.5}{T} \tag{20.24}$$

The change in gate-source voltage, given by Eq. (20.21), can now be written as

$$\frac{\partial V_{GS}}{\partial T} = V_{THN} \cdot TCV_{TH} - \left(\frac{1}{2}\right) \cdot \sqrt{\frac{2L_1}{W_1} \cdot \frac{VDD - VSS}{R \cdot KP(T)}} \left[ \frac{1}{R} \frac{\partial R}{\partial T} - \frac{1.5}{T} \right] \tag{20.25}$$

or, substituting into Eq. (20.21),

$$TC(I_o) = -\frac{1}{I_o} \cdot \frac{W_2 L_2}{W_1 L_1} \left[ \frac{V_{THN} \cdot TCV_{TH}}{R} - \frac{1}{R} \sqrt{\frac{L_1}{W_1} \frac{VDD - VSS}{2RKP(T)}} \left[ \frac{1}{R} \frac{\partial R}{\partial T} - \frac{1.5}{T} \right] \right] - \frac{1}{R} \frac{\partial R}{\partial T} \tag{20.26}$$

Microchip Ex. 1055, p. 463
Microchip v. Aptiv
IPR2024-00494

This result will be applied in the following example.

**Example 20.6**

Determine the temperature coefficient of the simple current mirror of Ex. 20.1. Compare the temperature coefficient determined with a PSPICE simulation.

The temperature coefficient of the simple current mirror around the temperature 300 °K is determined using Eq. (20.26) by

$$TC(I_o) = \frac{-1}{10\mu}\left[\overbrace{\frac{-0.0024}{380\times10^3}}^{-6.3\times10^{-9}} \overbrace{-\frac{1}{380\times10^3}\sqrt{\frac{5}{380\times10^3\cdot 2\cdot 50\times10^{-6}}}\left(0.002-\frac{1.5}{300}\right)}^{-2.86\times10^{-9}}\right] - .002$$

or the temperature coefficient of the output current at 27 °C is

$$TC(I_o) \approx -1,656\ ppm/°C = -0.17\ \%change/°C$$

The output current as a function of temperature is given by

$$I_o(T) = I_o(T_0)(1 + TC(I_o)(T - T_0)) = I(T_0)[1 - 0.0017(T - 27)]$$

The PSPICE simulation results, with TC1 of the resistor set to 0.002, are shown in Fig. 20.10. The hand calculations are close to the simulation results. Note that the temperature coefficient of the current source is not a constant number, but is a function of the operating temperature. This is apparent by the differing distances between the curves of Fig. 20.10. ■

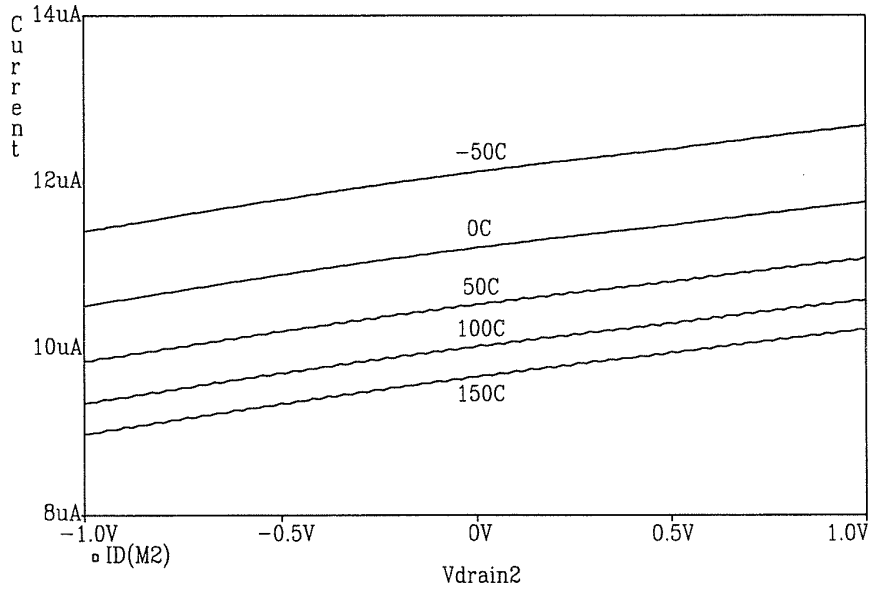



**Figure 20.10** Temperature characteristics of the simple current mirror of Ex. 20.1. Notice that the voltage on the drain of M2 is swept from -1 V to + 1 V.

Microchip Ex. 1055, p. 464
Microchip v. Aptiv
IPR2024-00494

An important question we can ask is "Can we design a simple current mirror with a TC of zero?" Rearranging Eq. (20.26) for a TC($I_o$) of zero gives

$$I_o R = \frac{W_2 L_2}{W_1 L_1}\left[ -V_{THN} \cdot TCV_{TH} + \sqrt{\frac{L_1}{W_1}\frac{VDD - VSS}{2RKP(T)}}\left[\frac{1}{R}\frac{\partial R}{\partial T} - \frac{1.5}{T}\right]\right] \cdot \frac{1}{\frac{1}{R}\frac{\partial R}{\partial T}} \quad (20.27)$$

This equation relates the output current and the resistor value needed for a current mirror with a TC of zero, given the temperature characteristics of the MOSFET and the resistor.

### 20.1.4 Transient Response

Another important characteristic of an electronic circuit is how well the circuit performs under AC or varying input signals. Figure 20.11 shows a typical setup for determining the transient or step response of the simple current mirror. The output current of the mirror is given by $I_o = \frac{\beta_2}{2}(V_{GS} - V_{THN})^2$ and is directly dependent on $V_{GS}$. We are trying to answer the question, "If the voltage across the current source changes abruptly, how does this affect $V_{GS}$ and thus the output current?" The gate-source voltage in this test setup is given by

$$V_{GS} = V_{step} \cdot \frac{Z_{GS}}{Z_{GS} + \frac{1}{j\omega C_{gd2}}} \quad (20.28)$$

where

$$Z_{GS} = \frac{R \| \frac{1}{g_{m1}}}{1 + j\omega(C_{gs1} + C_{gs2})R \| \frac{1}{g_{m1}}} \quad (20.29)$$

Combining these two equations gives the following

$$V_{GS} = V_{step} \cdot \frac{j\omega\left(R \| \frac{1}{g_{m1}}\right)C_{gd2}}{1 + j\omega(C_{gs1} + C_{gs2} + C_{gd2})R \| \frac{1}{g_{m1}}} \quad (20.30)$$

Assuming the step rises very quickly, the peak voltage change, $\Delta V_{GS}$, is given by

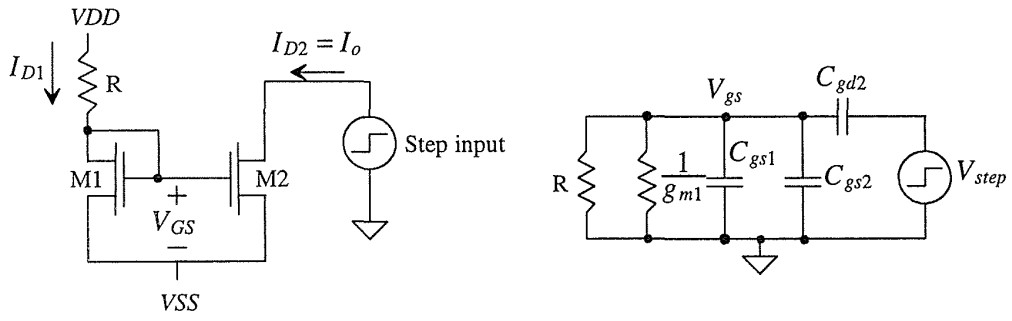



**Figure 20.11** Transient response of the simple current mirror.

Microchip Ex. 1055, p. 465
Microchip v. Aptiv
IPR2024-00494