

Analysis and Design of Data Converters

Master the art of data converter design with this definitive textbook, a detailed and accessible guide ideal for students and practicing engineers.

Key Features

- Razavi's distinctive and intuitive pedagogical approach, building up from elementary components to complex systems.
- Step-by-step transistor-level designs and simulations offer a practical, hands-on understanding of key design concepts.
- Comprehensive coverage of essential topics including sampling circuits, comparator design, digital-to-analog converters, flash topologies, SAR and pipelined architectures, time-interleaved converters, and oversampling systems.
- Over 250 examples pose thought-provoking questions, reinforcing core concepts and helping students develop confidence.
- Over 350 end-of-chapter homework problems to test student understanding, with solutions available for course instructors.

Developed by leading author Behzad Razavi, and addressing all the principles and design concepts essential to today's engineers, this is the ideal text for senior undergraduate and graduate-level students and professional engineers who aspire to excel in data converter analysis and design.

Behzad Razavi is a Professor of Electrical Engineering at the University of California, Los Angeles. He has received numerous teaching and education awards, has served as an IEEE Distinguished Lecturer, and is a member of the US National Academy of Engineering and a Fellow of the IEEE. His previous textbooks include *Fundamentals of Microelectronics*, *RF Microelectronics*, *Design of Analog CMOS Integrated Circuits*, and *Design of CMOS Phase-Locked Loops*.

“It is always a great pleasure to read Professor Razavi’s books. This one is no exception. It is a timely update of his prior book on data converters. It is a must-read for anyone who is interested in data converter design.”

Nan Sun, Tsinghua University

“Writing a book on data converters that covers all fundamental principles, provides practical design insights, and can remain relevant in this ever changing field is very difficult. Professor Razavi did it.”

Dante Muratore, Delft University of Technology

“This book stands out as a definitive resource on data converters. Professor Razavi strikes the perfect balance between intuitive explanations of complex concepts and the mathematical rigor necessary for a deep technical understanding. Whether you are a student seeking clarity or a professional looking for a reliable reference, this book is an indispensable guide.”

Pavan Kumar Hanumolu, University of Illinois Urbana–Champaign

Analysis and Design of Data Converters

BEHZAD RAZAVI

University of California, Los Angeles





Shaftesbury Road, Cambridge CB2 8EA, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre,
New Delhi – 110025, India

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment,
a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of
education, learning and research at the highest international levels of excellence.

www.cambridge.org

Information on this title: www.cambridge.org/highereducation/isbn/9781009602235

DOI: 10.1017/9781009602266

© Behzad Razavi 2025

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

When citing this work, please include a reference to the DOI 10.1017/9781009602266

First published 2025

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

A Cataloging-in-Publication data record for this book is available from the Library of Congress

ISBN 978-1-009-60223-5 Hardback

Additional resources for this publication at www.cambridge.org/dataconverters

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

To the memory of my brother Hojjat

Contents

Preface	xv
Acknowledgments	xvi
About the Author	xviii
1 Introduction to Data Conversion	1
1.1 Analog-to-Digital Conversion	1
1.2 Digital-to-Analog Conversion	4
2 Basic Circuit Concepts	6
2.1 Charge Conservation	6
2.2 Charge Sharing	8
2.3 Integrated Capacitors	10
2.4 Offset, Gain Error, and Nonlinearity	13
3 Sampling Circuits	20
3.1 Basic Sampling Concepts	20
3.1.1 Ideal Sampler	20
3.1.2 Zero-Order-Hold Sampler	21
3.1.3 Track-and-Hold Circuit	23
3.1.4 Choice of Sampler Time Constant	24
3.2 Basic Sampling Circuits	25
3.3 Distortion Due to Switch Resistance	29
3.4 Complementary Sampling Switches	31
3.5 Other Switch Imperfections	35
3.5.1 Review of MOS Capacitances	35
3.5.2 Clock Feedthrough	35
3.5.3 Channel Charge Injection	36
3.5.4 kT/C Noise	38
3.6 Differential Sampling	40
3.7 Use of Dummy Switches	42
3.8 Bootstrapped Sampling Circuit	43
3.8.1 Preliminary Ideas	43
3.8.2 Complete Bootstrapped Circuit	45
3.8.3 Bootstrapping Imperfections	49
3.9 Performance Parameters	53
3.9.1 Speed	53
3.9.2 Linearity	54
3.9.3 Static and Dynamic Nonlinearity	55
3.10 Effect of Clock Jitter	55
4 Design Study of a Sampling Circuit	60
4.1 General Considerations	60
4.2 Design Procedure	61
4.3 Simulation Issues	61
4.4 Track-Mode Distortion	62
4.5 Basic Sampler	63
4.6 Sampler with MOS Switches	63
4.7 Device Stress Issues	66
	vii

4.8	Input Current Issues	67
5	Comparator Design Principles	70
5.1	A Brief Note on Offsets	70
5.2	Basic Concepts	72
5.3	Amplification by Positive Feedback	74
5.4	Controlled Positive Feedback	80
5.5	A Simple Comparator	81
5.6	A Better Comparator	83
5.7	Comparator Design Parameters	84
5.7.1	Input Offset	84
5.7.2	Speed	85
5.7.3	Electronic Noise	87
5.7.4	Kickback Noise	88
5.8	The StrongArm Comparator	89
5.8.1	Differential Pairs with Capacitive Loads	89
5.8.2	Addition of Positive Feedback	91
5.8.3	Complete StrongArm Comparator	94
5.8.4	Basic Properties of StrongArm Comparator	96
5.8.5	The Need for an RS Latch	98
6	Advanced Comparator Concepts	102
6.1	Analysis of Transistor Mismatches	102
6.1.1	Basic Concepts	102
6.1.2	Dependence of Mismatches on Channel Area	105
6.2	Metastability	107
6.2.1	Metastability in Current-Steering Comparators	107
6.2.2	Metastability in StrongArm Comparator	111
6.3	Noise in StrongArm Comparator	112
6.3.1	Noise in Time Domain	113
6.3.2	Noise in Amplification Mode	115
6.4	Simulation of Comparator Noise	117
6.5	Offset Cancellation	119
6.5.1	Analog Offset Cancellation	119
6.5.2	Digital Offset Cancellation	123
6.5.3	Offset Cancellation in StrongArm Comparator	125
7	Design Study of a Comparator	129
7.1	General Considerations	129
7.2	Choice of Device Dimensions	130
7.3	Basic Waveforms	131
7.4	Offset and Speed Optimization	132
7.5	Addition of RS Latch	133
7.6	Metastability	134
7.7	Input-Referred Noise	137
7.8	Kickback Noise	137
8	General DAC Concepts	142
8.1	Basic Concepts	142
8.2	DAC Performance Parameters	143
8.2.1	Static Parameters	144
8.2.2	Dynamic Parameters	150

8.3	Digital Codes in DACs	153
8.4	Clocking in DACs	154
9	Resistor-Ladder DACs	158
9.1	Basic Structures	158
9.2	Speed Limitations	160
9.3	Differential RDACs	162
9.4	Effect of Gradients	163
9.5	Effect of Random Mismatches	165
9.6	Problem of Reference Bounce	168
10	Current-Steering DACs	174
10.1	Basic Topologies	174
10.2	Basic Matching Considerations	177
10.2.1	Deterministic Mismatches	178
10.2.2	Random Mismatches	179
10.3	Binary-Weighted and Thermometric IDACs	180
10.4	R-2R IDACs	183
10.5	Segmented IDACs	184
10.6	Nonlinearity Due to Output Impedance of Current Sources	186
10.7	Transistor Sizing	192
10.8	Matrix Architecture	192
10.8.1	Basic Idea	192
10.8.2	Effect of Gradients	194
10.8.3	Signal and Power Distribution	196
10.9	Speed and Reference Bounce Issues	200
10.10	Dynamic Errors in IDACs	202
10.10.1	Output Impedance at High Frequencies	202
10.10.2	Nonlinear Memory Effect	206
11	Capacitor DACs	210
11.1	Basic Topologies	210
11.2	Binary-Weighted and Thermometric CDACs	212
11.3	Speed Limitations	214
11.4	Effect of Random Mismatches	216
11.5	Capacitor Structures	217
11.6	Problem of Reference Bounce	222
11.7	Split CDACs	224
11.8	Choice of Unit Capacitance Value	229
11.9	Matrix Architecture for CDACs	231
12	Introduction to Nyquist-Rate ADCs	236
12.1	General Considerations	236
12.2	A First Look at Quantization	237
12.2.1	Basic Concepts	237
12.2.2	Formulation of Quantization Error	238
12.3	ADC Performance Parameters	241
12.3.1	Offset	241
12.3.2	Gain Error	241
12.3.3	Differential Nonlinearity	242
12.3.4	Integral Nonlinearity	243
12.3.5	SNDR and ENOB	246

12.3.6	SFDR	248
12.3.7	Static and Dynamic Performance	248
12.3.8	Figure of Merit	248
12.4	Nyquist-Rate ADC Architectures	249
13	Flash ADCs	251
13.1	Basic Flash ADC	251
13.2	Sampling in Flash ADCs	253
13.3	Thermometer-to-Binary Conversion	254
13.3.1	ROM Decoder	254
13.3.2	MUX Decoders	256
13.3.3	ONEs-Counter Decoders	257
13.4	Nonidealities	260
13.4.1	Comparator Offset	260
13.4.2	Input Capacitance	262
13.4.3	Comparator Metastability	262
13.4.4	Kickback Noise	265
13.4.5	Input-Dependent Effects	269
13.5	Bubble Correction	272
13.6	Fully-Differential Flash ADCs	273
13.6.1	Fully-Differential Comparators	274
13.6.2	U-Shaped Flash Architecture	280
13.7	Flash ADC Variants	281
14	SAR ADC Fundamentals	284
14.1	General Concepts	284
14.1.1	Quantization by Binary Search	284
14.1.2	Basic SAR Structure	284
14.1.3	Concept of Time Trellis	287
14.1.4	Simple SAR Logic	287
14.1.5	Computing V_{in} from D_{out}	289
14.2	Charge Redistribution SAR ADCs	290
14.2.1	SAR ADCs with CDACs	290
14.2.2	Basic Top-Plate Sampling	290
14.2.3	Single-Ended Top-Plate-Sampling ADC	293
14.2.4	Differential Top-Plate-Sampling ADC	298
14.2.5	Bottom-Plate-Sampling ADC	299
14.2.6	Differential Bottom-Plate-Sampling ADC	303
14.3	Clock Generation	307
15	SAR ADC Nonidealities	310
15.1	Sampling Nonidealities	310
15.1.1	Switch Resistance	310
15.1.2	Switch Junction Capacitance	312
15.2	DAC Nonidealities	316
15.2.1	DAC kT/C Noise	316
15.2.2	DAC Offset and Gain Error	317
15.2.3	DAC Nonlinearity	318
15.2.4	Reference Bounce	319
15.2.5	Reference Noise	320
15.3	Comparator Nonidealities	321
15.4	Speed Limitations	322

15.4.1	Comparator Response	322
15.4.2	DAC Response	323
15.4.3	Tapering and Minimum Delay	326
16	Advanced SAR ADCs	330
16.1	Speed Improvement	330
16.1.1	Asynchronous Operation	330
16.1.2	SAR ADCs with Multibit Quantizers	334
16.1.3	Hybrid Flash/SAR ADCs	338
16.2	DAC Complexity Reduction	346
16.2.1	Monotonic Switching	346
16.2.2	Three-Level Bottom-Plate Sampling	348
17	Fundamentals of Pipelined ADCs	353
17.1	Binary Search Revisited	353
17.2	Basic Pipelining	355
17.3	Brief Review of Op Amp Models	357
17.4	Precision Multiply-by-2 Stages	361
17.4.1	Non-Flip-Around Amplifier	362
17.4.2	Flip-Around Amplifier	364
17.5	Multiplying DACs	366
17.5.1	Non-Flip-Around MDAC	366
17.5.2	Flip-Around MDAC	368
17.6	Basic Pipelined ADC	370
17.7	Concept of Residue	373
17.8	Multibit Pipelined Stages	376
17.9	Comparison of SAR and Pipelined ADCs	380
18	Nonidealities in Pipelined ADCs	383
18.1	General Considerations	383
18.1.1	MDAC Nonidealities	383
18.1.2	Residue Manipulations	383
18.2	Effect of Comparator Offset	386
18.2.1	Offset in the First Stage	386
18.2.2	Offset in Second Stage	386
18.3	Effect of Op-Amp Offset	391
18.4	Effect of Switch Charge Injection	392
18.4.1	Charge Injection in Non-Flip-Around MDAC	393
18.4.2	Charge Injection in Flip-Around MDAC	394
18.5	kT/C Noise	396
18.5.1	Non-Flip-Around MDAC	396
18.5.2	Flip-Around MDAC	396
18.6	Effect of Op-Amp Nonidealities	396
18.6.1	Finite Output Impedance and Bandwidth	396
18.6.2	Finite Op-Amp Gain	401
18.6.3	Op-Amp Noise	407
18.7	Effect of Capacitor Mismatch	412
18.8	Effect of the Reference Bounce	414
18.9	Timing Mismatch	415
19	Advanced SAR and Pipelining Techniques	418
19.1	Overlap in Hybrid Flash/SAR ADCs	418
19.1.1	Comparator Offset Revisited	418

19.1.2	Overlap and Redundancy	419
19.1.3	Effect of Timing Mismatch	422
19.1.4	Residue in Hybrid Flash/SAR ADCs	422
19.1.5	Effect of SAR Offset	423
19.2	1.5-Bit/Stage Pipelined ADCs	425
19.2.1	Basic Idea	425
19.2.2	Residue Behavior	427
19.2.3	Cascade Example	428
19.2.4	Omission of Resistor Ladder	430
19.3	Open-Loop Residue Amplification	431
19.4	Hybrid SAR/Pipelined ADCs	432
19.4.1	Hybrid Example 1	432
19.4.2	Hybrid Example 2	433
19.4.3	Hybrid Example 3	434
19.4.4	Hybrid Example 4	436
20	Interleaved ADCs	439
20.1	General Considerations	439
20.1.1	Basic Idea	439
20.1.2	Advantages of Interleaving	442
20.1.3	Disadvantages of Interleaving	443
20.2	Effect of Mismatches	443
20.2.1	Offset Mismatch	444
20.2.2	Gain Mismatch	446
20.2.3	Phase Mismatch	447
20.2.4	Bandwidth Mismatch	452
20.3	Charge Sharing and Interchannel Crosstalk	453
20.4	Interleaving Techniques	455
20.4.1	Two-Rank Interleaving	455
20.4.2	Reduction of Interchannel Crosstalk	456
20.5	Interchannel Mismatch Calibration	457
20.5.1	General Considerations	457
20.5.2	Foreground and Background Calibration	460
20.5.3	Offset Calibration	460
20.5.4	Gain Calibration	462
20.5.5	Phase Calibration	463
20.6	Clock Generation for Interleaved ADCs	471
21	Introduction to Oversampling ADCs	479
21.1	Quantization Noise Revisited	479
21.2	Basic Concepts	483
21.2.1	Q-Noise Model	483
21.2.2	Noise in a Feedback System	483
21.2.3	Simple Noise-Shaping ADC	485
21.2.4	Noise Shaping in Time Domain	487
21.3	Discrete-Time $\Delta\Sigma$ Modulator	488
21.3.1	Discrete-Time Integrator	488
21.3.2	First-Order $\Delta\Sigma$ Loop	492
21.3.3	Q-Noise Calculation	497
21.4	Continuous-Time First-Order $\Delta\Sigma$ Modulator	498
21.5	Second-Order $\Delta\Sigma$ Modulators	499
21.5.1	Basic Idea	499

21.5.2	Transfer Functions	499
21.5.3	Total Quantization Noise	502
21.6	Advantages of Oversampling	503
21.6.1	General Advantages	503
21.6.2	CT Modulator Advantages	503
21.7	Disadvantages of Oversampling	504
21.8	Performance Metrics	504
21.9	Properties of 1-Bit Quantizers	505
22	Basic Design of $\Delta\Sigma$ Modulators	509
22.1	First-Order DT Loop Design	509
22.1.1	Improved DT Integrator	509
22.1.2	First-Order Loop with 1-Bit Quantizer	512
22.2	Second-Order DT Loop with 1-Bit Quantizer	518
22.2.1	SNR Calculation	520
22.2.2	Additional Insights	521
22.3	First-Order Continuous-Time Loop Design	522
22.3.1	Basic Operation	522
22.3.2	Loop Waveforms	523
22.3.3	SNR Calculation	527
22.4	Second-Order Continuous-Time Loop Design	529
22.4.1	SNR Calculation	530
22.4.2	Integrator Output Swings	531
22.5	Generation of Nonoverlapping Clocks	531
23	Nonidealities in $\Delta\Sigma$ Modulators	534
23.1	DT Modulator Nonidealities	534
23.1.1	Capacitor Mismatch	534
23.1.2	kT/C Noise	535
23.1.3	Op-Amp Noise	537
23.1.4	Op-Amp Gain	538
23.1.5	Op-Amp Offset	540
23.1.6	Op-Amp Speed	542
23.1.7	Comparator Offset and Noise	545
23.1.8	Reference Settling	545
23.1.9	Clock Jitter	546
23.2	Problem of Idle Tones	547
23.3	Noise Folding Due to Nonlinearity	548
23.4	CT Modulator Nonidealities	550
23.4.1	Slewing at Op Amp Output	550
23.4.2	Noise Sources	551
23.4.3	Resistive Load at Op Amp Output	553
23.4.4	Clock Jitter	553
23.4.5	Comparator Delay	555
23.5	Stability Considerations	557
24	Advanced Oversampling ADCs	559
24.1	Cascaded Modulators	559
24.1.1	Basic Idea	559
24.1.2	Higher-Order Cascades	562
24.1.3	Problem of Gain Error	563
24.1.4	Remedies for Gain Mismatch	565

24.2 $\Delta\Sigma$ Modulators with Multibit Quantizers 567

 24.2.1 Basic Idea 567

 24.2.2 Problem of DAC Nonlinearity 568

 24.2.3 Data-Weighted Averaging 569

24.3 Modulators with Feedforward 573

24.4 Decimation Filtering 575

 24.4.1 Basic Idea 575

 24.4.2 Counter-Based Design 579

Index 582

Preface

Data converters are an integral part of most electronic systems and play a key role in communications and signal processing. Examples include wireless and wireline transceivers, imaging applications, audio and video processing, and instrumentation. Despite decades of development, data converters continue to present new challenges as we push for higher speeds and lower power consumption.

Since I published my first book, *Principles of Data Conversion System Design*, in 1995, the field of data converters has grown and evolved dramatically. Owing to extensive work by industry and academia over the past 30 years, many new design techniques have been introduced at both the circuit level and the architecture level. Moreover, new analysis techniques have been developed that offer deeper insights. A completely new book was therefore due.

This textbook addresses the need for a detailed, accessible treatment of data converters that can serve both students and practicing engineers. Aiming for classroom adoption and/or self-study, the book offers hundreds of analysis examples and end-of-chapter problems. In addition, it advocates a bottom-up synthesis mentality by (a) describing how to build a complex system from elementary components such that the role of each component is appreciated and (b) providing step-by-step transistor-level designs and simulations.

We begin with fundamental building blocks such as sampling circuits, comparators, and digital-to-analog converters. Next, we deal with Nyquist-rate analog-to-digital converters (ADCs), focusing on flash, successive-approximation, pipelined, and interleaved architectures. We then turn to oversampling converters, introduce the concept of noise shaping, study the basic design of such architectures, and analyze their imperfections. We also present more advanced topics such as cascaded topologies and data-weighted averaging.

As with my previous books, I follow four pedagogical principles in writing. (1) I explain why the reader needs to know the concepts to be studied. (2) I place myself in the reader's shoes and predict the questions and confusions that the reader may face while seeing these concepts for the first time. (3) I pretend to know only as much as the reader and begin with the core of the concept so that I can hook the reader - even at the cost of rigor. (4) I then add the necessary details so as to create a complete and rigorous picture.

This book is accompanied by a set of slides and a solutions manual.

Behzad Razavi

Acknowledgments

I am indebted to numerous individuals who have assisted with the publication of this book. First, a band of reviewers:

Pietro Andreani, Lund University	Leonardo Papale, Eindhoven University of Technology
Mostafa Ayes, University of Southern California	Sharvil Patil, Analog Devices
Ivor Chan, University of Macau	Kevin Pelzers, Pelzers Integrated Circuits
Dong-Jin Chang, Chungnam National University	Michael Pietzko, Universitat Ulm
Yujie Chen, UCLA	Chithira Ravi, Synopsys
Wood Chiang, Brigham Young University	Vishnu Ravinuthula, Texas Instruments
Delong Cui, Broadcom	Stijn Ringeling, Eindhoven University of Technology
Denis Daly, Apple	Seung-Tak Ryu, KAIST
Mohamed Darwish, Apple	Bibhu Sahoo, University at Buffalo
Manar El-Chammas, Omni Design Technologies	Vineeth Sarma, Analog Devices
Ahmed Elshater, Mediatek	Remco Schalk, Eindhoven University of Technology
Brett Forejt, Texas Instruments	Yuting Shen, NXP
Miguel Gandara, Mediatek	Hajime Shibata, Analog Devices
Manideep Gande, Nvidia	Hyun Woo Shin, KAIST
Yuji Gendai, THine Electronics	Kwan Hoon Song, KAIST
Pieter Harpe, Eindhoven University of Technology	Shiyu Su, University of Waterloo
Xiyu He, Tsinghua University	Nan Sun, Tsinghua University
Benjamin Hersherberg, Intel	Charlie Tahar, KAIST
Matias Jara, Broadcom	Yunsong Tao, Tsinghua University
Paul Josef Kaesser, Universitat Ulm	Jonathan Ungethum, Universitat Ulm
John Kauffman, Universitat Ulm	Frank Van der Goes, Broadcom
Guansheng Li, Broadcom	Stan van der Ven, IMEC
Hanyue Li, IMEC	Hongrui Wang, Tsinghua University
Juzheng Liu, University of Southern California	Yujie Wang, UCLA
Kent Edrian Lozada, KAIST	Jan Westra, Broadcom
Yanquan Luo, Tsinghua University	David-Peter Wiens, Universitat Ulm
Raymond Mabilangan, KAIST	Haoming Xin, IMEC
Ahmed Mahmoud, Universitat Ulm	Bishoy Zaky, Universitat Ulm
Jan Mulder, Broadcom	Minglei Zhang, University of Macau
Ali Nazemi, Broadcom	Qiaochu Zhang, University of Virginia
Dong-Ryeol Oh, Jeju National University	Meiyi Zhou, IMEC
Kyle van Oosterhout, Eindhoven University of Technology	Yan Zhu, University of Macau
Henry Park, Mediatek	
Kun Woo Park, KAIST	

My special thanks go to Pietro Andreani, who heroically reviewed the entire book and caught many mistakes. Grazie, Piero! In addition, Mike Chen, Pieter Harpe, Harry Lee, Rui Martins, Afshin Momtaz, Maurits Ortmanns, Richard Schreier, and Nan Sun have contributed to this book.

I also wish to thank the staff at Cambridge University Press for their support during the production of the book, particularly, Elizabeth Horne, Sarah Lewis, and Rachel Norridge.

My wife, Angelina, typed the entire book. I am very grateful to her.

Behzad Razavi

About the Author

Behzad Razavi received the BSEE degree from Sharif University of Technology in 1985 and the MSEE and PhDEE degrees from Stanford University in 1988 and 1992, respectively. He was with AT&T Bell Laboratories and Hewlett-Packard Laboratories until 1996. Since 1996, he has been Associate Professor and subsequently Professor of electrical engineering at University of California, Los Angeles. His current research includes wireless and wireline transceivers and data converters.

Professor Razavi was an Adjunct Professor at Princeton University from 1992 to 1994, and at Stanford University in 1995. He served on the Technical Program Committees of the International Solid-State Circuits Conference (ISSCC) from 1993 to 2002 and VLSI Circuits Symposium from 1998 to 2002. He has also served as Guest Editor and Associate Editor of the *IEEE Journal of Solid-State Circuits*, *IEEE Transactions on Circuits and Systems*, and *International Journal of High Speed Electronics*. He presently serves as the Editor-in-Chief of the *IEEE Solid-State Circuits Letters*.

Professor Razavi received the Beatrice Winner Award for Editorial Excellence at the 1994 ISSCC, the best paper award at the 1994 European Solid-State Circuits Conference, the best panel award at the 1995 and 1997 ISSCC, the TRW Innovative Teaching Award in 1997, the best paper award at the IEEE Custom Integrated Circuits Conference in 1998, and the McGraw-Hill First Edition of the Year Award in 2001. He was the co-recipient of both the Jack Kilby Outstanding Student Paper Award and the Beatrice Winner Award for Editorial Excellence at the 2001 ISSCC. He received the Lockheed Martin Excellence in Teaching Award in 2006, the UCLA Faculty Senate Teaching Award in 2007, and the CICC Best Invited Paper Award in 2009 and in 2012. He was the co-recipient of the 2012 and the 2015 VLSI Circuits Symposium Best Student Paper Awards and the 2013 CICC Best Paper Award. He was also recognized as a top author in the 50-year and 75-year history of ISSCC. He received the 2012 Donald Pederson Award in Solid-State Circuits. He was also the recipient of the American Society for Engineering Education PSW Teaching Award in 2014. Professor Razavi is a member of the US National Academy of Engineering and a fellow of the National Academy of Inventors. He received the 2017 IEEE CAS John Choma Education Award.

Professor Razavi has served as an IEEE Distinguished Lecturer and is a Fellow of IEEE. He is the author of *Principles of Data Conversion System Design* (IEEE Press, 1995), *RF Microelectronics* (Prentice Hall, 1998, 2012) (translated to Chinese, Japanese, and Korean), *Design of Analog CMOS Integrated Circuits* (McGraw-Hill, 2001, 2016) (translated to Chinese, Japanese, and Korean), *Design of Integrated Circuits for Optical Communications* (McGraw-Hill, 2003, Wiley, 2012), *Fundamentals of Microelectronics* (Wiley, 2006) (translated to Korean, Portuguese, and Turkish), *Design of CMOS Phase-Locked Loops* (Cambridge, 2020), and the editor of *Monolithic Phase-Locked Loops and Clock Recovery Circuits* (IEEE Press, 1996), and *Phase-Locking in High-Performance Systems* (IEEE Press, 2003).

Introduction to Data Conversion

Data conversion refers to the transformation of analog signals to their digital counterparts (“analog-to-digital” conversion) or vice versa (“digital-to-analog” conversion). In this chapter, we begin with the former, present some of its applications, and describe its basic implementation. We then repeat these steps for the latter.

1.1 Analog-to-Digital Conversion

Naturally-occurring signals are found in analog form. Our voice, an image captured by the pixels in a digital camera, and radiations received from far-away galaxies exemplify such signals. In most systems, we wish to process, carry, and store this information in digital form; hence the need for analog-to-digital converters (ADCs).

As an example, suppose a farmer facing severe drought uses a large number of sensors in an orchard that measure the water received by each tree and wirelessly transmit the information to the farmer’s phone. It is much simpler in this case to digitize the sensors’ outputs and deal with only ONEs and ZEROS beyond that point. Of course, the power consumed by each “node” (the sensor box) must be minimized because the farmer is not inclined to change the batteries frequently. The cost plays a critical role here as well.

There are also situations in which *digital* signals must be treated as analog information and must be *digitized*! Consider, for example, a USB cable connecting a high-resolution video camera to a computer (Fig. 1.1). The camera applies the image to an ADC and a digital signal processor (DSP) for various operations, e.g., automatic focus or brightness adjustment. The DSP binary output, $x(t)$, now travels over the cable, experiencing its limited bandwidth and emerging severely “distorted” as $y(t)$. Since it is difficult to distinguish the ONEs and ZEROS in this waveform, we may decide to digitize it, compensate for the cable’s loss, and detect the bits reliably in the digital domain.

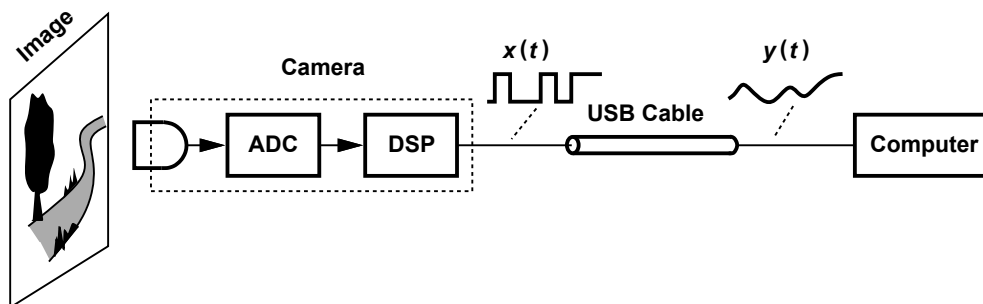


Figure 1.1 Distortion of binary data in a USB cable.

Figure 1.2 shows a conceptual ADC, where $x(t)$ is generally a “continuous-time” and “continuous-amplitude” signal. That is, it has a value at every point in time and the value can be any real (or complex) number between an upper and a lower bound. The output of a microphone is an example of such a signal. The ADC (binary) output, on the other hand, occurs only at discrete points in time, T_S , $2T_S$, etc., and assumes *discrete* amplitude values because of the finite number of bits available to represent the level. For example, a 4-bit ADC provides a binary output, $D_4D_3D_2D_1$, ranging from 0000 to 1111, and hence offers only 16 amplitude levels. We say an ADC converts a continuous-time, continuous-amplitude input to a discrete-time, discrete-amplitude output.

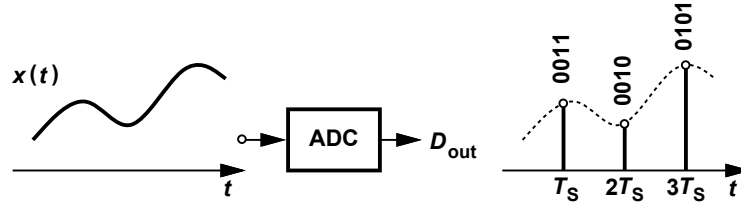


Figure 1.2 Conceptual operation of an ADC.

Example 1.1

Why should the digital output of an ADC appear only at discrete points in time?

Solution

From one perspective, we recognize that the DSP following the ADC can operate up to some speed and can therefore accept only up to a certain number of ADC output samples per second. But, more fundamentally, to represent the information carried by $x(t)$ in Fig. 1.2, the ADC need only provide a certain number of samples per second. Nyquist’s sampling theorem tells us that the sampling rate, $f_S = 1/T_S$, must be only slightly greater than twice the bandwidth of $x(t)$ to avoid aliasing. That is, it’s not necessary to sample the signal any faster.

Example 1.2

Must the digital output samples in Fig. 1.2 appear at integer multiples of T_S ? This scheme is called “uniform sampling.”

Solution

In general, sampling need not be uniform. Depending on how “busy” the signal is, we can adjust the sampling rate. For example, when a microphone output is quiet, the ADC sampling frequency can be lowered. Nonetheless, uniform sampling is by far the most common.

From these thoughts emerges the basic implementation depicted in Fig. 1.3. Here, a sampler driven by a clock frequency of f_S takes uniform samples of $x(t)$, generating $y(t)$. The latter is a discrete-time but continuous-amplitude signal (why?). Next, $y(t)$ is applied to a digitizer, more precisely called a “quantizer,” which approximates the input analog values by a finite number of levels. The following example illustrates the concept of (amplitude) quantization.

Example 1.3

A 2-bit ADC receives a voltage varying from 0 V to 400 mV. Sketch the input-output characteristic of the circuit.

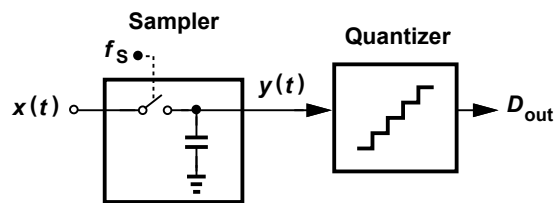


Figure 1.3 Sampling and quantization as fundamental operations for A/D conversion.

Solution

As shown in Fig. 1.4, we first identify the input range on the horizontal axis and divide it by $2^2 = 4$. For $0 < V_{in} < 100$ mV, we have $D_{out} = 00$. As V_{in} exceeds 100 mV, $D_{out} = 01$. Similarly, as V_{in} crosses 200 mV and 300 mV, D_{out} increments to 10 and 11, respectively.

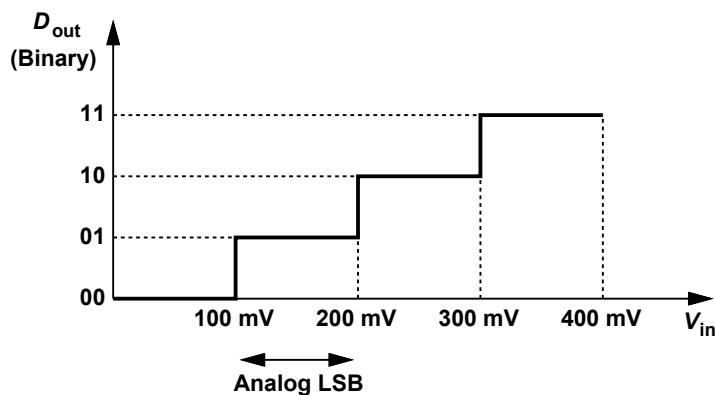


Figure 1.4 Input-output characteristic of a 2-bit quantizer.

The foregoing example indicates that quantization is similar to “truncation” rather than to “rounding.” An input voltage of 160 mV, for example, is mapped to $D_{out} = 01$, as if it were 101 mV. We could also write $D_{out} \equiv [V_{in}/100 \text{ mV}]$, where the square brackets denote the “integer part” of the quantity. (Recall $y = [x]$ from calculus.) We also say the ADC has an analog least-significant bit (LSB) of 100 mV. We deal with sampling in Chapter 2 and with quantization in Chapter 12.

A quantizer with 2^N levels has a “resolution” of N bits. We expect that a greater N makes the quantized signal a more accurate copy of the analog input. That is, the *error* introduced by the quantizer decreases. Called “quantization noise,” this error leads to a finite signal-to-noise ratio (SNR) at the output. We derive this SNR in terms of N for a sinusoidal input in Chapter 12, but the result proves useful even at the beginning of our studies:

$$\text{SNR} = 6.02N + 1.74 \text{ dB.} \quad (1.1)$$

For example, a 10-bit ADC cannot provide an SNR greater than 62 dB.

Depending on the application, N lies between 4 bits and 20 bits, with $6 \leq N \leq 12$ the most common range. For example, the camera in a mobile device employs 10-bit or 12-bit ADCs to digitize images. Lower resolutions would translate to a lower SNR, yielding a “noisy” image.

1.2 Digital-to-Analog Conversion

Digital-to-analog converters (DACs) serve two purposes. First, they interface digital processors with an analog “environment.” For example, the pixels on a laptop display must receive an analog voltage level so as to produce a certain light intensity. Thus, the digital information delivered by the graphics card must be translated to analog form before reaching the display (Fig. 1.5). Since our eyes are sensitive to any imperfections on the display, the DAC must reproduce the information with high “fidelity,” i.e., with minimal noise and distortion.

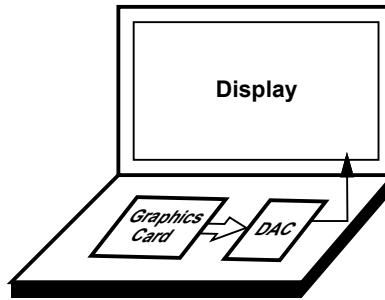


Figure 1.5 Use of a DAC to drive the pixels in a laptop display.

The second application of DACs is within ADCs. As explained in Chapters 14 and 17, some ADC architectures incorporate DACs.

Shown conceptually in Fig. 1.6, a DAC receives a digital input, D_{in} , and generates an analog output, A_{out} . We note from the input-output characteristic example that A_{out} rises in steps equal to Δ as D_{in} goes from 00 to 11, exhibiting four analog levels. The output can be any analog quantity, including voltage, current, or even time.

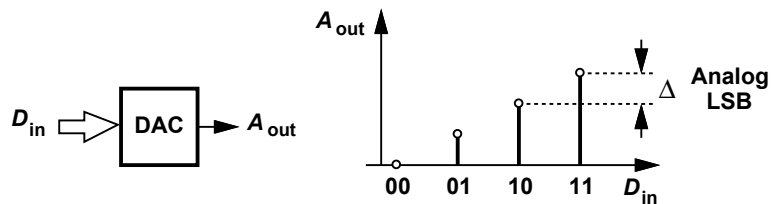


Figure 1.6 Conceptual operation of a DAC.

Example 1.4

A sinusoidal digital input is applied to a 2-bit DAC. Sketch the output as a function of time.

Solution

To construct the output, we first note that both D_{in} and A_{out} can assume only one of four values. The input sinusoid thus appears as shown in Fig. 1.7(a).¹ For each input sample, the DAC produces an integer multiple of the analog step, Δ , and *holds* it until the next value arrives [Fig. 1.7(b)]. The resulting waveform barely resembles a sinusoid.

¹The reader may wonder why the digital input values are not evenly spaced in time. We ignore this point here.

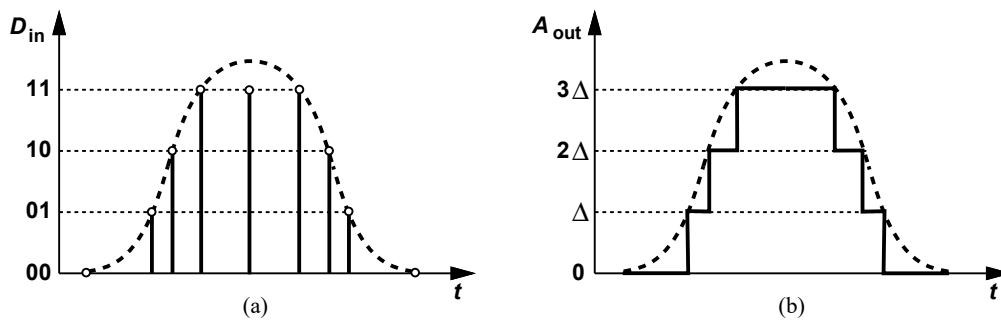


Figure 1.7 (a) A digital sinusoid applied to a DAC, and (b) the DAC output.

Example 1.5

We expect a DAC to perform the inverse of an ADC's operation. Does a DAC convert a discrete-time, discrete-amplitude signal to a continuous-time, continuous-amplitude output?

Solution

The DAC output is continuous in time but not in amplitude. The waveform observed in Fig. 1.7(b) exhibits jumps between discrete amplitude levels, creating unwanted high-frequency components (Chapter 8). For this reason, stand-alone DACs are followed by a low-pass filter so as to produce a smoother waveform (Fig. 1.8).

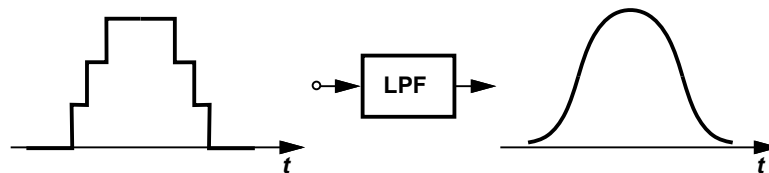


Figure 1.8 Use of a low-pass filter to remove high-frequency components from a DAC output.

Basic Circuit Concepts

Many data converters rely heavily on switched-capacitor (SC) circuits. While the reader is assumed to have a solid understanding of circuit theory, there are a number of basic concepts that should be revisited before we delve into data converter design. This chapter briefly reviews these concepts, beginning with charge conservation and charge sharing. We then study integrated capacitor structures and three general circuit imperfections: offset, gain error, and nonlinearity.

We should make two remarks. If one plate of a capacitor receives charge in the amount of $+Q$, the other *must* receive $-Q$. Also, we say a switch is “on” when it conducts and “off” when it does not; these correspond to the “closed” and “open” states, respectively.

2.1 Charge Conservation

We know from physics that electric charge must be conserved. Applied to the charge arriving at a node, this principle translates to Kirchhoff’s current law (KCL). While the exact definition of charge conservation is beyond our scope, we wish to verbalize it in a particular language suited to switched-capacitor circuits.

Let us begin with the circuit of Fig. 2.1(a), where the capacitor starts with a zero initial condition and charges to V_B after S_1 turns on, drawing a total charge of $Q = C_1 V_B$ from the battery. This positive charge

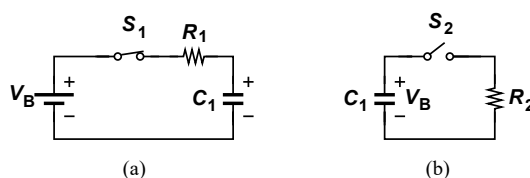


Figure 2.1 (a) A capacitor charging through a resistor, and (b) the capacitor discharging through a resistor.

resides on the top plate of C_1 , attracting a negative charge equal to $-C_1 V_B$ to its bottom plate. We can also say that positive charge flows from the top terminal of the battery to the top plate of C_1 and *from* the bottom plate of C_1 to the bottom terminal of V_B . Thus, it appears that positive current flows clockwise around the circuit.

In the next step, we turn S_1 off and, as shown in Fig. 2.1(b), connect another resistor across C_1 . Now, positive charge flows from the top plate of C_1 , through R_2 , to its bottom plate. As time progresses, the positive and negative charge cancel, yielding zero volts across C_1 .

In some circuits, as in the foregoing examples, the total charge on the capacitor(s) changes after a switching event. We thus say the charge on the capacitor(s) is not conserved. To appreciate the significance of this point, let us consider a simple circuit in which the capacitor charge *is* conserved. As shown in Fig. 2.2(a), a voltage step is applied to the left plate of C_1 while its right plate is not attached to any other device. Assume the initial voltage on C_1 is zero. How does V_{out} behave? Since no current can flow through C_1 , the voltage across C_1

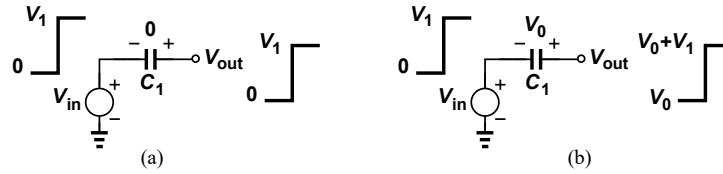


Figure 2.2 (a) A step applied to a floating capacitor, and (b) step response with an initial condition.

remains zero, demanding that $V_{out} = V_{in} = V_1 u(t)$, where $u(t)$ denotes the unit step. The reader may recall from circuit theory that we can view C_1 as a short circuit at $t = 0^+$ and arrive at the same conclusion. The charge on C_1 is zero before and after the step. This is because the right plate of C_1 has no path through which it can receive charge. In other words, even though V_{in} is capable of delivering charge to the left plate of C_1 , it does not.

We now repeat the above experiment but with an initial voltage of V_0 on C_1 [Fig. 2.2(b)]. Since the charge on the right plate is conserved, so is the charge on the left plate, yielding a constant voltage on C_1 equal to V_0 . The output therefore jumps from V_0 to $V_0 + V_1$. We say the capacitor passes *changes* in the input voltage in both examples.

Example 2.1

The voltage source shown in Fig. 2.3 applies a random signal to the left plate of C_1 with a zero initial

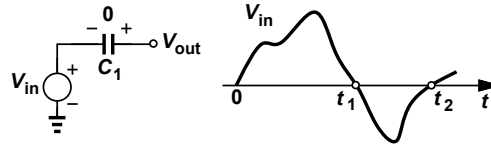


Figure 2.3 A random signal applied to a floating capacitor.

condition. Determine V_{out} at $t = t_1$ and $t = t_2$.

Solution

In this case, $V_{out} = V_{in}$ at all times; thus, $V_{out} = 0$ at $t = t_1$ and $t = t_2$. From another perspective, the net change in V_{in} is zero from $t = 0$ to $t = t_1$, and so is the change in V_{out} . The same point applies to $t = t_2$.

Let us turn to the slightly more complex arrangement depicted in Fig. 2.4(a), assuming zero initial condi-

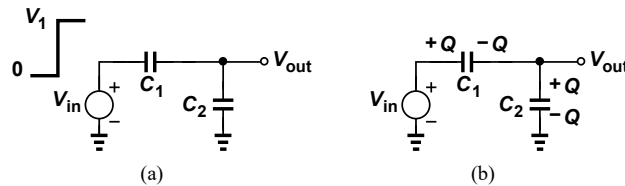


Figure 2.4 (a) A step applied to a capacitive voltage divider, and (b) charge distribution on the capacitors.

tions. We analyze the circuit in three steps. First, we ask, is the charge on the capacitors conserved here? We recognize that the right plate of C_1 can receive charge only from the top plate of C_2 . That is, the sum of the charges on these two plates is conserved and equal to zero.

Second, we surmise that the positive step generated by V_{in} may deliver positive charge to the left plate of C_1 . Denoting this charge by $+Q$, we observe that the right plate of C_1 *must* receive $-Q$, which can only come from the top plate of C_2 [Fig. 2.4(b)]. Upon losing $-Q$, this plate now carries an amount of charge equal to $+Q$, requiring that the bottom plate of C_2 absorb $-Q$ from the ground. We say a charge equal to Q flows clockwise around the loop. We also note that two capacitors in series must carry equal charges if they begin with zero initial conditions.

In the third step, we write the voltages on C_1 and C_2 as Q/C_1 and Q/C_2 , respectively, and enforce Kirchhoff's voltage law (KVL):

$$V_1 = \frac{Q}{C_1} + \frac{Q}{C_2}. \quad (2.1)$$

That is,

$$Q = \frac{C_1 C_2}{C_1 + C_2} V_1, \quad (2.2)$$

as expected for a series combination. It follows that

$$V_{out} = \frac{Q}{C_2} \quad (2.3)$$

$$= \frac{C_1}{C_1 + C_2} V_1. \quad (2.4)$$

The voltage across C_1 is equal to $[C_2/(C_1 + C_2)]V_1$ (why?). The capacitive divider attenuates the input voltage change. If undesirable, this effect is minimized by choosing C_1 much greater than C_2 .

We have seen that the charge on C_1 is not conserved in Fig. 2.1(b) but that on C_1 and C_2 in Fig. 2.4 is. Why is that? The principal difference is that the former contains a *resistive* path through which charge can flow while the latter does not. We articulate the latter's property by drawing a closed surface (also called a "Gaussian surface") around the right plate of C_1 and the top plate of C_2 (Fig. 2.5), recognizing that the total charge within this surface must remain constant. We also say X is a "high-impedance" node.

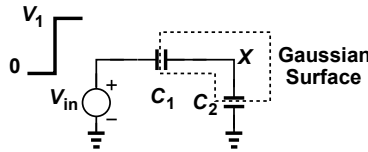


Figure 2.5 Concept of Gaussian surface for two capacitors.

2.2 Charge Sharing

Charge sharing commonly occurs in many data converter circuits—intentionally or not—and merits an in-depth study.

In the circuit of Fig. 2.6(a), the initial voltages on C_1 and C_2 are equal to V_0 and zero, respectively. We draw a Gaussian surface around the capacitors' top plates and note that its total charge must be constant. Thus, after S_1 turns on [Fig. 2.6(b)], the initial charge on C_1 , $C_1 V_0$, is shared between the two capacitors. Denoting the new voltage by V_X , we write

$$C_1 V_0 = C_1 V_X + C_2 V_X \quad (2.5)$$

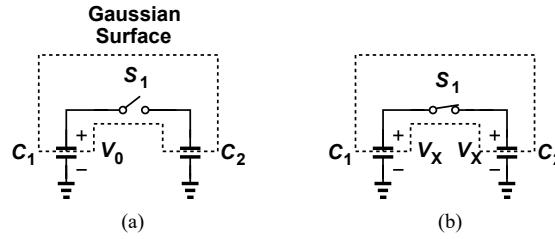


Figure 2.6 (a) Two capacitors within a Gaussian surface, and (b) conservation of charge before and after the switch turns on.

and hence

$$V_X = \frac{C_1}{C_1 + C_2} V_0. \quad (2.6)$$

If undesirable, charge sharing can be minimized by selecting $C_1 \gg C_2$.

Example 2.2

Calculate the final capacitor voltages in Fig. 2.7(a) if the initial conditions are V_0 for C_1 and zero for C_2 and C_3 . The switch turns on at $t = 0$.

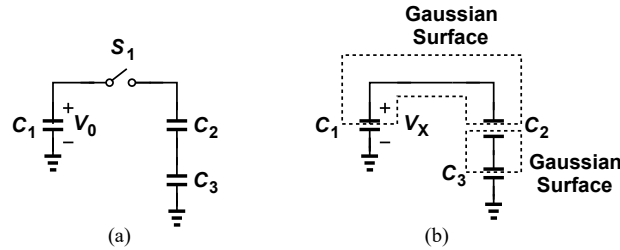


Figure 2.7 (a) Two series capacitors sharing charge with another, and (b) use of Gaussian surfaces for analysis.

Solution

As shown in Fig. 2.7(b), we identify two Gaussian surfaces whose internal charge must be constant. But the circuit is simple enough to allow us to write charge conservation as

$$C_1 V_0 = C_1 V_X + \frac{C_2 C_3}{C_2 + C_3} V_X. \quad (2.7)$$

That is,

$$V_X = \frac{C_1}{C_1 + \frac{C_2 C_3}{C_2 + C_3}} V_0. \quad (2.8)$$

This voltage is attenuated by C_2 and C_3 , yielding $[C_3/(C_2 + C_3)]V_X$ across the former and $[C_2/(C_2 + C_3)]V_X$ across the latter (why?).

Example 2.3

In the circuit of Fig. 2.8(a), the initial voltages on C_1 and C_2 are equal to V_0 and zero, respectively. Explain what happens after the switch turns on, assuming that the op amp has a high gain.

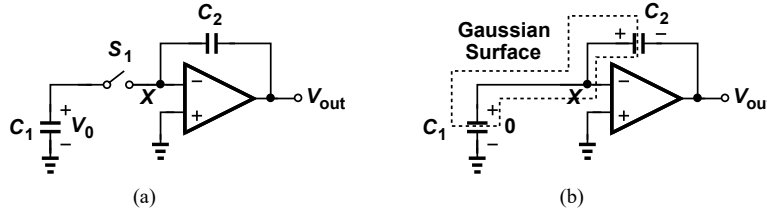


Figure 2.8 (a) A charged capacitor switched to a virtual ground, and (b) use of a Gaussian surface for analysis.

Solution

After S_1 turns on, positive charge flows from the top plate of C_1 to the left plate of C_2 . The high op amp gain ensures that node X is a virtual ground, forcing the voltage across C_1 to fall to zero [Fig. 2.8(b)]. Consequently, all of the initial charge on C_1 is transferred to C_2 , creating a voltage across it equal to $C_1 V_0 / C_2$. In this case, the Gaussian surface encloses the top plate of C_1 and the left plate of C_2 . Since V_{out} is equal to the negative of the voltage across C_2 (why?),

$$V_{out} = -\frac{C_1}{C_2} V_0. \quad (2.9)$$

We say the circuit amplifies the initial voltage by a factor of $-C_1/C_2$.

It is important to recognize the different charge sharing mechanisms in Figs. 2.6(b) and 2.8(b): in the former, C_1 still retains some charge because its final voltage is *not* zero, whereas in the latter, C_1 loses all of its charge owing to the virtual ground. This distinction proves useful in our studies of data converter circuits.

2.3 Integrated Capacitors

Capacitors can be realized in different forms in CMOS technology. The simplest candidate is the MOS capacitor, obtained by tying the source and drain of a device together [Fig. 2.9(a)]. Offering $C_{AB} \approx W L C_{ox}$, this topology typically provides the *densest* structure (i.e., one with the greatest capacitance per unit area). For example, in 28-nm technology, the gate capacitance is around 2 fF/ μm^2 . However, such a value is obtained only if the MOS device is *turned on*. In fact, the capacitance is a nonlinear function of V_{AB} [Fig. 2.9(b)], proving unsuitable for most switched-capacitor circuits.

A much more linear capacitor can be realized by creating a vertical stack of metal plates, as shown in Fig. 2.9(c). We recognize that metal-9 and metal-8 plates form a capacitor, and so do metal-8 and metal-7 plates, etc. The connections shown in the figure place these capacitors in parallel, yielding

$$C_{AB} = C_9 + C_8 + C_7. \quad (2.10)$$

The stack can include lower metal layers as well to provide a greater capacitance density, but it is still less dense than the MOS structure.

The parallel-plate geometry of Fig. 2.9(c) suffers from a parasitic capacitance, C_p , to the substrate, leading to the model shown in Fig. 2.9(d). This parasitic is given by the capacitance between the bottom-most layer— M_6 in our example—and the grounded substrate. Of course, we can decide on which side of C_{AB} the parasitic component should be placed. Typically, C_p is around 5% to 10% of C_{AB} . To appreciate the effect of C_p , let us study the following two examples.

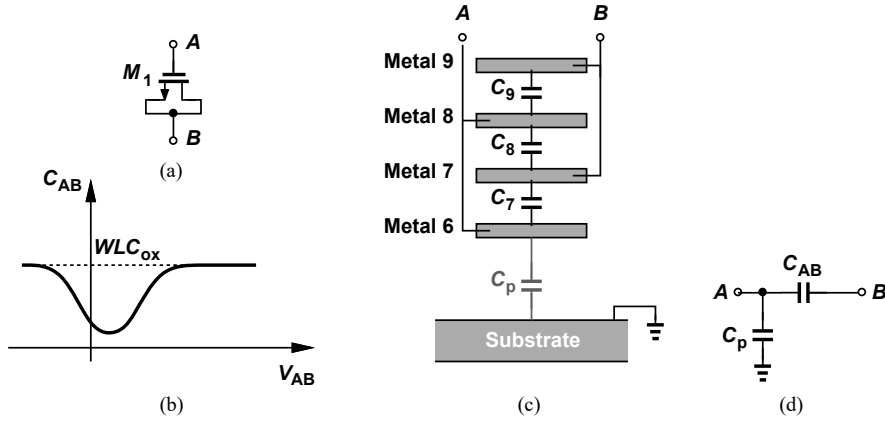


Figure 2.9 (a) Simple NMOS device acting as a capacitor, (b) its C-V characteristic, (c) capacitor made of metal plates, and (d) circuit model showing parasitic component.

Example 2.4

We wish to apply a signal to a common-source (CS) amplifier through a capacitor. Shown in Fig. 2.10(a), such an arrangement employs a current mirror to define the bias of M_1 properly. Resistor R_G ensures that M_b does not interfere with the signal path. How should the value of C_b be chosen? On which side of C_b should its parasitic be placed?

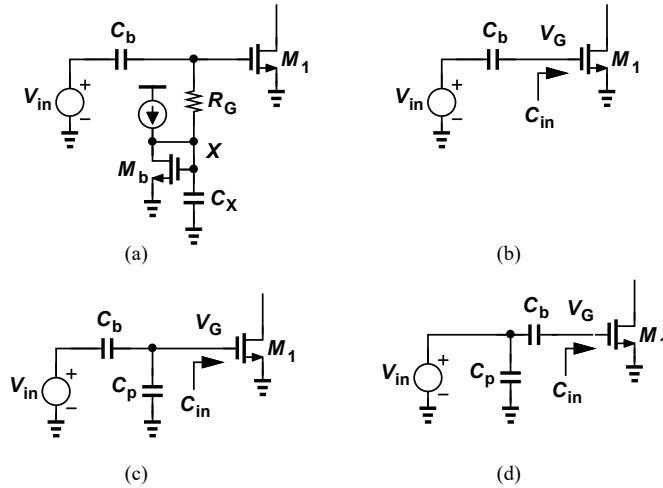


Figure 2.10 (a) Common-source stage with input capacitive coupling, (b) voltage division at the input, (c) effect of C_p at the gate, and (d) effect of C_p on the input source side.

Solution

The value of C_b is governed by two considerations. First, the high-pass filter formed by C_b and R_G must not attenuate the input signal.¹ If the lowest frequency component of V_{in} is denoted by f_L , we have

$$\frac{1}{2\pi R_G C_b} \ll f_L. \quad (2.11)$$

¹We assume C_X is large enough to keep node X at ac ground.

This constraint implies that C_b need not be very large if R_G has a sufficiently high value, providing flexibility in the choice of these components.

The second consideration is less flexible and relates to the attenuation caused by the input capacitance of M_1 [Fig. 2.10(b)], which occurs even if (2.11) is met. We write

$$V_G = \frac{C_b}{C_b + C_{in}} V_{in}, \quad (2.12)$$

noting that C_b should be selected about 5 to 10 times the value of C_{in} .

If the parasitic capacitance of C_b is placed on the right side [Fig. 2.10(c)], then it adds to C_{in} , raising the attenuation factor. We thus prefer to place C_p on the left side [Fig. 2.10(d)].

Example 2.5

The ac-coupled CS amplifier of the previous example is driven by another CS amplifier (Fig. 2.11). Explain how C_p affects the performance.

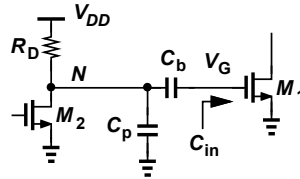


Figure 2.11 Capacitive coupling between two stages.

Solution

The total capacitance seen at node N is equal to $C_p + C_b C_{in} / (C_b + C_{in})$. Since C_p is proportional to C_b , increasing the coupling capacitor slows down the response at node N . For this reason, the coupling capacitors in our circuits cannot be arbitrarily large even if chip area consumption is not a concern.

To increase the capacitance density of the structure shown in Fig. 2.9(c), we can add more metal layers to it. Depicted in Fig. 2.12(a) is an example incorporating nine layers. However, the capacitor also suffers from a large parasitic owing to the close proximity of metal 1 to the substrate. In general, the minimum C_p , as a percentage of C_{AB} , occurs for four to five metal layers [Fig. 2.12(b)]. Here, the horizontal axis represents the number of *topmost* metal layers, starting with metal 9 and metal 8 in our example.

A denser capacitor geometry exploits the *lateral* electric field between metal lines. As shown in Fig. 2.13, two closely-spaced traces exhibit substantial capacitance between them. We can therefore arrange in parallel a large number of *narrow* traces to obtain a large capacitance per unit area [Fig. 2.13(b)]. Interestingly, this capacitor displays parasitics associated with *both* of its terminals: $C_{p1} + C_{p3} + C_{p5}$ from A to ground and $C_{p2} + C_{p4} + C_{p6}$ from B to ground. This “fringe” capacitor can be extended to two or more layers as shown in Fig. 2.13(c), providing a greater density and smaller normalized parasitics. The vertical stacking benefits from the capacitance between the top traces and the bottom traces as well.

The value of these capacitor geometries is difficult to calculate by hand. We therefore resort to layout parasitic extraction tools to obtain the capacitances.

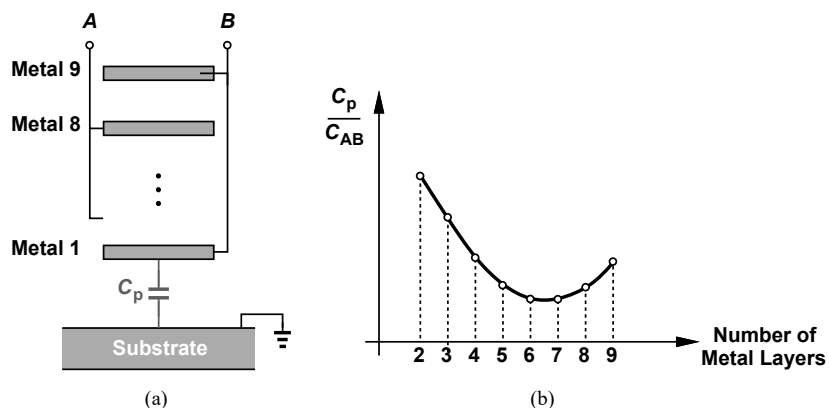


Figure 2.12 (a) Capacitor made of nine metal layers, and (b) normalized parasitic vs. number of layers.

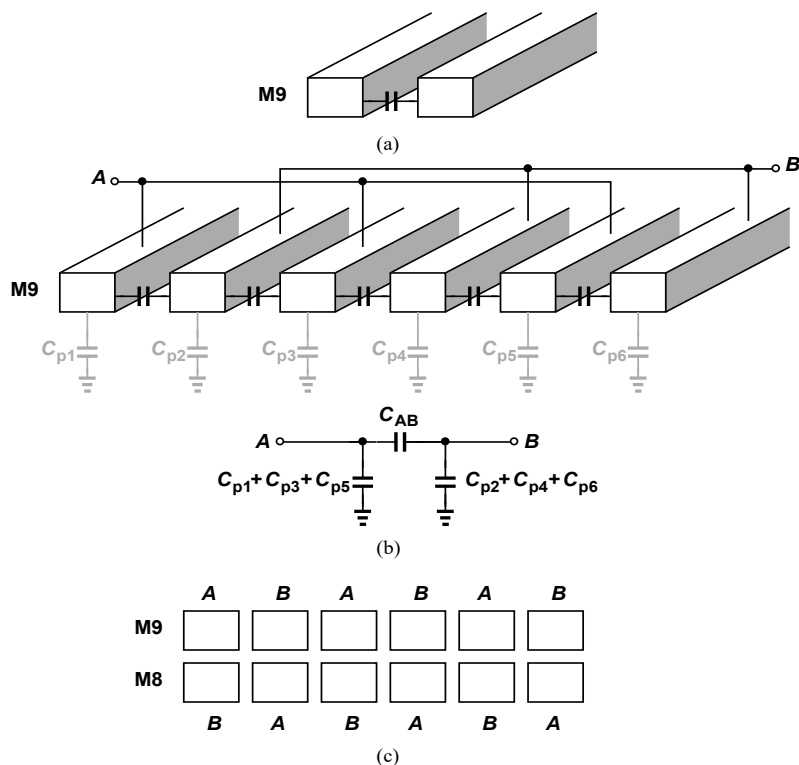


Figure 2.13 (a) Fringe capacitance between two parallel lines, (b) extension of the idea to six lines, and (c) side view of the structure with a metal-8 group.

2.4 Offset, Gain Error, and Nonlinearity

In the study of data converter circuits, we often face three types of imperfections: “offset,” “gain error,” and “nonlinearity.” It is important not to confuse these three effects.

Offset The term “offset” refers to a vertical shift of the input-output characteristic of a circuit or system. Illustrated in Fig. 2.14(a), an offset may be created by asymmetries. For example, a differential pair exhibits an offset at its output if its load resistors are not matched [Fig. 2.14(b)]. This means that for a zero differential

input, the differential output is not zero and is equal to $\Delta R_D I_{SS}/2$ [Fig. 2.14(c)].

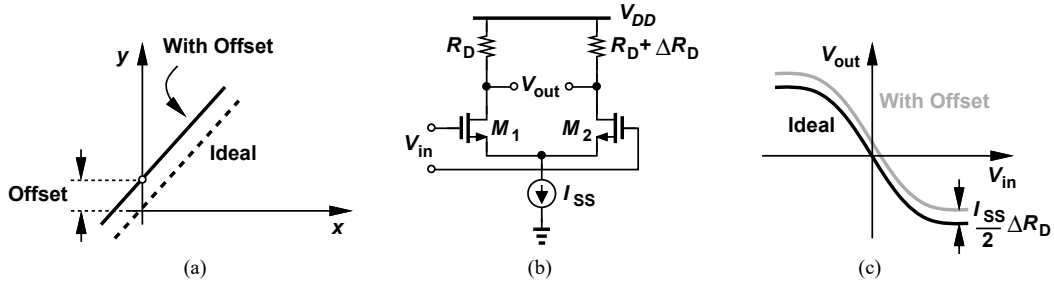


Figure 2.14 (a) Offset in a system, (b) differential pair with mismatched loads, and (c) effect of offset on the pair's characteristic.

Gain Error The term “gain error” represents a departure of the *slope* of the input-output characteristic from its nominal value (Fig. 2.15). For example, the slope can be 2.1 instead of 2.0, implying a gain error of $0.1/2.0 = 5\%$. The differential pair of Fig. 2.14(b) provides a gain of $-g_{m1,2} R_D$ if $\Delta R_D = 0$, and incurs a gain error if the load resistors are not equal to their nominal value. For example, if both resistors increase by

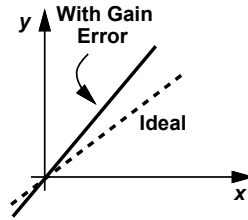


Figure 2.15 Gain error in a system.

10% in one corner of the process, the gain error is +10%.

Nonlinearity The term “nonlinearity” refers to the *variation* of the slope as the input varies. As shown in Fig. 2.16(a), the slopes at x_1 and x_2 are unequal. Thus, the characteristic cannot be expressed by a straight line. We can also view the slope at a given point as the small-signal gain in the vicinity of that point. For example, a differential pair exhibits the large-signal characteristic depicted in Fig. 2.16(b), suffering from a lower gain (smaller slope) as $|V_{in}|$ increases.

It is important to recognize that the nonlinear characteristic of a differential circuit is an *odd* function. Arising from symmetry and exemplified by the differential pair behavior of Fig. 2.16(b), this property means that the output values for $-V_1$ and $+V_1$ are equal and opposite.

The effect of nonlinearity can be quantified by measuring the departure from the ideal case. In Fig. 2.16(a), for example, the characteristic incurs an error of Δy_2 at x_2 due to nonlinearity. We can also approximate the characteristic by a polynomial:

$$y = \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + \dots \quad (2.13)$$

We recognize that $y \approx \alpha_1 x$ if x is so small as to make the nonlinear terms negligible. Thus, α_1 is the slope (gain) in the vicinity of $x = 0$. For a differential circuit, on the other hand, the characteristic is an odd function, and the odd symmetry requires that the even-order terms be absent:

$$y_{\text{diff}} = \alpha_1 x + \alpha_3 x^3 + \dots \quad (2.14)$$

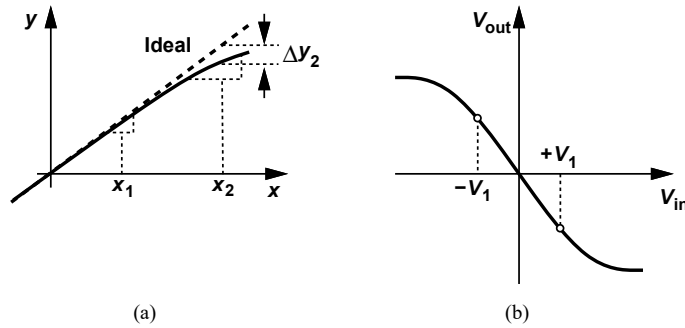


Figure 2.16 (a) Nonlinearity in a system, and (b) nonlinearity in a differential pair.

This guarantees that changing x to $-x$ causes y to change to $-y$.

Another approach to quantifying nonlinearity is to apply a sinusoidal input and examine the output. If $x = A \cos \omega_1 t$, Eq. (2.13) yields

$$y(t) = \alpha_1 A \cos \omega_1 t + \alpha_2 A^2 \cos^2 \omega_1 t + \alpha_3 A^3 \cos^3 \omega_1 t + \dots \quad (2.15)$$

Noting that $\cos^2 \omega_1 t = (1 + \cos 2\omega_1 t)/2$ and $\cos^3 \omega_1 t = (3 \cos \omega_1 t + \cos 3\omega_1 t)/4$, we have

$$y(t) = \left(\alpha_1 A + \frac{3}{4} \alpha_3 A^3 \right) \cos \omega_1 t + \frac{\alpha_2 A^2}{2} + \frac{\alpha_2 A^2}{2} \cos 2\omega_1 t + \frac{\alpha_3 A^3}{4} \cos 3\omega_1 t + \dots \quad (2.16)$$

The terms containing $2\omega_1$ and $3\omega_1$ are called the second and third “harmonics,” respectively. We say the input has incurred “harmonic distortion.” Note the stronger dependence of the higher harmonics upon A .

Example 2.6

Study the nonlinearity of single-ended and differential circuits.

Solution

Consider the nonlinear single-ended system in Fig. 2.17(a), assuming $x(t) = A \sin \omega t$. Mapping the values

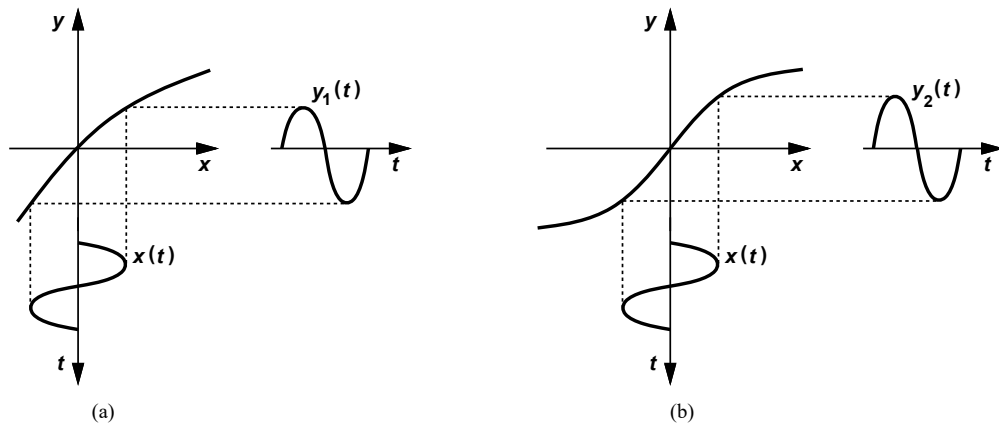


Figure 2.17 Sine wave traveling through a (a) single-ended or (b) differential nonlinear system.

of $x(t)$ to $y(t)$ through the characteristic, we observe that, in this example, the negative half cycles of $y_1(t)$ are taller than the positive ones, i.e., the signal is distorted. Now, we turn to the differential characteristic in

Fig. 2.17(b) and apply $x(t)$. The symmetry of the circuit implies that the behavior of $y_2(t)$ when $x(t) > 0$ is exactly the same as the behavior of $y_2(t)$ when $x(t) < 0$. Thus, $y_2(t)$ is an odd function of time.

Example 2.7

Can a characteristic containing x^2 be free from the second harmonic?

Solution

Yes, it can. Consider the situation depicted in Fig. 2.18(a), where $x(t) = A \sin \omega_1 t$ is applied to $y = \alpha_3 x^3$. The output is free from the second harmonic. Now, we shift both the characteristic and the input dc level by

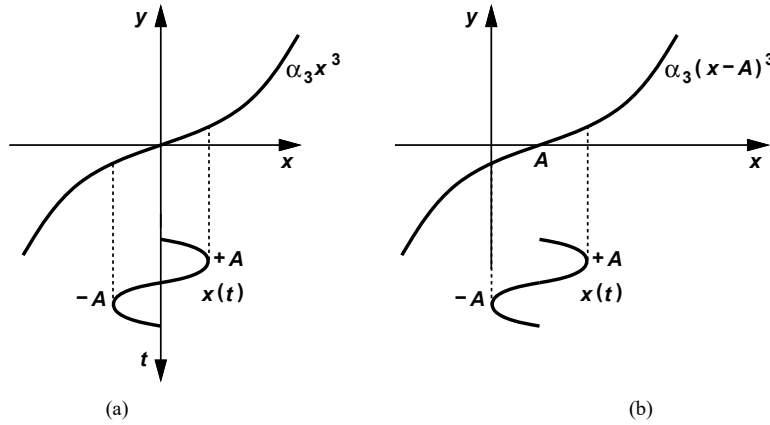


Figure 2.18 Sine wave experiencing (a) symmetric third-order nonlinearity, or (b) nonlinearity with offset.

A [Fig. 2.18(b)], expecting intuitively that the output waveform does *not* change because

$$y = \alpha_3 (x - A)^3 \quad (2.17)$$

$$= \alpha_3 (A \sin \omega_1 t + A - A)^3 \quad (2.18)$$

$$= \alpha_3 (A \sin \omega_1 t)^3. \quad (2.19)$$

That is, even though $y = \alpha_3 (x - A)^3 = \alpha_3 x^3 - 3\alpha_3 x^2 A + 3\alpha_3 x A^2 - A^3$ contains an x^2 term, the output does not have a second harmonic. This is generally true if the input is symmetric around the characteristic's axis of symmetry.

Some nonlinear systems cannot be characterized by a simple polynomial. In such cases, we can evaluate the effect of nonlinearity by applying a sinusoid and computing the harmonic distortion.

Harmonic distortion is specified by a number of quantities. The second harmonic distortion, HD_2 , is defined as the ratio of the amplitudes of the second harmonic and the fundamental. For $y = \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$, and $\alpha = A \cos \omega_1 t$, we have

$$\text{HD}_2 = \frac{\alpha_2 A^2 / 2}{\alpha_1 A + 3\alpha_3 A^3 / 4} \quad (2.20)$$

$$\approx \frac{\alpha_2 A}{2\alpha_1}, \quad (2.21)$$

if the second term in the denominator is neglected. For the third order,

$$\text{HD}_3 = \frac{\alpha_3 A^3/4}{\alpha_1 A + 3\alpha_3 A^3/4} \quad (2.22)$$

$$\approx \frac{\alpha_3 A^2}{4\alpha_1}. \quad (2.23)$$

We also define the total harmonic distortion, THD, as

$$\text{THD} = \frac{P_2 + P_3 + \cdots}{P_1}, \quad (2.24)$$

where P_j denotes the power of the j th harmonic.

Problems

- 2.1. Determine $V_{out}(t)$ in Fig. 2.3 if C_1 holds an initial condition equal to V_0 . We can roughly say that C_1 acts as a battery in this case.
- 2.2. Suppose capacitor C_1 in Fig. 2.4 holds an initial condition equal to V_0 . Determine V_{out} after the input step is applied.
- 2.3. Suppose we tie a resistor R_1 from node X in Fig. 2.5 to ground.
 - (a) Plot $V_X(t)$.
 - (b) Do we still have a Gaussian surface surrounding node X ?
- 2.4. If C_1 and C_2 in Fig. 2.6(a) begin with initial conditions equal to V_1 and V_2 , respectively, determine their voltage after switch is turned on.
- 2.5. Suppose C_1 and C_2 in Fig. 2.8(a) begin with initial conditions equal to V_1 and V_2 , respectively. Determine V_{out} after S_1 is turned on.
- 2.6. We wish to apply a small voltage difference between the gates of M_1 and M_2 in Fig. 2.14(b) so as to null the output offset. How much should this voltage difference be? This is called the “input-referred” offset.
- 2.7. A difference pair is designed for a nominal small-signal voltage gain of 2. Neglect channel-length modulation. How much gain error do we observe if the tail current is raised by 10%?
- 2.8. Repeat the previous problem if the transistors’ gate oxide thickness is 10% greater than the nominal value.
- 2.9. The input-output characteristic of a circuit is expressed as $y = \alpha_0 + \alpha_1 x$. Does this circuit satisfy superposition? Strictly speaking, the offset, α_0 , makes the circuit nonlinear, but we mostly disregard this point.
- 2.10. A nonlinear circuit is characterized by $y = \alpha_1 x + \alpha_2 x^2$. If we apply $x(t) = A_1 \cos \omega_1 t + A_2 \cos \omega_2 t$, determine $y(t)$ and its frequency components.
- 2.11. A circuit characterized by $y = \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3$ receives $x(t) = A \cos \omega_1 t$. Determine the maximum value of A if the second harmonic must be 60 dB (a factor of 1000) below the fundamental (the component at ω_1).
- 2.12. Repeat the previous problem if the third harmonic must be 60 dB below the fundamental.
- 2.13. A differential circuit characterized by $y = \alpha_1 x + \alpha_3 x^3$ senses $x(t) = A \cos \omega_1 t$. Determine the greatest tolerable value of α_3 if the third harmonic must remain 60 dB below the fundamental.
- 2.14. Consider the common-source stage depicted in Fig. 2.19, where M_1 operates in the saturation region and behaves as a square-law device. Neglecting channel-length modulation, determine the second harmonic amplitude at the output if $x(t) = V_1 \cos \omega_2 t + V_0$.

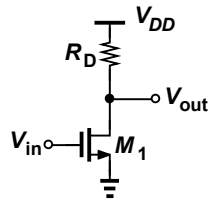


Figure 2.19 CS stage.

- 2.15. We wish the second harmonic in Fig. 2.19 to remain 60 dB below the fundamental. How large can V_1 be? Does V_0 affect this result?
- 2.16. In the common-source stage of Fig. 2.20, channel-length modulation and body effect are negligible. If M_1 operates in the saturation region and $V_{in} = V_1 \cos \omega_1 t + V_0$, determine $V_{out}(t)$. Does the circuit exhibit harmonic distortion?

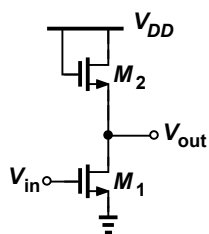


Figure 2.20 Another CS stage.

- 2.17.** From the result obtained in the previous problem, compute the voltage gain of the circuit. Does this agree with small-signal calculations?
- 2.18.** If we replace M_2 in Fig. 2.20 with a diode-connected PMOS device, repeat Problem 2.16.
- 2.19.** Repeat Problem 2.16 for the circuit shown in Fig. 2.21.

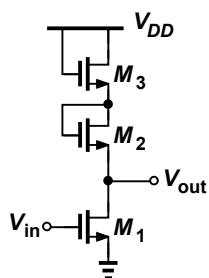


Figure 2.21 CS stage with diode loads.

- 2.20.** From the previous problem, calculate the voltage gain of the circuit and compare it to the result of small-signal analysis.

Sampling Circuits

As mentioned in Chapter 1, analog-to-digital conversion involves two operations, namely, sampling and quantization. In this chapter, we deal with the former. We begin with the general concept of sampling and its mathematical formulation. We then study the behavior of MOS transistors as sampling switches, and analyze their imperfections. We conclude with a detailed treatment of bootstrapped sampling circuits.

3.1 Basic Sampling Concepts

3.1.1 Ideal Sampler

Samplers generally convert a continuous-time waveform to a discrete-time signal. An “ideal” sampler simply multiplies its input by a train of impulses, as shown in Fig. 3.1(a). Here, T_S denotes the sampling period, and

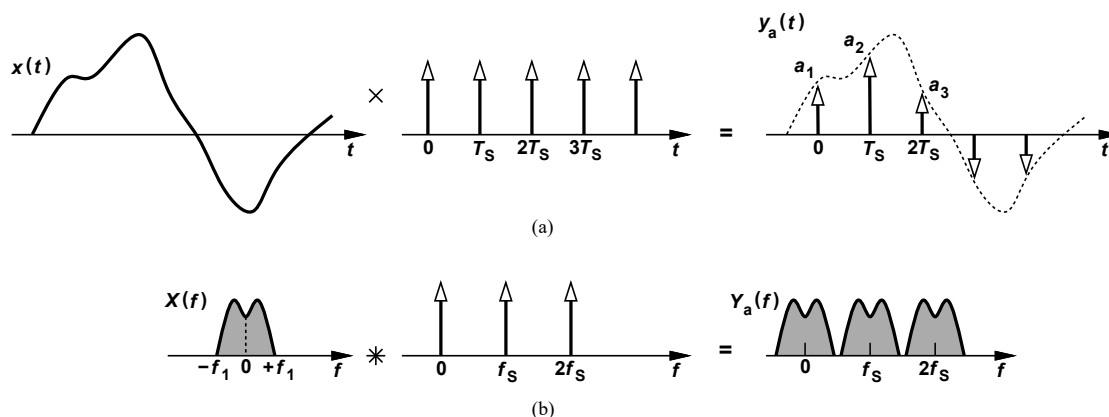


Figure 3.1 (a) Ideal sampling, and (b) its resulting spectrum.

$f_S = 1/T_S$ is called the “sampling rate.” The output can be expressed as

$$y_a(t) = x(t) \sum_{k=-\infty}^{+\infty} \delta(t - kT_S). \quad (3.1)$$

In the frequency domain, the Fourier transform of $x(t)$ is convolved with that of the impulse train, which itself is a train of impulses. Illustrated in Fig. 3.1(b), this operation creates copies of the input spectrum

centered at integer multiples of the sampling rate. We have

$$Y_a(f) = X(f) \star \frac{1}{T_S} \sum_{n=-\infty}^{+\infty} \delta(f - nf_S) \quad (3.2)$$

$$= \frac{1}{T_S} \sum_{n=-\infty}^{+\infty} X(f - nf_S). \quad (3.3)$$

We observe that the shifted copies of $X(f)$ experience overlap (“aliasing”) if the input signal bandwidth, f_1 , exceeds half of the sampling rate (Nyquist’s theorem) (Fig. 3.2). The frequency range from zero to $f_S/2$ is called the “first Nyquist zone.”

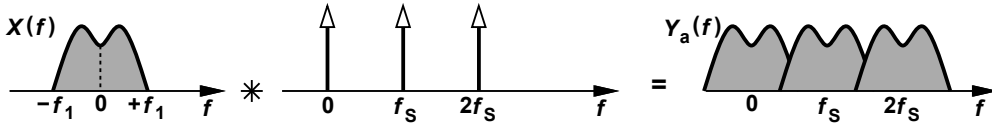


Figure 3.2 Aliasing due to excessive signal bandwidth.

3.1.2 Zero-Order-Hold Sampler

The ideal sampling depicted in Fig. 3.1(a) proves impractical as the output does not last long enough for the subsequent stage to process it. For example, a quantizer sensing the sampler’s output requires a certain amount of time to digitize the analog value. For this reason, we prefer to implement a “sample-and-hold” function, whereby the input is sampled and the result is held (frozen) for some time. As shown in Fig. 3.3(a), at each sampling point, the output immediately assumes the input value and then remains constant until the next sampling instant. This function is also called a zero-order hold (ZOH).

The Fourier transform of $y_b(t)$ can be obtained by noting that this signal is equivalent to that in Fig. 3.1(a), but convolved with a pulse [Fig. 3.3(b)]:

$$y_b(t) = [x(t) \sum_{k=-\infty}^{+\infty} \delta(t - kT_S)] \star p(t). \quad (3.4)$$

Thus, the Fourier transform of $y_a(t)$ is multiplied by that of $p(t)$:

$$Y_b(f) = \left[\frac{1}{T_S} \sum_{n=-\infty}^{+\infty} X(f - nf_S) \right] \cdot P(f), \quad (3.5)$$

where $P(f)$ denotes the Fourier transform of $p(t)$. We know that $P(f)$ has the shape of a sinc function, but since $p(t)$ is a pulse centered at $T_S/2$, $P(f)$ emerges as

$$P(f) = e^{-j2\pi f T_S/2} \frac{\sin \pi f T_S}{\pi f}. \quad (3.6)$$

It follows that

$$Y_b(f) = e^{-j\pi f T_S} \frac{\sin \pi f T_S}{\pi f T_S} \sum_{n=-\infty}^{+\infty} X\left(f - \frac{n}{T_S}\right). \quad (3.7)$$

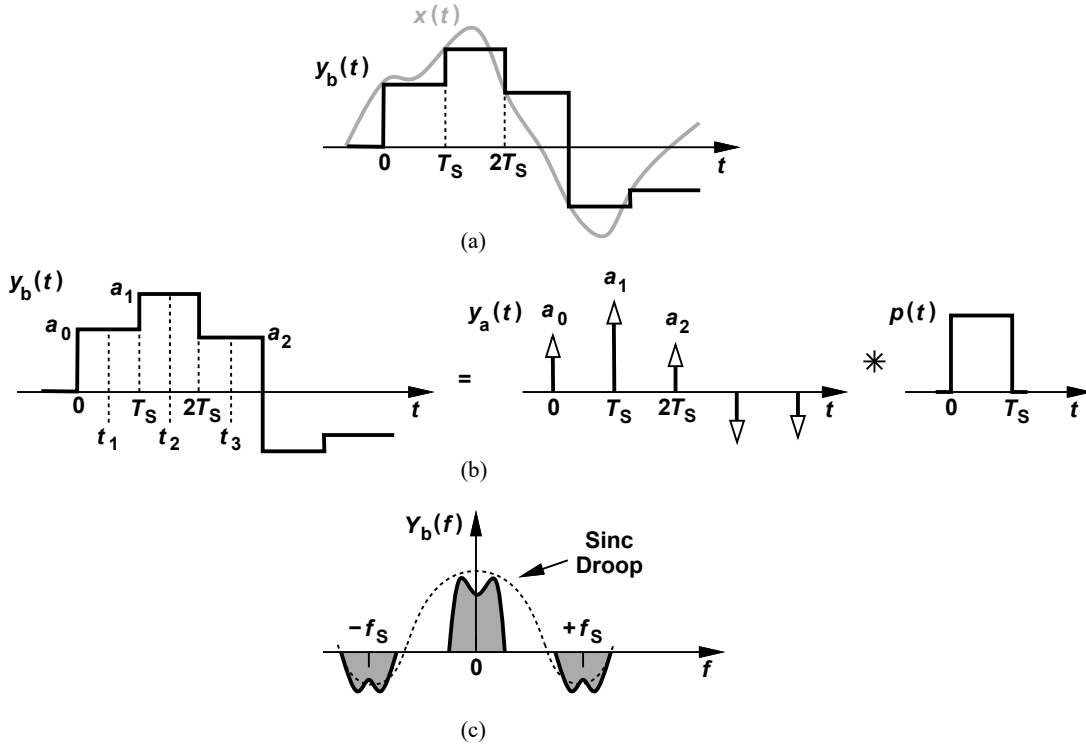


Figure 3.3 (a) Zero-order hold, (b) output viewed as convolution of two waveforms, and (c) output spectrum.

Figure 3.3(c) plots the result, revealing that even the part centered around $f = 0$ experiences a “sinc droop” as it is multiplied by $\sin(\pi f T_s)/(\pi f T_s)$. In essence, the high-frequency components in the input signal are attenuated due to the sinc envelope.

Fortunately, the sinc envelope vanishes if the zero-order-hold sampler is followed by a quantizer. To understand this point, consider the $y_a(t)$ and $y_b(t)$ waveforms in Figs. 3.1 and 3.3(b). We assume that (a very fast) quantizer digitizes the samples provided by $y_a(t)$ at time instants $t = 0, T_s, \dots$, i.e., a_0, a_1 , etc. Next, we assume that a more practical quantizer digitizes the levels of $y_b(t)$ at $t = t_1, t_2, t_3, \dots$, where $t_2 - t_1 = T_s$, $t_3 - t_2 = T_s$, etc. We observe that the values sensed and digitized by the two quantizers are equal, concluding that the second quantizer output is also free from the sinc envelope. In essence, the quantizer treats $y_b(t)$ as a discrete-time signal, which is equivalent to impulse sampling. In summary, the cascade of a ZOH circuit and a quantizer does not suffer from the sinc droop effect. This is the case in analog-to-digital converters.

Example 3.1

How can we implement a circuit that approximates a ZOH?

Solution

Consider the arrangement shown in Fig. 3.4, where switch S_1 connects C_1 to the input when CK is high. Thus, V_{out} “tracks” the input for ΔT seconds and is frozen (held) for $T_s - \Delta T$ seconds. If ΔT is sufficiently short, V_{out} resembles a ZOH waveform. We ask, however, does the capacitor voltage find enough time to reach V_{in} in such a short time? We expect that the on-resistance of S_1 would pose limitations, as explained below.

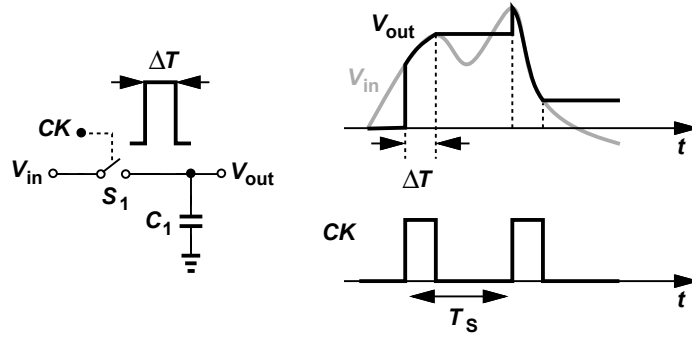


Figure 3.4 (a) Sampling circuit, and (b) its waveforms.

3.1.3 Track-and-Hold Circuit

The finite time constant associated with S_1 and C_1 (and possibly the output impedance of the previous stage) in Fig. 3.4 does not allow V_{out} to track V_{in} instantaneously. As shown in Fig. 3.5(a), when S_1 turns on at t_1 , V_{out} begins to approach V_{in} but requires a certain amount of time. We must therefore keep S_1 on for a sufficiently long time interval to ensure that the input is “acquired” with acceptable precision.

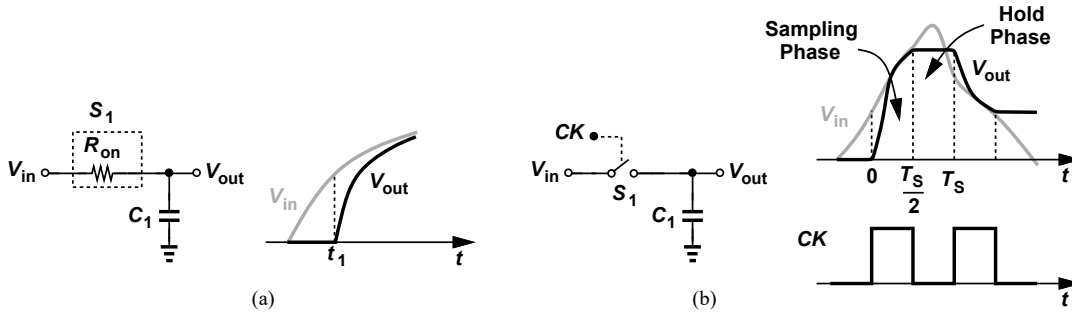


Figure 3.5 (a) Sampling circuit in acquisition mode, and (b) track-and-hold circuit.

From these observations emerges the third type of sampling, namely, the “track-and-hold” circuit (also called a “sample-and-hold” circuit). Depicted in Fig. 3.5(b), such a structure is driven by a clock waveform that turns S_1 on for, say, 50% of T_S . We select the circuit’s time constant short enough that $|V_{in} - V_{out}|$ becomes adequately small at $t = kT_S + T_S/2$.

Example 3.2

Can the sampling time in Fig. 3.5(b) exceed $T_S/2$?

Solution

Yes, it can. Recall from Chapter 1, however, that the sampler is followed by a quantizer, which processes the *held* signal. Thus, a greater sampling period translates to less time for quantization. For this reason, we often favor the quantizer by allocating *less* than $T_S/2$ to the sampler.

Among the three sampling schemes studied above, only the track-and-hold topology proves practical. We hereafter assume this structure for all of our sampling functions and refer to it as the “sampler” or the “sampling circuit.” From the discussion in the previous section, we surmise that V_{out} in Fig. 3.5(b) suffers from a sinc droop but, in conjunction with a quantizer, still behaves as in an ideal sampler. We also call the

time intervals between kT_S and $kT_S + T_S/2$ the “sampling phase” or the “acquisition phase,” and those between $kT_S + T_S/2$ and $(k + 1)T_S$ the “hold phase.”

We hereafter change our notation and represent the clock frequency by f_{CK} and its period by T_{CK} .

3.1.4 Choice of Sampler Time Constant

Let us assume the simple RC model shown in Fig. 3.6(a) for a sampler when the switch is turned on. Suppose

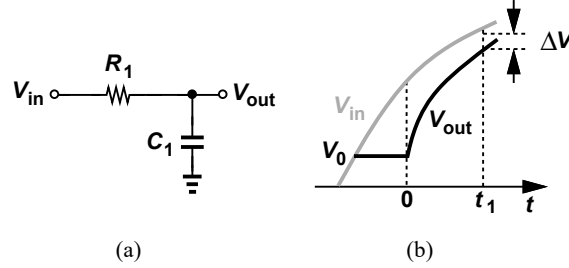


Figure 3.6 (a) Sampling circuit model, and (b) its waveforms.

the circuit runs with a clock frequency of f_{CK} and senses an analog input at a frequency, f_{in} , slightly less than $f_{CK}/2$ so as to satisfy Nyquist’s theorem. How should we select the time constant R_1C_1 ? We can approach the question in the frequency domain and say the circuit’s bandwidth, $1/(2\pi R_1C_1)$, must be wide enough to accommodate the input signal at $f_{in} \approx f_{CK}/2$:

$$\frac{1}{2\pi R_1C_1} \geq \frac{f_{CK}}{2}. \quad (3.8)$$

That is,

$$R_1C_1 \leq \frac{1}{\pi f_{CK}} \quad (3.9)$$

$$\leq \frac{T_{CK}}{\pi}, \quad (3.10)$$

where $T_{CK} = 1/f_{CK}$ denotes the clock period. But should such a limit not depend on the desired sampling *accuracy*? In practice, R_1C_1 must be quite smaller. To understand why, consider the scenario depicted in Fig. 3.6(b), where the switch turns on at $t = 0$ and V_{out} begins from a previous value, V_0 . The circuit must sample the input with adequate accuracy in the given time, t_1 , before the switch turns off. That is, ΔV must be sufficiently small to satisfy the precision of the overall ADC. As a rule of thumb, we target $\Delta V < 0.5$ LSB, and select R_1C_1 short enough to guarantee this condition for the maximum signal change from $t = 0$ to $t = t_1$.

Example 3.3

In the simplified situation shown in Fig. 3.7, V_{in} is a step from zero to the full-scale value, V_1 . Determine how long it takes V_{out} to reach within 0.5 LSB of V_{in} in a 10-bit system, where the LSB is equal to $V_1/2^{10}$.

Solution

For a 10-bit system, $0.5 \text{ LSB} = (V_1/1024)/2$. For the difference between the input and the output to drop to this value, we have

$$V_{out}(t_1) = V_1 \left(1 - \exp \frac{-t_1}{R_1C_1} \right) = V_1 - \frac{V_1}{2048}. \quad (3.11)$$

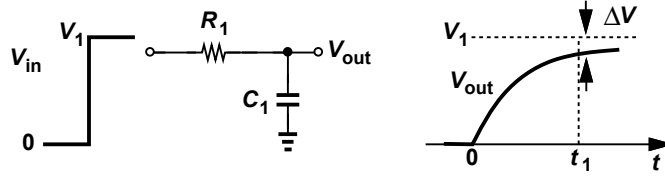


Figure 3.7 Illustration of sampling error.

It follows that

$$t_1 = R_1 C_1 \ln(2048) \quad (3.12)$$

$$= 7.62 R_1 C_1. \quad (3.13)$$

We say a 10-bit sampler requires 7.6 constants to “acquire” the input. If $t_1 = T_{CK}/2$, then we must have

$$R_1 C_1 \leq \frac{T_{CK}}{15.24}, \quad (3.14)$$

a more stringent bound than that in (3.10). For an N -bit ADC, we write $t_1 = (N + 1)R_1 C_1 \ln 2$ and hence

$$R_1 C_1 \leq \frac{T_{CK}}{2(N + 1) \ln 2}. \quad (3.15)$$

In summary, for typical ADC resolutions, the choice of the sampler time constant is dictated by the time necessary to acquire the input. If this condition is met, the sampler bandwidth is also sufficient to avoid attenuating the input.

3.2 Basic Sampling Circuits

We wish to implement sampling circuits using MOS devices. Recall from basic analog design that an NMOS transistor operating in the deep-triode region behaves as a resistor with a value of

$$R_{on} = \frac{1}{\mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})}, \quad (3.16)$$

where μ_n denotes the electron mobility and C_{ox} the gate oxide capacitance per unit area. If in the saturation region, the transistor carries a current equal to

$$I_D = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})^2, \quad (3.17)$$

where we have assumed a simple square-law device and neglected channel-length modulation.

The arrangement shown in Fig. 3.5(b) can serve as a starting point for implementing samplers. Let us realize S_1 by an NMOS transistor [Fig. 3.8(a)] and analyze the circuit’s acquisition behavior. We consider two extreme cases for V_{in} . First, suppose $V_{in} = V_{DD}$ and the initial condition on C_1 is zero [Fig. 3.8(b)]. Then, M_1 begins in the saturation region (why?), drawing a current from V_{DD} and charging C_1 . The reader may recognize M_1 acts as a diode-connected device or a source follower here. The large overdrive voltage, $V_{GS} - V_{TH} = V_{DD} - V_{TH}$, at $t = 0$ results in a high drain current. Assuming a square-law transistor,

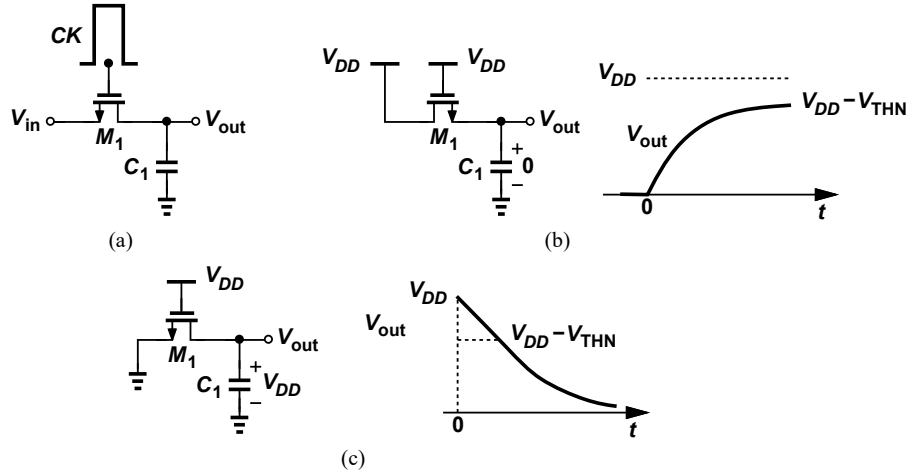


Figure 3.8 (a) Sampling using an NMOS device, (b) acquisition of an input equal to V_{DD} , and (c) acquisition of an input equal to zero.

neglecting channel-length modulation and body effect, and noting that $V_{GS} = V_{DD} - V_{out}$, we have

$$I_D = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{DD} - V_{out} - V_{TH})^2 = C_1 \frac{dV_{out}}{dt}. \quad (3.18)$$

Thus,

$$\frac{1}{2} \mu_n \frac{C_{ox}}{C_1} \frac{W}{L} dt = \frac{dV_{out}}{(V_{DD} - V_{out} - V_{TH})^2}. \quad (3.19)$$

Integrating the left side from 0 to t and the right side from 0 to V_{out} , we have

$$V_{out} = V_{DD} - V_{TH} - \frac{1}{\frac{1}{2} \mu_n \frac{C_{ox}}{C_1} \frac{W}{L} t + \frac{1}{V_{DD} - V_{TH}}}. \quad (3.20)$$

We observe that $V_{out} \rightarrow V_{DD} - V_{TH}$ as $t \rightarrow \infty$. This is because, for $V_{out} = V_{DD} - V_{TH}$, the transistor turns off (unless we include subthreshold conduction). The key point here is that V_{out} does *not* approach $V_{in} = V_{DD}$ in this case, i.e., the circuit does not acquire the input properly.

In the second extreme case, we assume that $V_{in} = 0$ and the initial condition on C_1 is equal to V_{DD} [Fig. 3.8(c)]. Since $V_{out} > V_{in}$, the source and drain terminals are swapped, the transistor begins in saturation with an overdrive equal to $V_{DD} - V_{TH}$, and C_1 is discharged at a constant rate:

$$\frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{DD} - V_{TH})^2 = C_1 \frac{dV_{out}}{dt}. \quad (3.21)$$

As V_{out} falls below $V_{DD} - V_{TH}$, M_1 enters the triode region, yielding

$$\frac{1}{2} \mu_n C_{ox} \frac{W}{L} [2(V_{GS} - V_{TH})V_{out} - V_{out}^2] = C_1 \frac{dV_{out}}{dt}. \quad (3.22)$$

We solve this equation in Problem 3.2, but note for now that V_{out} approaches 0 because M_1 eventually acts as a resistor.

Example 3.4

Explain what happens to the V_{out} waveforms in Figs. 3.8(b) and (c) if W/L is doubled.

Solution

Since the transistor becomes stronger, we expect that C_1 charges or discharges faster (Fig. 3.9). Note, however, that V_{out} still approaches $V_{DD} - V_{TH}$ in the case of $V_{in} = V_{DD}$.

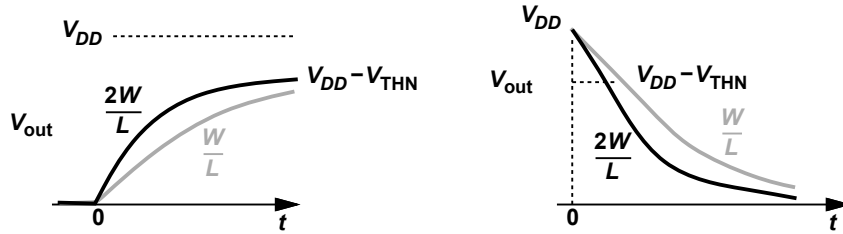


Figure 3.9 Sampling waveforms for different switch W/L ratios.

Example 3.5

In the circuit of Fig. 3.10(a), V_X varies from 0 to V_{DD} . Plot the on-resistance of M_1 as a function of V_X , noting that the transistor resides in the deep-triode region for $V_X < V_{DD} - V_{TH}$ (why?).

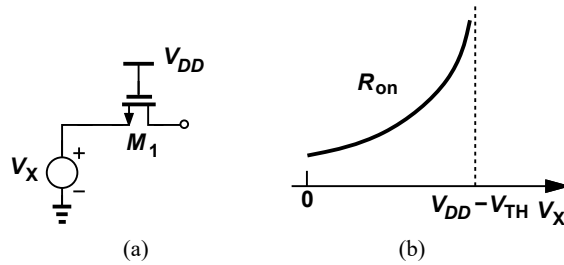


Figure 3.10 (a) MOS device acting as a resistor, and (b) its resistance vs. V_X .

Solution

The reader should not be puzzled by the fact that the drain of M_1 is not attached to any other devices. Since $I_D = 0$, the source and drain voltages are equal and M_1 acts as a resistor. In other words, a MOSFET can be on and carry no current. We express the on-resistance as

$$R_{on} = \frac{1}{\mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})} \quad (3.23)$$

$$= \frac{1}{\mu_n C_{ox} \frac{W}{L} (V_{DD} - V_X - V_{TH})} \quad (3.24)$$

These equations are valid only for $V_{GS} \geq V_{TH}$. Plotted in Fig. 3.10(b), R_{on} rises sharply as $V_X \rightarrow V_{DD} - V_{TH}$. That is, the switch turns off.

Example 3.6

How can we construct the plot of Fig. 3.10(b) in circuit simulations?

Solution

We apply a constant small voltage difference (e.g., 10 mV) between the source and drain (Fig. 3.11) and plot V_1/I_1 versus V_X as V_X varies from zero to $V_{DD} - V_{TH}$.

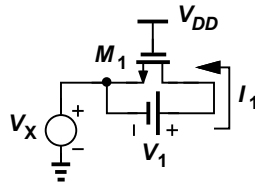


Figure 3.11 Circuit arrangement for measuring switch resistance.

Example 3.7

Can we always assume that MOS switches reside in the triode region during sampling?

Solution

As seen in Figs. 3.8(b) and (c), the MOS device can operate in saturation for some time. Nonetheless, if the circuit acquires the input successfully, then the output voltage is eventually close to the input voltage and $|V_{DS}|$ is small, driving the MOSFET into the deep-triode region. This is always true near the end of the acquisition phase (in a well-designed sampler).

The foregoing study indicates that a single NMOS or PMOS switch does not allow a sampler's output to track a rail-to-rail input faithfully. In fact, even if V_{in} in Fig. 3.8(b) is somewhat lower than $V_{DD} - V_{TH}$, the performance degrades because of the low overdrive voltage of M_1 . To understand this point, let us conduct experiments in which V_{in} is a low-swing sinusoid, $V_1 \cos \omega_1 t$, riding atop a dc level, V_0 . As illustrated in Fig. 3.12(a), we first select V_0 slightly above zero. After the acquisition transient subsides, V_{out} simply follows

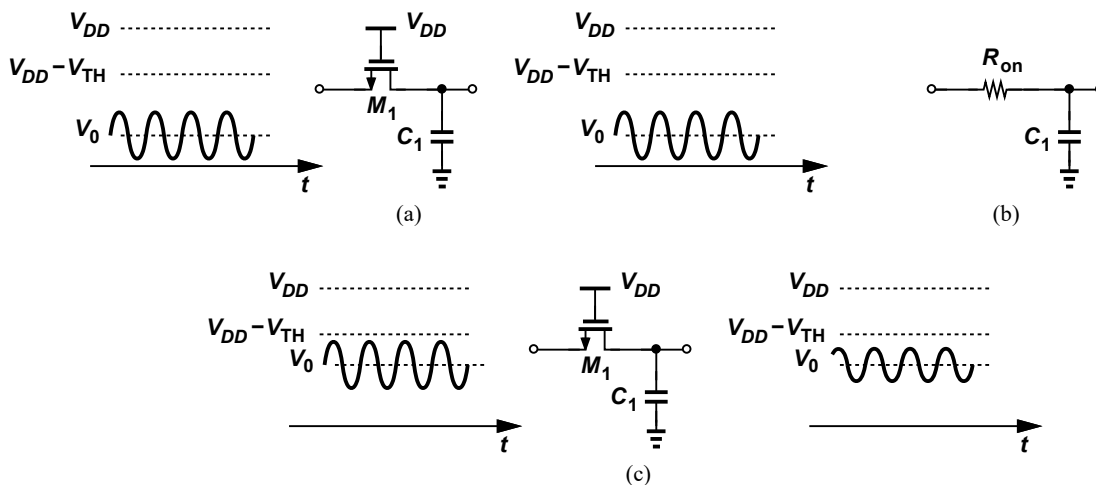


Figure 3.12 (a) Sampling circuit sensing an input near ground, (b) equivalent circuit, and (c) the case of input near $V_{DD} - V_{TH}$.

V_{in} , i.e., $V_{out} \approx V_{in}$ and M_1 operates in the deep-triode region. We can thus approximate its on-resistance as

$$R_{on} = \frac{1}{\mu_n C_{ox} \frac{W}{L} (V_{DD} - V_0 - V_{TH})}, \quad (3.25)$$

where the sinusoidal fluctuation is neglected. The circuit reduces to that in Fig. 3.12(b), where the RC network causes attenuation and phase shift:

$$V_{out}(t) = V_0 + \frac{V_1}{\sqrt{R_{on}^2 C_1^2 \omega_1^2 + 1}} \cos[\omega_1 t - \tan^{-1}(R_{on} C_1 \omega_1)]. \quad (3.26)$$

If $R_{on} C_1$ is sufficiently small, both the attenuation and the phase shift are negligible. For this reason, we prefer a high overdrive, $V_{DD} - V_0 - V_{TH}$, in Eq. (3.25).

In the second experiment, we choose V_0 slightly below $V_{DD} - V_{TH}$ [Fig. 3.12(c)], recognizing that the transistor overdrive, $V_{DD} - V_0 - V_{TH}$, is small and R_{on} large. Consequently, the output carries a heavily attenuated and phase-shifted copy of the input.

Let us summarize our thoughts. Even if the sampling switch does not turn off and even if we allow sufficient time for acquisition, the circuit of Fig. 3.8(a) still behaves differently for signal levels near ground or near $V_{DD} - V_{TH}$. We expect that the variation of the switch on-resistance leads to distortion. We also predict that the distortion is frequency-dependent; after all, if the sinusoids in Figs. 3.12(a) and (c) vary slowly, the attenuation and phase shift in Eq. (3.26) become negligible, yielding the same outputs in the two experiments. We formulate these concepts in the next section.

3.3 Distortion Due to Switch Resistance

Recall from Chapter 2 that nonlinearity in a system generally translates to harmonic distortion. If the nonlinearity is frequency-dependent, we cannot characterize it by plotting the output versus the input because the result would depend on how *fast* the input is varied. This is the case for the sampling circuit studied in the previous section. Since this topology does not lend itself to a polynomial characteristic, we resort to the use of an input sinusoid and seek the output harmonic distortion.

From the small-signal computations in the previous section, we can predict the large-signal behavior of the sampling circuit. As shown in Fig. 3.13, we apply an input sinusoid swinging from zero to slightly below

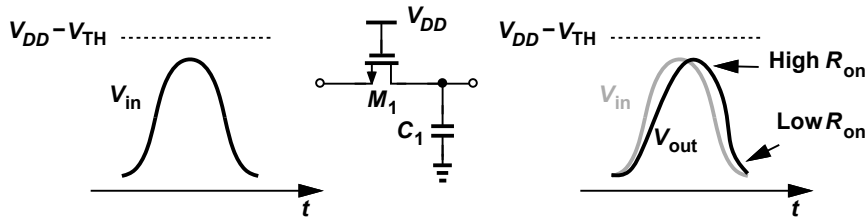


Figure 3.13 Effect of variable switch resistance.

$V_{DD} - V_{TH}$. Recall from Eq. (3.26) that the signal experiences both attenuation and phase shift. Let us focus on the latter for now and recognize that it is *greater* at higher input levels. Thus, the input-dependent phase shift distorts V_{out} .

We wish to formulate the distortion in the acquisition mode. The key point that will emerge from our analysis is that, even if we select the circuit's time constant small enough to have $R_{on}^2 C_1^2 \omega^2 \ll 1$ in Eq. (3.26), i.e., even if the attenuation is negligible, the input-dependent phase shift, $-\tan^{-1}(R_{on} C_1 \omega_1)$, can still lead to significant distortion.

We now proceed with the analysis by observing that, for a periodic input, the switch on-resistance in Fig. 3.13 also varies periodically (Fig. 3.14). We can therefore express R_{on} as a Fourier series:

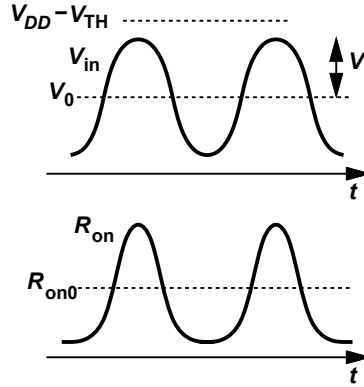


Figure 3.14 Variation of switch on-resistance with time.

$$R_{on} = R_{on0} + a_1 \cos \omega_1 t + a_2 \cos 2\omega_1 t + \dots, \quad (3.27)$$

where R_{on0} denotes the average value. Let us then return to Eq. (3.26), assume $R_{on}^2 C_1^2 \omega_1^2 \ll 1$ and also $\tan^{-1} R_{on} C_1 \omega_1 \ll 1$ rad so that $\tan^{-1} R_{on} C_1 \omega_1 \approx R_{on} C_1 \omega_1$:

$$V_{out}(t) \approx V_0 + V_1 \cos(\omega_1 t - R_{on} C_1 \omega_1) \quad (3.28)$$

$$\approx V_0 + V_1 \cos[\omega_1 t - C_1 \omega_1 (R_{on0} + a_1 \cos \omega_1 t + a_2 \cos 2\omega_1 t + \dots)]. \quad (3.29)$$

Expanding the cosine and writing $\cos \epsilon \approx 1$ and $\sin \epsilon \approx \epsilon$ for $\epsilon \ll 1$ rad, we have

$$V_{out}(t) \approx V_0 + V_1 \cos \omega_1 t + V_1 (\sin \omega_1 t) [C_1 \omega_1 (R_{on0} + a_1 \cos \omega_1 t + a_2 \cos 2\omega_1 t + \dots)]. \quad (3.30)$$

As expected, $V_{out}(t)$ contains all harmonics of the signal:

$$V_{out}(t) \approx V_0 + V_1 \cos \omega_1 t + V_1 R_{on0} C_1 \omega_1 \sin \omega_1 t + \frac{V_1 a_1 C_1 \omega_1}{2} \sin 2\omega_1 t + \frac{V_1 a_2 C_1 \omega_1}{2} (\sin 3\omega_1 t - \sin \omega_1 t) + \dots \quad (3.31)$$

We say the signal experiences “harmonic distortion” (HD).

Interestingly, the amplitude of the fundamental is no longer equal to V_1 because of the new terms involving ω_1 . Nevertheless, in a good design, we have $R_{on0} C_1 \omega_1 \ll 1$ and $a_2 C_1 \omega_1 / 2 \ll 1$; thus, the fundamental amplitude is close to V_1 . The second and third harmonic amplitudes normalized to that of the fundamental are given respectively by:

$$\text{HD}_2 = \frac{a_1 C_1 \omega_1}{2}, \quad (3.32)$$

$$\text{HD}_3 = \frac{a_2 C_1 \omega_1}{2}. \quad (3.33)$$

Example 3.8

If the periodic R_{on} waveform in Fig. 3.14 is approximated by a sinusoid swinging between R_1 and R_2 (Fig. 3.15). Determine the resulting distortion.

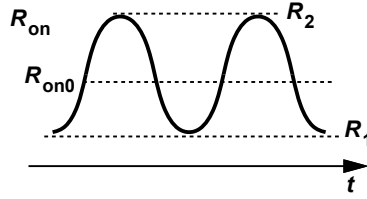


Figure 3.15 Sinusoidal approximation of R_{on} .

Solution

We observe that, in Eq. (3.27), $R_{on0} = (R_2 + R_1)/2$ and a_1 is equal to the peak amplitude of the sinusoid, $a_1 = (R_2 - R_1)/2$. Thus,

$$HD_2 = \frac{(R_2 - R_1)C_1\omega_1}{4}. \quad (3.34)$$

Higher harmonics are absent in this case.

Example 3.9

Explain what happens to HD_2 and HD_3 if the width of M_1 in Fig. 3.13 is doubled.

Solution

Since $R_{on} \propto (W/L)^{-1}$, the on-resistance is halved for all input levels (Fig. 3.16). The Fourier expansion in (3.27) is therefore divided by 2. It follows that a_1 and a_2 are halved and so are HD_2 and HD_3 . We then observe

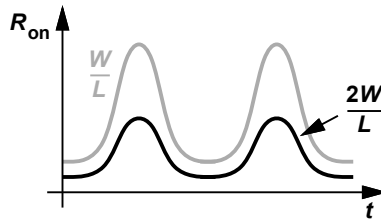


Figure 3.16 Switch on-resistance for different W/L ratios.

that wider switches are preferable. Of course, as we increase W , the drain junction capacitance (which adds to C_1) also rises, introducing its own nonlinearity and hence input-dependent phase shift. We return to this point in Section 3.4.

3.4 Complementary Sampling Switches

The performance provided by a single MOS switch is typically inadequate. In order to construct a sampling circuit that accommodates rail-to-rail swings, let us first repeat the experiment shown in Fig. 3.10 for a PMOS device. Depicted in Fig. 3.17(a), the on-resistance is given by

$$R_{on} = \frac{1}{\mu_p C_{ox} \left(\frac{W}{L} \right)_P (V_{in} - |V_{THP}|)}, \quad (3.35)$$

approaching infinity as V_{in} falls toward $|V_{THP}|$. The complementary shapes of R_{on} in Figs. 3.10(b) and 3.17(a) suggest that we can place the two transistor types in parallel [Fig. 3.17(b)] so as to allow V_{in} to reach

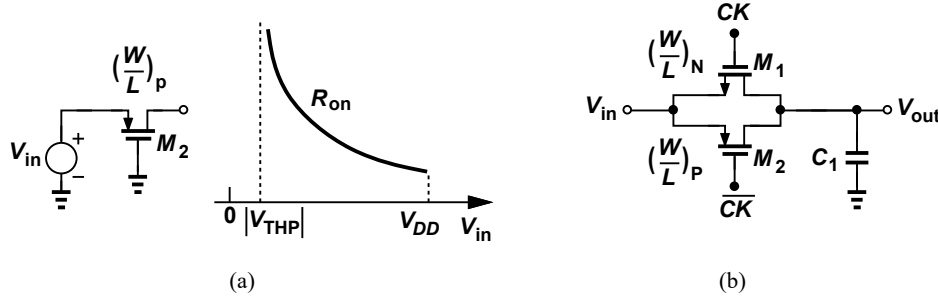


Figure 3.17 (a) PMOS device acting as a resistor, and (b) use of complementary devices to avoid infinite on-resistance.

zero and V_{DD} . We have

$$R_{on1} || R_{on2} = \frac{1}{\mu_n C_{ox} \left(\frac{W}{L}\right)_N (V_{DD} - V_{in} - V_{THN})} || \frac{1}{\mu_p C_{ox} \left(\frac{W}{L}\right)_P (V_{in} - |V_{THP}|)} \quad (3.36)$$

$$= \frac{1}{C_{ox}} \cdot \frac{1}{D}, \quad (3.37)$$

where

$$D = \mu_n \left(\frac{W}{L}\right)_N V_{DD} + \left[\mu_p \left(\frac{W}{L}\right)_P - \mu_n \left(\frac{W}{L}\right)_N \right] V_{in} - \mu_n \left(\frac{W}{L}\right)_N V_{THN} - \mu_p \left(\frac{W}{L}\right)_P |V_{THP}|. \quad (3.38)$$

Interestingly, it appears that, if we choose $\mu_p(W/L)_P = \mu_n(W/L)_N$, i.e., if M_1 and M_2 have equal strengths. The total resistance becomes independent of V_{in} , and the distortion mechanism studied in the previous section vanishes.

Example 3.10

Explain intuitively why $R_{on1} || R_{on2}$ becomes independent of V_{in} if M_1 and M_2 have equal strengths. Does this occur for the entire range of V_{in} ?

Solution

We can view the denominators of R_{on1} and R_{on2} as *conductances*, noting that they simply add because they are in parallel. The NMOS conductance falls linearly with V_{in} , whereas that of the PMOS device rises linearly. Thus, proper choice of their slopes results in a constant sum. However, as illustrated in Fig. 3.18, this cancelation holds only for $|V_{THP}| < V_{in} < V_{DD} - V_{THN}$. Equation (3.37) fails to predict the behavior of $R_{on1} || R_{on2}$ outside this range because the expressions for R_{on1} and R_{on2} are only valid for $V_{in} < V_{DD} - |V_{THN}|$ and $|V_{THP}| < V_{in}$, respectively.

The foregoing example reveals that $R_{on1} || R_{on2}$ in Fig. 3.17(b) still varies even if M_1 and M_2 have equal strengths. In reality, the situation is more complex because of short-channel effects. We expect the general behavior depicted in Fig. 3.19, where $R_{on1} || R_{on2}$ rises from R_1 to R_{max} and then falls to R_2 .

Example 3.11

Consider the arrangement shown in Fig. 3.20(a) and assume $R_1 \approx R_2$ in Fig. 3.19. If we have $V_{in} = V_{DD}/2 - (V_{DD}/2) \cos \omega_1 t$, sketch $R_{on1} || R_{on2}$ as a function of time.

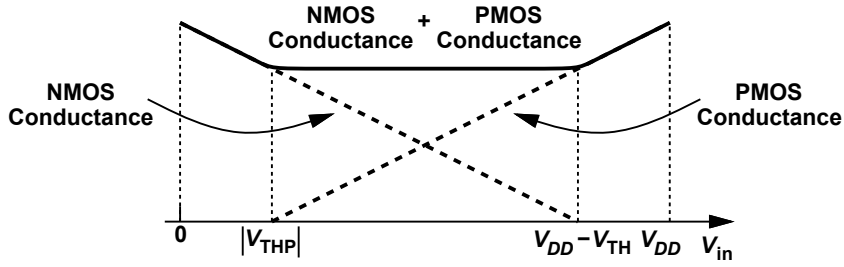


Figure 3.18 Conductances of NMOS, PMOS, and complementary switches vs. the input voltage.

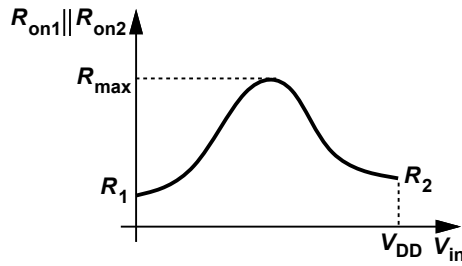


Figure 3.19 General behavior of a complementary switch.

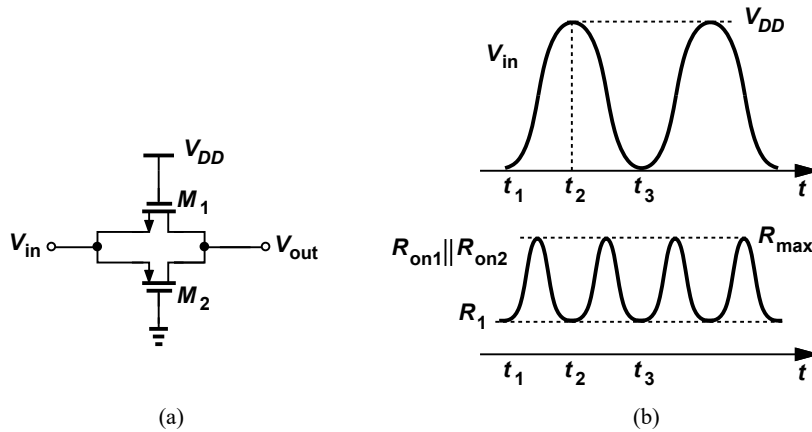


Figure 3.20 (a) Arrangement for studying on-resistance of complementary switches, and (b) variation of on-resistance with time.

Solution

As V_{in} rises from 0 to V_{DD} , $R_{on1} || R_{on2}$ rises and falls once. That is, from t_1 to t_2 , the switch resistance makes a full cycle [Fig. 3.20(b)]. We conclude that the resistance varies at a rate of $2\omega_1$ if $R_1 \approx R_2$.

Distortion Analysis We can repeat the analysis of Section 3.3 for the sampling circuit shown in Fig. 3.17(b), bearing in mind that the switch resistance varies as depicted in Fig. 3.20(b). We assume a rail-to-rail input: $V_{in} = V_{DD}/2 + (V_{DD}/2) \cos \omega_1 t$. We also assume that $R_1 \approx R_2$ and hence $R_{on1} || R_{on2}$ has a

fundamental frequency equal to $2\omega_1$. Expressing the resistance as a Fourier series,

$$R_{on1}||R_{on2} = R_{on0} + a_1 \cos 2\omega_1 t + a_2 \cos 4\omega_1 t + \dots, \quad (3.39)$$

we rewrite Eq. (3.30) for this case:

$$V_{out}(t) \approx \frac{V_{DD}}{2} + \frac{V_{DD}}{2} \cos \omega_1 t + \frac{V_{DD}}{2} (\sin \omega_1 t) [C_1 \omega_1 (R_{on0} + a_1 \cos 2\omega_1 t + a_2 \cos 4\omega_1 t + \dots)]. \quad (3.40)$$

The circuit therefore generates only odd harmonics:

$$HD_3 = \frac{a_1 C_1 \omega_1}{2}, \quad (3.41)$$

$$HD_5 = \frac{a_2 C_1 \omega_1}{2}. \quad (3.42)$$

Example 3.12

Approximating the on-resistance waveform in Fig. 3.20(b) by a sinusoid, determine a_1 .

Solution

The value of a_1 is given by the amplitude of $R_{on1}||R_{on2}$, namely, $(R_{max} - R_1)/2$.

How serious is the distortion problem here? Recall from Example 3.3 that a 10-bit ADC requires a sampling time constant less than $T_{CK}/15.24$. Let us assume in Fig. 3.21 that $a_1 = R_{on0} - R_1$. For HD_3 to

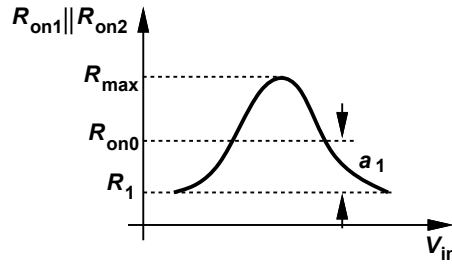


Figure 3.21 On-resistance variation example.

be at the 10-bit level, we first write from Chapter 1: $SNR = 6.02N + 1.74 \text{ dB} \approx 62 \text{ dB}$. Next, we assume the distortion is at the same level as the quantization noise and hence $HD_3 \approx -62 \text{ dB}$. If the input is at the Nyquist rate, i.e., if ω_1 in (3.41) is equal to $2\pi/(2T_{CK})$, we have

$$HD_3 = \frac{a_1 C_1}{2} \frac{2\pi}{2T_{CK}} \leq 8 \times 10^{-4}, \quad (3.43)$$

and hence

$$a_1 C_1 \leq \frac{T_{CK}}{2000}. \quad (3.44)$$

Notably, the *variation* of the time constant must be extremely small, motivating us to seek switches whose on-resistance does not vary as much as shown in Fig. 3.19.

Let us return to the complementary switches in Fig. 3.17(b) and ask what happens if we double their widths. The total on-resistance, $R_{on1}||R_{on2}$, falls by a factor of 2, and so do the Fourier coefficients in

Eq. (3.39). This means that the normalized distortion values in (3.41) and (3.42) are also halved if C_1 remains constant.

Can we continue to make the switches in Fig. 3.17(b) wider so as to reduce the distortion? We must take into account another effect, namely, that the junction capacitances of M_1 and M_2 also scale with their widths, introducing a proportionally greater *nonlinear* capacitance in parallel with C_1 . Shown in Fig. 3.22, C_{DB1} and C_{DB2} simply add to C_1 and vary periodically if V_{in} is periodic. Consequently, the phase shift in (3.28) becomes input-dependent even if R_{on} is constant. The reader can expand $C_1 + C_{DB1} + C_{DB2}$ as a Fourier series and repeat the foregoing calculations to find the harmonic distortion.

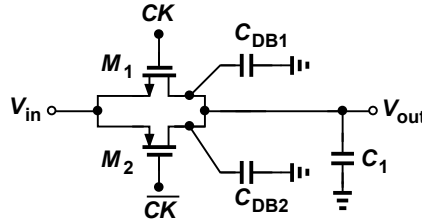


Figure 3.22 Effect of sampling devices' junction capacitances.

The key point here is that wider transistors reduce the on-resistance variation but raise the distortion due to C_{DB1} and C_{DB2} . We therefore expect an optimum choice for W_1 and W_2 . However, as these widths increase, some other switch imperfections also manifest themselves, as studied in the next section.

3.5 Other Switch Imperfections

In addition to their input-dependent on-resistance, MOS sampling switches exhibit three other nonidealities: clock feedthrough, channel charge injection, and kT/C noise. We study these phenomena here. The reader is encouraged to review the concepts of offset, gain error, and nonlinearity in Chapter 2.

3.5.1 Review of MOS Capacitances

A MOSFET exhibits capacitance between its gate and its source and drain terminals even if it is off. Illustrated in Fig. 3.23(a), part of this capacitance arises from the physical overlap between the gate material and the source and drain diffusions, and another from the fringe electric field lines. In practice, these capacitances incur additional components because the metal contacts on the source and drain act as tall walls, creating fringe fields to the gate [Fig. 3.23(b)]. These components are lumped on each side, called the “overlap capacitance,” and modeled as shown in Fig. 3.23(c). Here, C_{ov} represents the total off-state capacitance per unit width between the gate and the source (or between the gate and the drain).

In the deep-triode region, a MOSFET exhibits equal gate-source and gate-drain capacitances. As shown in Fig. 3.24(a), the total gate-channel capacitance, WLC_{ox} , is lumped into two equal components. Taking into account the overlap capacitances as well, we arrive at the model in Fig. 3.24(b) for deep-triode operation. Note that C_{GS} and C_{GD} are proportional to W .

3.5.2 Clock Feedthrough

Consider the sampling circuit shown in Fig. 3.25(a), where $V_{in} = 0.5$ V and M_1 is initially on. At $t = t_1$, the gate voltage begins to fall. We expect that C_{GD} couples this falling transition to C_1 , affecting the sampled voltage. From Fig. 3.25(b), we view C_{GD} and C_1 as a capacitive voltage divider and write

$$\Delta V = -\frac{C_{GD}}{C_{GD} + C_1} V_{DD}. \quad (3.45)$$

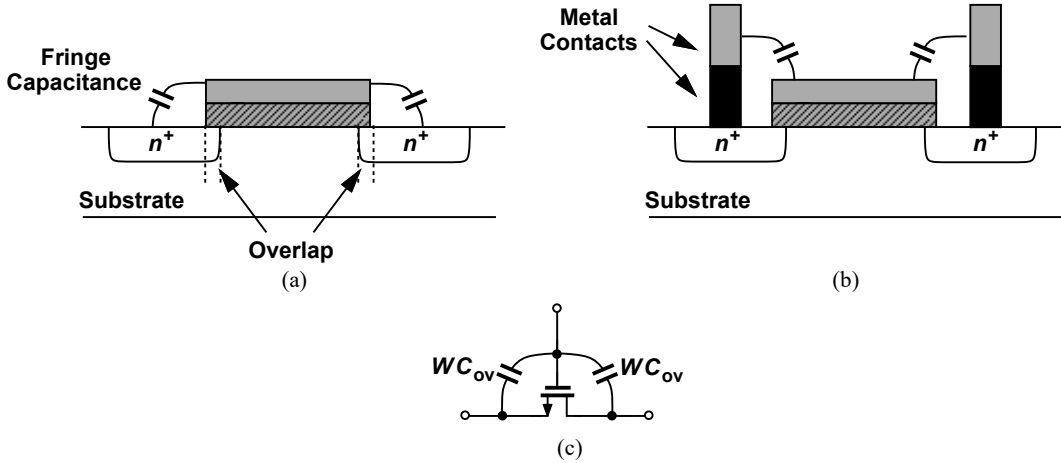


Figure 3.23 (a) MOS device with its overlap and fringe capacitances, (b) additional fringe capacitances due to contacts, and (c) equivalent circuit (excluding other capacitances).

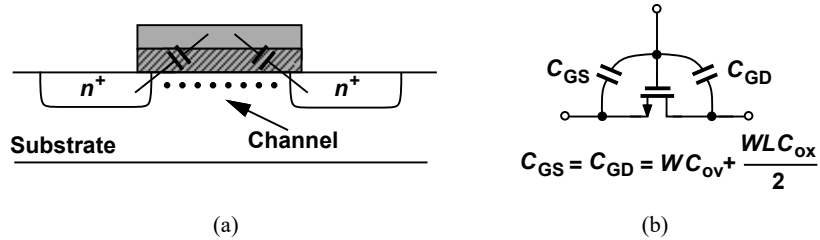


Figure 3.24 (a) Gate capacitance partitioning between source and drain, and (b) equivalent circuit model in the deep-triode region.

Called the “clock feedthrough,” this “pedestal” alters the voltage sampled on C_1 . It appears that ΔV is independent of V_{in} and acts as a constant *offset*, a good approximation since C_{GD} exhibits little voltage dependence.

In summary, clock feedthrough introduces a constant offset. Thus, widening the switch to reduce its on-resistance inevitably exacerbates this effect. Moreover, the error can be reduced by increasing C_1 , but at the cost of acquisition speed. We say the sampling circuit suffers from a trade-off between speed and precision.

3.5.3 Channel Charge Injection

Recall from device physics that a MOS device turns on when a channel of charge is formed below the gate oxide. For example, an NMOS transistor acting as a switch contains a channel of electrons when it is on—even if it carries no current (Example 3.5). Now, consider the situation shown in Fig. 3.26, where an NMOS switch is on, allowing V_{out} to reach $V_{in} = 0.5$ V. What happens when the gate voltage of M_1 falls to zero?

For the transistor to turn off, its channel charge must depart. The charge exits through the source and drain terminals, causing a pedestal, ΔV , in the voltage stored on C_1 . The splitting of the channel charge between the source and the drain depends on the symmetry of the impedances seen at these two nodes and some other factors. As a worst-case estimate, we assume all of the charge lands on C_1 .

We must now determine what type of corruption this charge creates: offset, gain error, or nonlinearity. To this end, we express the channel charge as

$$Q_{ch} = WLC_{ox}(V_{GS} - V_{TH}), \quad (3.46)$$

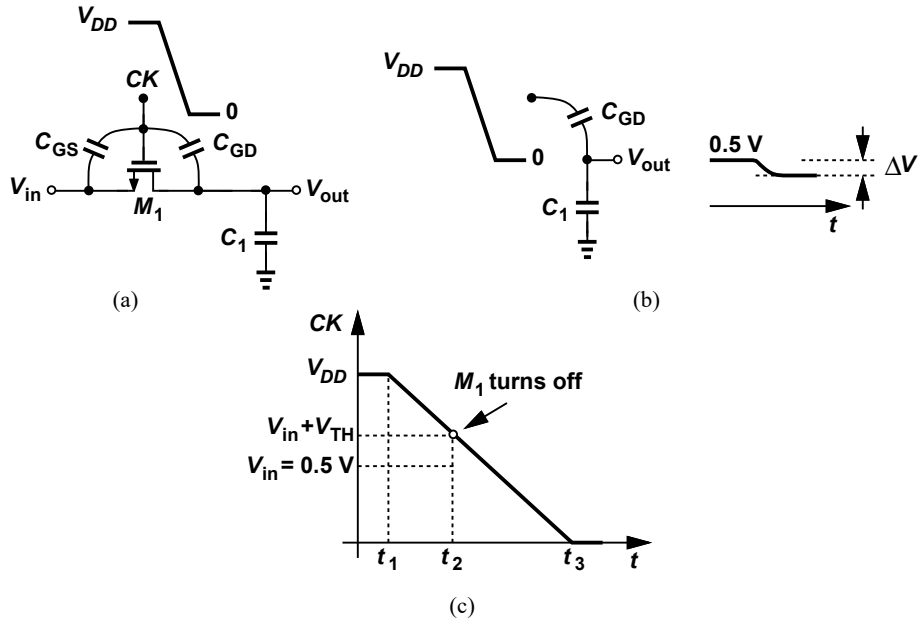


Figure 3.25 (a) Sampling circuit, (b) illustration of clock feedthrough, and (c) effect of finite clock fall time.

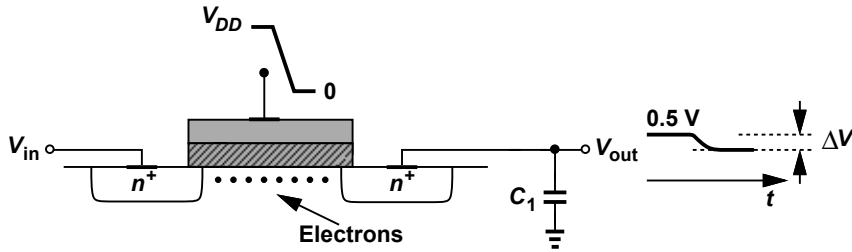


Figure 3.26 Channel charge injection onto sampling capacitor.

recognizing that $Q_{ch} = 0$ if $V_{GS} - V_{TH} = 0$ and the device is off. Note that $V_{GS} = V_{DD} - V_{in}$. If all of this charge is injected onto C_1 in Fig. 3.26, we have

$$\Delta V = \frac{WLC_{ox}(V_{GS} - V_{TH})}{C_1} \quad (3.47)$$

$$= \frac{WLC_{ox}(V_{DD} - V_{in} - V_{TH})}{C_1}. \quad (3.48)$$

The above equation merits two remarks. First, the constant (input-independent) terms on the right-hand side correspond to an offset in the sampler's input-output characteristic. For example, the term including V_{DD} represents a constant jump at the output. Second, the terms that depend on V_{in} reveal an input-dependent jump at the output. If this jump is linearly proportional to V_{in} , then it signifies a gain error. For example, the term on the right-hand side that contains V_{in} leads to such an effect. But are there any terms that are a *nonlinear* function of V_{in} ? We recognize that, in the presence of body effect, V_{TH} varies with V_{in} and V_{out} according to

$$V_{TH} = V_{TH0} + \gamma(\sqrt{V_{in} + 2\phi_B} - \sqrt{2\phi_B}), \quad (3.49)$$

where V_{TH0} denotes the threshold when the source-bulk voltage, V_{in} , is zero, γ is the body effect coefficient, and ϕ_B is the built-in potential. Thus, body effect creates a small nonlinear component in ΔV .

In a manner similar to the clock feedthrough effect in the previous section, charge injection imposes a speed-precision trade-off on the sampling circuit. In fact, if we write the switch on-resistance as $[\mu_n(W/L)C_{ox}(V_{GS} - V_{TH})]^{-1}$ and multiply it by C_1 to obtain the acquisition time constant, τ , the reader can prove that

$$\Delta V = \frac{L^2}{\mu_n \tau}. \quad (3.50)$$

In other words, the product of the error, ΔV , and τ is equal to L^2/μ_n and only a function of the CMOS technology in which the circuit is designed.

3.5.4 kT/C Noise

Recall from Example 3.7 that the MOSFET in a sampling circuit operates in the deep-triode region before the acquisition mode ends because the input and output voltages are nearly equal. We can therefore model the transistor by an on-resistance, R_{on} , and attribute to it a thermal noise spectrum given by $4kTR_{on}$, where $k = 1.38 \times 10^{-23}$ J/K and T is the absolute temperature [Fig. 3.27(a)]. To obtain the output noise spectrum, we set the (ac) value of V_{in} to zero and note from Fig. 3.27(b) that the noise experiences a low-pass response.

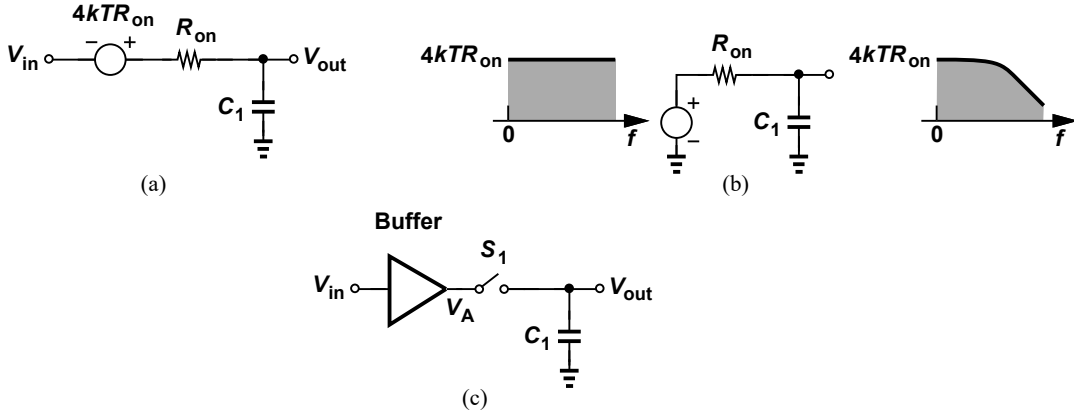


Figure 3.27 (a) Sampling circuit including on-resistance noise, (b) simplified model, and (c) effect of input buffer.

To compute the output noise spectrum, $\overline{V_{n,out}^2}$, we multiply $4kTR_{on}$ by the squared magnitude of the transfer function:¹

$$\overline{V_{n,out}^2}(f) = 4kTR_{on} \frac{1}{R_{on}^2 C_1^2 (2\pi f)^2 + 1}. \quad (3.51)$$

The low-pass spectrum is relatively flat within the bandwidth of the circuit. When the switch turns off, the noise voltage is stored on C_1 , thereby corrupting the sampled signal. We wish to find the rms value of this sampled noise. As an approximation of the sampled noise, we simply calculate the area under the output spectrum given by (3.51) and take the square root. We have

$$\overline{V_{n,out,tot}^2} = \int_0^\infty \frac{4kTR_{on}}{R_{on}^2 C_1^2 (2\pi f)^2 + 1} df. \quad (3.52)$$

¹The magnitude of a complex number $a + jb$ is equal to $\sqrt{a^2 + b^2}$.

Noting that

$$\int \frac{dx}{x^2 + 1} = \tan^{-1} x, \quad (3.53)$$

We obtain

$$\overline{V_{n,out,tot}^2} = \frac{kT}{C_1}, \quad (3.54)$$

and

$$V_{n,out,rms} = \sqrt{\frac{kT}{C_1}}. \quad (3.55)$$

Interestingly, the result is independent of the switch on-resistance. As an example, if $T = 300$ K and $C_1 = 1$ pF, the sampled output noise is $64 \mu\text{V}_{\text{rms}}$.

The problem of kT/C noise proves serious. Since the sampler noise voltage must remain well below 1 LSB of the overall ADC, we must choose C_1 large enough, thus making the design of the *preceding* stage more difficult. Depicted in Fig. 3.27(c) is a typical arrangement where a voltage buffer drives the sampler and must deliver V_A to C_1 with acceptably low distortion.

Example 3.13

If V_A in Fig. 3.27(c) is given by $V_0 + V_1 \cos \omega_1 t$, how much is the maximum current that the buffer must provide to C_1 ?

Solution

We have $I = C_1 dV_A/dt = -\omega_1 V_1 C_1 \sin \omega_1 t$. The maximum current is therefore equal to $C_1 \omega_1 V_1$. For example, with $C_1 = 1$ pF, $\omega_1 = 2\pi(1 \text{ GHz})$, and $V_1 = 0.3$ V, we obtain $I = 1.9$ mA. If the buffer fails to provide this amount of current, V_A cannot follow V_{in} and the buffer “slews,” causing distortion in V_A .

Example 3.14

The buffer in Fig. 3.27(c) is realized by a source follower [Fig. 3.28(a)]. Identify the distortion mechanisms in the acquisition mode.

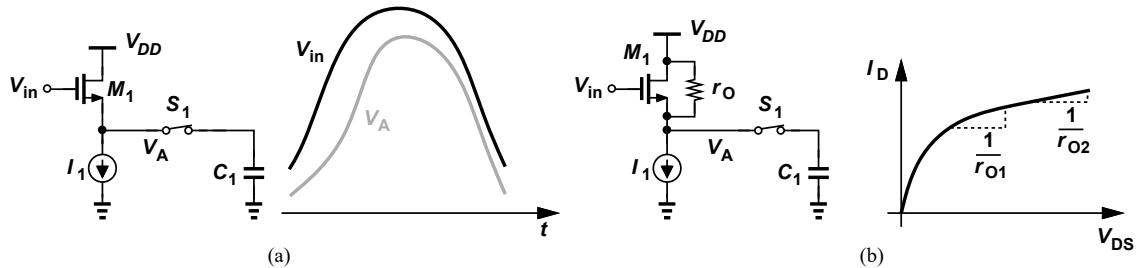


Figure 3.28 (a) Source follower acting as a buffer, and (b) effect of voltage-dependent r_O .

Solution

Since the current delivered to or taken from C_1 by the source follower depends on the slope of V_A , we observe

that when V_{in} rises, M_1 must provide both I_1 and $C_1 dV_A/dt$, thereby exhibiting a greater V_{GS} . Conversely, when V_{in} falls, V_{GS} decreases. Thus, the input-output level shift varies with the slope of the signal, distorting V_A .

If M_1 experiences body effect, then it has a larger threshold for higher input levels, and hence a greater V_{GS} . In this case, the level shift varies with the signal voltage, also distorting V_A .

A third mechanism relates to the output resistance of M_1 , which *varies* to some extent with V_{DS} . As shown in Fig. 3.28(b), the slope of a MOSFET's I_D - V_{DS} characteristic decreases with V_{DS} even in the saturation region. Consequently, the voltage gain of the source follower, $r_O/(r_O + 1/g_m)$, becomes input-dependent, introducing distortion.

3.6 Differential Sampling

We know from basic analog design that differential operation is generally preferred. In this spirit, we construct the differential sampling circuit shown in Fig. 3.29(a). Here, V_{in1} and V_{in2} vary differentially and have

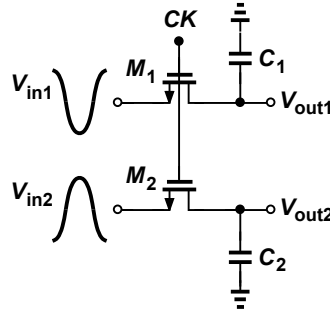


Figure 3.29 Differential sampling circuit.

the same common-mode (CM) level, and so do V_{out1} and V_{out2} . We assume $V_{in1} = V_0 + V_1 \cos \omega_1 t$ and $V_{in2} = V_0 - V_1 \cos \omega_1 t$. An immediately obvious advantage of differential operation is that the total swing provided by $V_{out1} - V_{out2}$ is twice that of each single-ended output. That is, the LSB size of the subsequent quantizer can be doubled, leading to certain design simplifications (Chapter 13).

It is instructive to revisit the MOS switch imperfections described previously in the context of this topology. We consider four phenomena: on-resistance nonlinearity, clock feedthrough, charge injection, and kT/C noise.

The on-resistance of M_1 and M_2 still varies with each single-ended input as computed in Section 3.3. However, since V_{in1} and V_{in2} are 180° out of phase, so are the periodic functions representing R_{on1} and R_{on2} . Before writing these functions, we should ask: what happens to the Fourier series of a periodic waveform if it is shifted in time by half of its period? We expect that the average does not change, the first harmonic is shifted by 180° , the second harmonic by 360° , the third harmonic by $3 \times 180^\circ$, etc. That is, only the odd harmonics are multiplied by -1 . We then have from (3.27):

$$R_{on1} = R_{on0} + a_1 \cos \omega_1 t + a_2 \cos 2\omega_1 t + a_3 \cos 3\omega_1 t + \dots, \quad (3.56)$$

$$R_{on2} = R_{on0} - a_1 \cos \omega_1 t + a_2 \cos 2\omega_1 t - a_3 \cos 3\omega_1 t + \dots \quad (3.57)$$

Writing (3.31) for V_{out1} and V_{out2} , we obtain

$$\begin{aligned} V_{out1}(t) \approx & V_0 + V_1 \cos \omega_1 t + V_1 R_{on0} C_1 \omega_1 \sin \omega_1 t, \\ & + \frac{V_1 a_1 C_1 \omega_1}{2} \sin 2\omega_1 t + \frac{V_1 a_2 C_1 \omega_1}{2} (\sin 3\omega_1 t - \sin \omega_1 t) + \dots \end{aligned} \quad (3.58)$$

and

$$V_{out2}(t) \approx V_0 - V_1 \cos \omega_1 t - V_1 R_{on0} C_1 \omega_1 \sin \omega_1 t + \frac{V_1 a_1 C_1 \omega_1}{2} \sin 2\omega_1 t - \frac{V_1 a_2 C_1 \omega_1}{2} (\sin 3\omega_1 t - \sin \omega_1 t) + \dots \quad (3.59)$$

It follows that

$$V_{out1}(t) - V_{out2}(t) = 2V_1 \cos \omega_1 t + 2V_1 R_{on0} C_1 \omega_1 \sin \omega_1 t + V_1 a_2 C_1 \omega_1 (\sin 3\omega_1 t - \sin \omega_1 t) + \dots \quad (3.60)$$

The even harmonics are absent, another advantage of differential operation, but the third harmonic amplitude normalized to that of the fundamental remains unchanged:

$$HD_3 = \frac{a_2 C_1 \omega_1}{2}. \quad (3.61)$$

The reader is encouraged to repeat this analysis for the behavior expressed by Eq. (3.39).

Let us now turn to the problem of clock feedthrough. As depicted in Fig. 3.30, V_{out1} and V_{out2} experience a pedestal as the clock falls, i.e., this effect primarily introduces a CM jump at the output. Unlike the single-

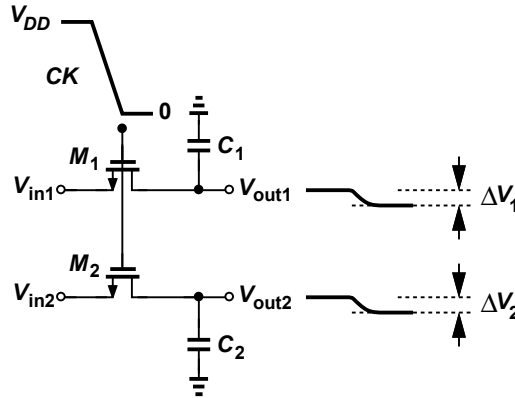


Figure 3.30 Effect of clock feedthrough in differential sampling.

ended circuit of Fig. 3.25(a), this topology does not suffer from an offset.²

The next effect is channel charge injection. Denoting the corresponding output pedestals by ΔV_1 and ΔV_2 , we have from (3.48):

$$\Delta V_1 - \Delta V_2 = \frac{WLC_{ox}}{C_1} (V_{in2} - V_{in1} + V_{TH2} - V_{TH1}). \quad (3.62)$$

The resulting gain error and nonlinearity are similar to those arising from clock feedthrough.

The last imperfection is kT/C noise. Since the noise of M_1 and the noise of M_2 in Fig. 3.29 are uncorrelated, we can write the total noise power in $V_{out1} - V_{out2}$ as $2kT/C_1$. However, differential operation allows us to double the output voltage swing for the signal. We can then say that the signal power, P_{sig} , is quadrupled whereas the noise power, P_{noise} , is doubled, yielding a twofold increase in the signal-to-noise ratio. This is yet another advantage of differential circuits.

In summary, the differential sampling circuit of Fig. 3.29 displays three advantages over its single-ended counterpart: (1) suppressed even harmonics, (2) no dc offset due to clock feedthrough and charge injection,

²Unless the mismatches between the differential paths are taken into account.

and (3) a greater SNR. We should point out that clock feedthrough and charge injection still cause a common-mode jump at the outputs, an issue that requires attention in low-voltage design. The following example illustrates this point.

Example 3.15

A sampler drives a differential pair as shown in Fig. 3.31. Explain why the CM jump at A and B is undesirable.

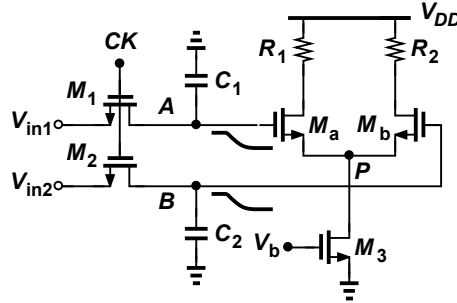


Figure 3.31 Differential sampler followed by a differential pair.

Solution

The downward jump at the gates of M_a and M_b reduces the tail voltage, V_P , by approximately the same value, possibly driving M_3 into the triode region. As a result, the CM rejection of the differential pair degrades and also the bias current of M_a and M_b decreases. For these reasons, we wish to minimize the CM jump.

3.7 Use of Dummy Switches

In order to reduce the common-mode jump studied in the previous section, we can add “dummy switches” to the differential sampler. Figure 3.32 shows one of the differential paths with the dummy device, M_3 , tied to

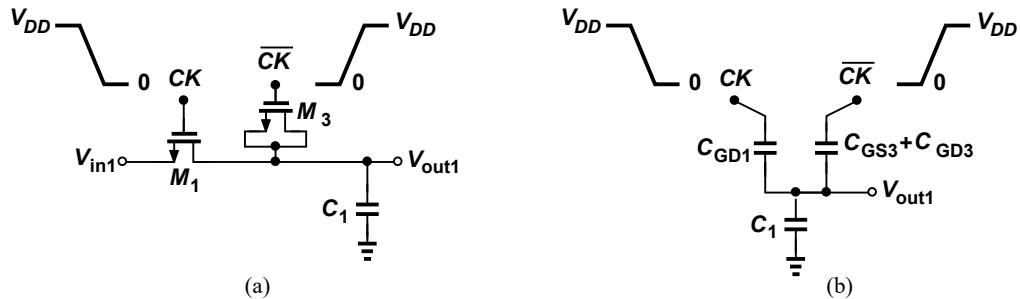


Figure 3.32 (a) Use of dummy transistor to cancel clock feedthrough, (b) equivalent circuit during turn-off.

the output and driven by $\overline{CK}$. When CK falls from V_{DD} to zero, the gate-drain capacitance of M_1 couples this transition to C_1 . At the same time, the gate of M_3 rises from zero to V_{DD} , taking charge out of C_1 through both its C_{GS} and its C_{GD} . The equivalent circuit in Fig. 3.32(b) then suggests that V_{out1} does not change if $C_{GD1} = C_{GD3} + C_{GS3}$. The symmetry of M_3 leads to $C_{GD3} = C_{GS3}$ and hence the condition $C_{GD1} = 2C_{GD3}$. We thus choose $W_3 = 0.5W_1$ and $L_3 = L_1$ for proper suppression of clock feedthrough.

Let us also examine the effect of the dummy switch in Fig. 3.32(a) on channel charge injection. We learned in Section 3.5.3 that M_1 injects its channel charge onto C_1 . In this case, as M_3 turns on, it requires channel

charge and draws it from C_1 . That is, some cancellation occurs. More specifically, if we assume that *half* of the channel charge of M_1 exits through its drain terminal and lands on C_1 , we can express this charge as $0.5W_1LC_{ox}(V_{DD} - V_{in1} - V_{TH})$. Transistor M_3 absorbs charge in the amount of $W_3LC_{ox}(V_{DD} - V_{out1} - V_{TH}) \approx W_3LC_{ox}(V_{DD} - V_{in1} - V_{TH})$. For these two charge packets to cancel each other, we require that $W_3 = 0.5W_1$, the same condition derived above for clock feedthrough suppression. Unfortunately, the cancellation of channel charge of M_1 depends on how it splits between the source and drain, which is difficult to define or even to model in simulations. We therefore do not rely much on dummy switches for charge injection cancellation.

3.8 Bootstrapped Sampling Circuit

In this section, we study a sampling circuit that has found wide application. The reader is encouraged to review Chapter 2 before proceeding here.

3.8.1 Preliminary Ideas

Our study of the switch on-resistance, clock feedthrough, and charge injection indicates that these effects introduce input-dependent errors primarily because the gate-source voltage, V_{GS} , varies with the input level. We then surmise that the performance can be improved if V_{GS} is maintained *constant* even though the input voltage changes with time. How is that possible? Let us conceptually place a battery between the gate and the source of the switch [Fig. 3.33(a)]. Now $V_{GS} = V_b$, a constant value. We say the gate is “bootstrapped”

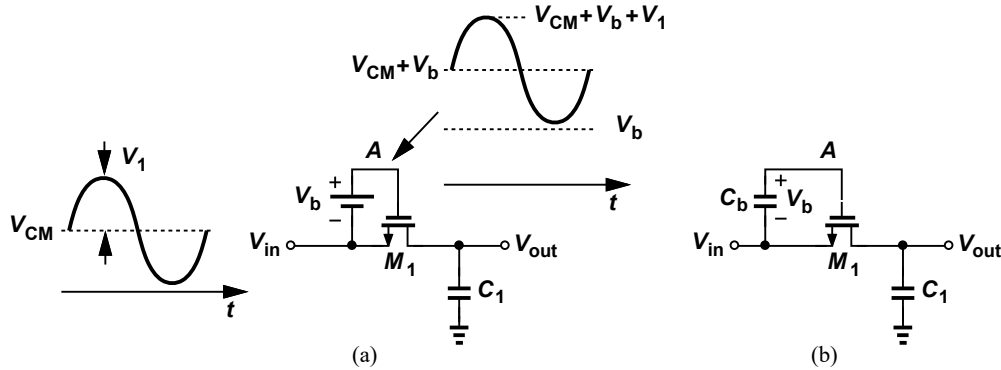


Figure 3.33 (a) Simple bootstrapped sampling switch, and (b) use of a capacitor to approximate the level-shift battery.

to the source. For example, if $V_b = V_{DD}$, V_{GS} is kept equal to V_{DD} as V_{in} swings up and down.

We should make three remarks. First, the battery shifts the common-mode level from V_{CM} to $V_{CM} + V_b$, causing the gate voltage to reach a peak value of $V_{CM} + V_b$ with respect to the ground. If $V_b = V_{DD}$, V_A rises *above* the supply voltage, a useful property in low-voltage design. Second, the battery keeps M_1 on and must be disconnected from it when we wish to turn this switch off. Third, with $V_b = V_{DD}$, M_1 benefits from a large overdrive and can therefore be narrower than a non-bootstrapped device.

In the next step, we must replace the battery with a more practical device. We approximate the battery by a charged capacitor [Fig. 3.33(b)], expecting that V_A still follows V_{in} with a constant CM difference equal to V_b . But the reader may think that the gate is a “floating” node as it is not tied to any other devices. The following examples provide a clearer understanding.

Example 3.16

Consider the simple circuit shown in Fig. 3.34. If V_{in} changes with time and C_b has an initial condition equal to V_b , can we say that V_A is always equal to $V_{in} + V_b$?

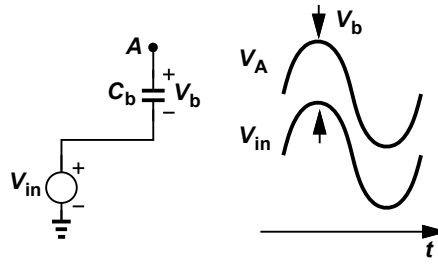


Figure 3.34 A floating capacitor driven by a voltage source.

Solution

Yes, we can. Since node A is not tied to any other devices, no current can flow through C_b . Thus, C_b maintains its charge, leading to $V_A = V_{in} + V_b$.

Example 3.17

In Fig. 3.35, C_p represents a small parasitic capacitance. Explain the behavior of V_A as V_{in} rises from zero. Assume C_b has a zero initial condition.

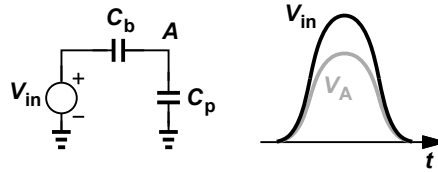


Figure 3.35 Effect of parasitic capacitance on bootstrapping.

Solution

In this case, C_b and C_p form a capacitive voltage divider, attenuating the input swings. To find V_A in terms of V_{in} , we recognize that the series combination of C_b and C_p is driven by V_{in} and receives charge in the amount of

$$Q = \frac{C_b C_p}{C_b + C_p} V_{in}. \quad (3.63)$$

This charge resides on the left plate of C_b and the top plate of C_p , yielding

$$V_A = \frac{Q}{C_p} \quad (3.64)$$

$$= \frac{C_b}{C_b + C_p} V_{in}. \quad (3.65)$$

Example 3.18

Repeat the foregoing example for the circuit shown in Fig. 3.36(a), where C_b has an initial condition equal to V_b , S_1 turns on at $t = 0$, and V_{in} begins from zero.

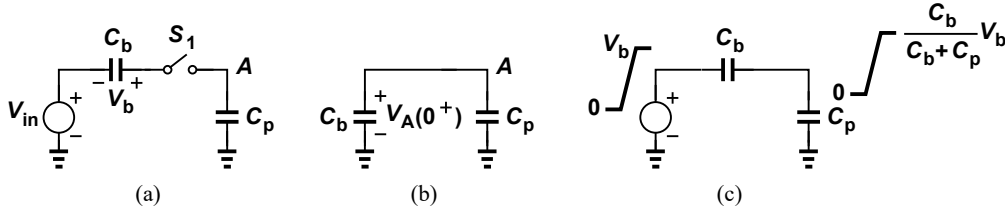


Figure 3.36 (a) Bootstrapping capacitor driving C_p , (b) its equivalent circuit, and (c) attenuation of voltage swing.

Solution

As S_1 turns on, C_b shares its charge with C_p . We draw the circuit at $t = 0^+$ as depicted in Fig. 3.36(b), assuming $V_{in}(0^+) = 0$, and equate the total charge on C_b and C_p to that on C_b before S_1 turns on:

$$(C_b + C_p)V_A(0^+) = C_b V_b. \quad (3.66)$$

That is, charge sharing yields

$$V_A(0^+) = \frac{C_b}{C_b + C_p} V_b, \quad (3.67)$$

revealing that V_A jumps instantaneously from zero to $[C_b/(C_b + C_p)]V_b$.

Before bringing V_{in} into the picture, we make two observations. First, the voltage on the two capacitors changes abruptly here because they can, ideally, deliver an infinite current to each other. Second, the initial condition on C_b can be viewed as a voltage step in series with C_b [Fig. 3.36(c)]. These points prove useful in the analysis of switched-capacitor circuits, which we encounter frequently in this book.

With an initial condition of $[C_b/(C_b + C_p)]V_b$ created on C_p , we study what happens as V_{in} rises. The circuit now acts as that in the previous example, attenuating the input swings. The total voltage at A is thus given by

$$V_A = \frac{C_b}{C_b + C_p} V_b + \frac{C_b}{C_b + C_p} V_{in}. \quad (3.68)$$

We can alternatively derive this result through the use of superposition. First, we consider the initial condition on C_b as an “input” and find its effect at A while assuming $V_{in} = 0$. This leads to (3.67). Next, we set the initial condition to zero and apply V_{in} , arriving at (3.65). Finally, we sum these results.

The circuit of Fig. 3.33(b) looks promising as it ensures a constant V_{GS} for M_1 while V_{in} varies with time. Let us neglect body effect and note that (1) the switch on-resistance, $1/[\mu_n C_{ox}(W/L)(V_{GS} - V_{TH})]$, is constant, avoiding the input-dependent phase shift studied in Section 3.3, and (2) the switch channel charge, $WLC_{ox}(V_{GS} - V_{TH})$, is also constant, creating only an offset at the output after M_1 turns off. We expect that clock feedthrough also becomes independent of V_{in} .

3.8.2 Complete Bootstrapped Circuit

The use of a precharged capacitor in Fig. 3.33(b) as a battery leads to a constant V_{GS} for the sampling switch while it is on. We now add more devices to the circuit for two purposes: (1) we must be able to turn the switch off, and (2) we must recharge C_b while the switch is off. The latter is necessary because C_b loses some charge in each cycle due to leakage currents and charge sharing, as explained below. We must therefore detach C_b from M_1 and tie it to V_{DD} and ground, and also connect the gate of M_1 to zero. From these thoughts emerges the arrangement shown in Fig. 3.37. When the circuit enters the hold mode, M_2 and M_3 turn off to disengage

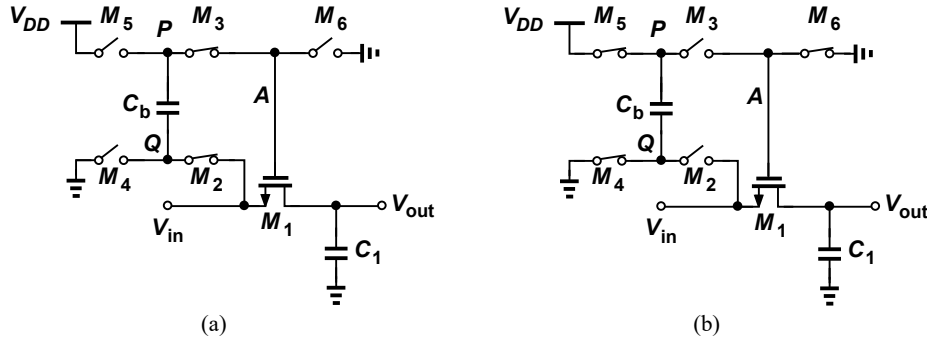


Figure 3.37 Bootstrapped circuit with necessary switches in (a) track mode, and (b) hold mode.

C_b from M_1 , and M_4 and M_5 turn on to charge C_b to V_{DD} . At the same time, M_6 turns on, pulling the gate of M_1 to zero.

The bootstrapped topology of Fig. 3.37 requires at least six switches and two capacitors, but it improves the performance so much that it has become a common choice in most designs. As we delve into this circuit, we must answer two questions. (1) Among M_2 - M_6 , which devices should be NFETs and which ones should be PFETs? (2) How do we choose the value of C_b and the dimensions of M_2 - M_6 ? Let us first examine the waveforms at different nodes of the circuit.

Example 3.19

If V_{in} in Fig. 3.37 is a rail-to-rail sinusoid, sketch the waveforms at A , P , and Q for some clock cycles.

Solution

In the acquisition mode, C_b is tied to the gate and source of M_1 , $V_A = V_P$, $V_Q = V_{in}$, and A and P track V_{in} but with a level shift of V_{DD} . As shown in Fig. 3.38, this occurs between t_1 and t_2 , between t_3 and t_4 , and between t_5 and t_6 . In the hold mode, e.g., between t_2 and t_3 , $V_A = 0$, $V_P = V_{DD}$, and $V_Q = 0$.

The waveforms plotted in Fig. 3.38 provide a first step toward choosing the types of switches in the circuit. We make the following observations: (1) V_A reaches as high as $2V_{DD}$ and as low as zero, (2) V_P is always greater than or equal to V_{DD} , and (3) V_Q is always less than or equal to V_{DD} .

We now decide between NMOS and PMOS devices for each switch in Fig. 3.37. Since M_4 and M_6 connect to ground, they must be realized as NFETs. Also, since M_5 ties the top plate of C_b to V_{DD} , it must be a PMOS transistor. Similarly, M_3 must be a PFET (why?). These thoughts lead us to the partial implementation depicted in Fig. 3.39(a). We still face two tasks: the choice of transistor type for M_2 , and the use of proper control signals for the gates of M_2 - M_6 . Switch M_2 , like M_1 , senses V_{in} , and must be formed by an NMOS device (why?). But such a transistor would turn off as V_{in} reaches a high level, e.g., near V_{DD} . This issue is resolved by allowing the gate of M_2 to be bootstrapped as well. In fact, since M_1 and M_2 are simultaneously on or off, we can simply tie their gates together [Fig. 3.39(b)]. We recognize that, when M_3 turns on, it connects the high voltage on P to A , thereby turning both M_1 and M_2 on.

Example 3.20

Prove that M_2 in Fig. 3.39(b) turns on after M_3 does.

Solution

As the circuit enters the acquisition mode, e.g., at $t = t_3$ in Fig. 3.38, A is released from ground, but it remains at zero until M_3 turns on. Afterwards, V_A reaches V_P and M_2 turns on.

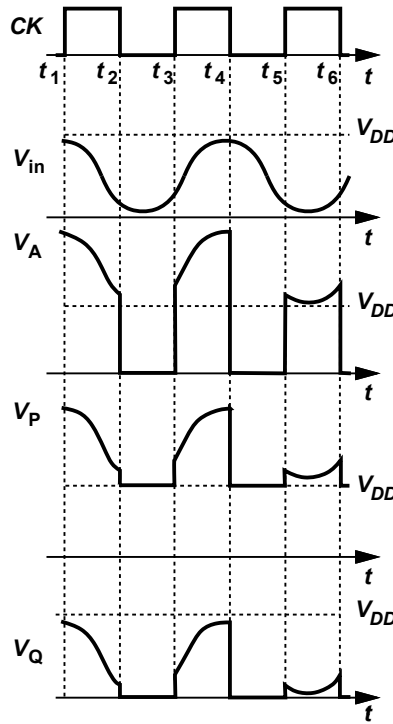


Figure 3.38 Bootstrapped circuit waveforms.

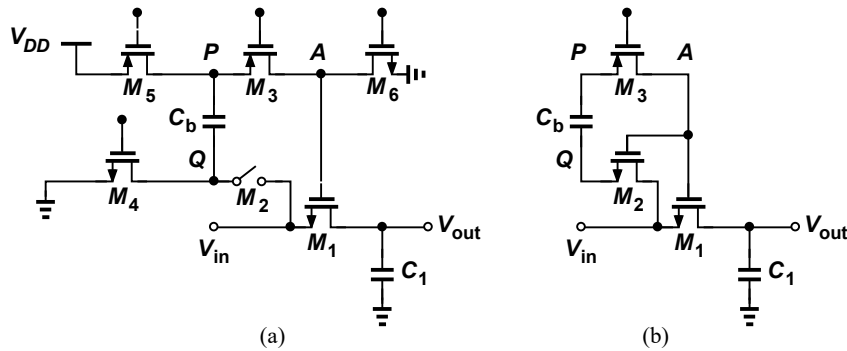


Figure 3.39 (a) Realization of some switches by MOSFETs, and (b) implementation of M_2 .

Choice of Clock Phases In order to implement the acquisition and hold modes in Fig. 3.39(a), we surmise that M_3 , M_4 , and M_6 must be controlled by $\overline{CK}$. But how about M_5 ? If driven by CK , M_5 is expected to turn off when CK is high and the circuit operates in the acquisition mode. However, for large input swings, M_5 does turn on. Illustrated in the simplified circuit of Fig. 3.40, this effect occurs because V_P is equal to $V_{in} + V_{DD}$, rising above V_{DD} . Consequently, the source voltage of M_5 exceeds its gate voltage, turning the device on if $V_P > V_{DD} + |V_{TH5}|$, i.e., if $V_{in} > |V_{TH5}|$.

From this observation, we conclude that, in the acquisition mode, the gate voltage of M_5 must rise in unison with its source voltage to ensure that its V_{GS5} does not reach V_{TH5} . This can be accomplished by bootstrapping this gate to V_{in} . The overall topology emerges as shown in Fig. 3.41(a).

The last modification necessary here relates to the maximum voltage at node A . For an input swing from 0 to V_{DD} , V_A reaches $2V_{DD}$ while M_6 is off. That is, the drain-source and drain-gate voltages of M_6 can

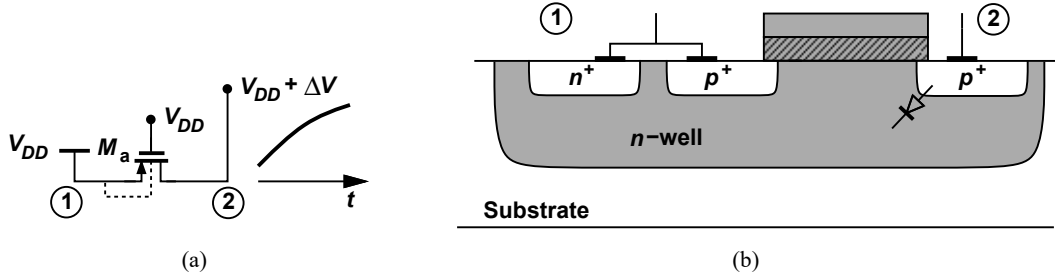


Figure 3.42 (a) A PMOS device sensing a high voltage at its drain, and (b) conduction of the PMOS drain-bulk junction diode.

3.8.3 Bootstrapping Imperfections

The complete bootstrapped sampling circuit of Fig. 3.41(b) provides high linearity at high speeds, but its performance is ultimately limited by certain imperfections. In this section, we investigate such effects.

Charge Sharing Recall that C_b in Fig. 3.41(b) is first charged to V_{DD} and then connected to the gate and source of the sampling switch, M_1 . We expect that C_b loses some charge to the capacitances present at the gate of M_1 , delivering a voltage less than V_{DD} . We identify these capacitances as follows. First, the gate capacitances of M_1 and M_2 appear in parallel with C_b , thereby sharing its charge (Fig. 3.43). Operating in

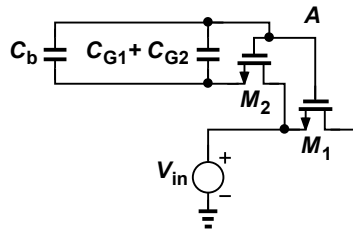


Figure 3.43 Charge sharing between C_b and other capacitances.

the deep-triode region, these two transistors present $C_{GS} = C_{GD} = WLC_{ox}/2 + WC_{ov}$ at their gates. That is, the total capacitance introduced by M_1 and M_2 in parallel with C_b is given by

$$C_{G1} + C_{G2} = (WL)_1 C_{ox} + 2W_1 C_{ov} + (WL)_2 C_{ox} + 2W_2 C_{ov}. \quad (3.69)$$

Example 3.21

Why do we merge the gate-source and gate-drain capacitances of M_1 and M_2 even though they are not exactly in parallel?

Solution

We can view the circuit as shown in Fig. 3.44, where R_{on1} and R_{on2} denote the on-resistances of M_1 and M_2 , respectively. We have tacitly assumed that R_{on1} and R_{on2} are sufficiently small to allow merging the four capacitances. In practice, this is a reasonable approximation.

The second group of capacitances appears from the top plate of C_b to ground. Illustrated in Fig. 3.45, these include the drain-bulk capacitance of M_5 , C_{DB5} , all of the capacitances of M_3 , $C_{GS3} + C_{GD3} + C_{DB3} + C_{SB3}$, and the drain capacitance of M_7 , $C_{GD7} + C_{DB7}$. We denote the sum by C_p .

To compute the voltage delivered by C_b , we draw the equivalent circuit shown in Fig. 3.46, apply superposition by setting V_{in} to zero, and write

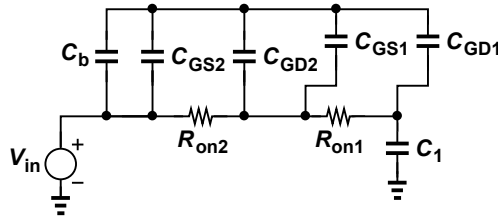


Figure 3.44 Illustration of transistor resistances and capacitances.

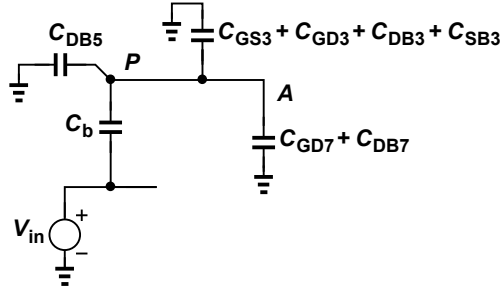


Figure 3.45 Simplified model for studying charge sharing.

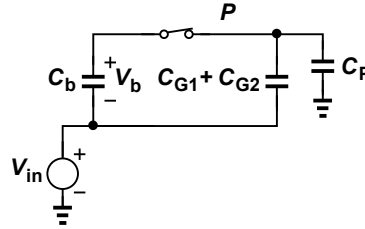


Figure 3.46 Equivalent circuit during sampling mode.

$$V'_b = \frac{C_b}{C_b + C_{G1} + C_{G2} + C_p} V_{DD}. \quad (3.70)$$

In a typical design, we wish to maintain $V'_b \geq 0.9V_{DD}$ by choosing C_b sufficiently large.

V_{GS} Variation The gate-source voltage of the sampling switch in Fig. 3.41(b) varies in the sampling mode due to two effects. First, the capacitance seen from node P to ground forms an attenuator and causes $V_P - V_{in}$ to change. This phenomenon is quantified by noting from Fig. 3.46 that V_{in} experiences an attenuation factor of $(C_b + C_{G1} + C_{G2}) / (C_b + C_{G1} + C_{G2} + C_p)$ as it reaches node P . We then apply superposition and write

$$V_P = V'_b + \frac{C_b + C_{G1} + C_{G2}}{C_b + C_{G1} + C_{G2} + C_p} V_{in}, \quad (3.71)$$

where V'_b is the initial voltage across C_b when $V_{in} = 0$. It follows that

$$V_P - V_{in} = V'_b - \frac{C_p}{C_b + C_{G1} + C_{G2} + C_p} V_{in}. \quad (3.72)$$

To minimize this effect, we must have $C_b + C_{G1} + C_{G2} + C_p \gg C_p$.

Example 3.22

A student decides to use a very large value for C_b to minimize the foregoing issues. Explain the trade-offs in the choice of C_b .

Solution

In addition to occupying a large area on the chip, a large capacitor also presents significant *parasitic* capacitances (Chapter 2). Thus, V_{in} must drive these capacitances as well.

The second phenomenon leading to V_{GS} variation stems from the limited bandwidth in the bootstrapping path [Fig. 3.47(a)]. If V_{in} changes at a high rate, the resistances in series with C_b do not allow V_P to track

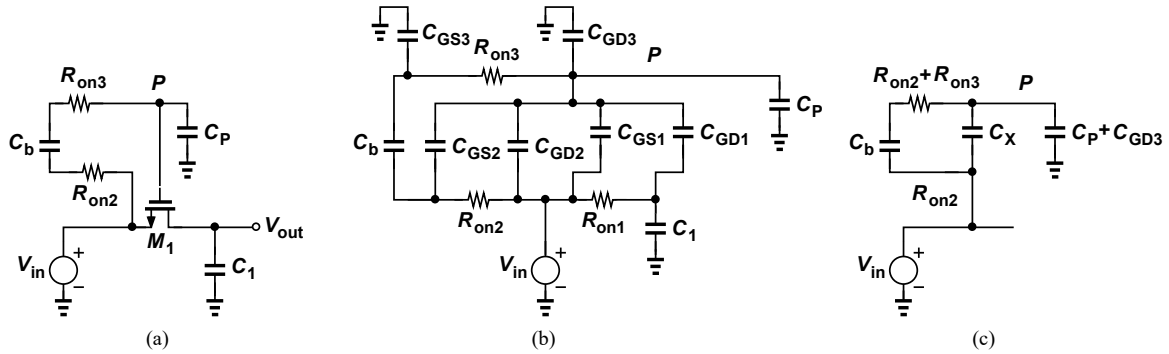


Figure 3.47 (a) Circuit for calculating bootstrapped path BW, (b) detailed equivalent circuit, and (c) approximate equivalent.

V_{in} accurately, i.e., V_{GS1} is not constant. To formulate this effect, we must compute the transfer function from V_{in} to $V_{GS1} = V_P - V_{in}$. We model the circuit as shown in Fig. 3.47(b), where $C_{GS,n} = C_{GD,n}$ for $n = 1, 2, 3$, because all three transistors operate in the deep-triode region. The model appears rather complex, but we can simplify it by assuming that (1) $V_{out} \approx V_{in}$ near the end of the acquisition mode, allowing us to merge C_{GS1} and C_{GD1} , and remove C_1 ;³ and (2) C_{GS2} , C_{GD2} , and C_{GS3} are negligible, a condition valid in typical designs. The remaining capacitances are readily merged, yielding the model depicted in Fig. 3.47(c). Here,

$$C_X \approx C_{GS1} + C_{GD1}, \quad (3.73)$$

and R_{on2} and R_{on3} appear in series because we have neglected C_{GS2} and C_{GS3} .

Example 3.23

Explain intuitively how V_P in Fig. 3.47(c) tracks V_{in} at very low and at very high frequencies.

Solution

At very low frequencies, the impedance of C_b is much greater than $R_{on2} + R_{on3}$, leading to the equivalent circuit shown in Fig. 3.48(a). We thus have

$$\frac{V_P}{V_{in}} = \frac{C_b + C_X}{C_b + C_X + C_P + C_{GD3}}. \quad (3.74)$$

³This point can also be seen by assuming that $R_{on1} \approx 0$ and noting that C_1 has little effect on the circuit because it is driven by V_{in} .

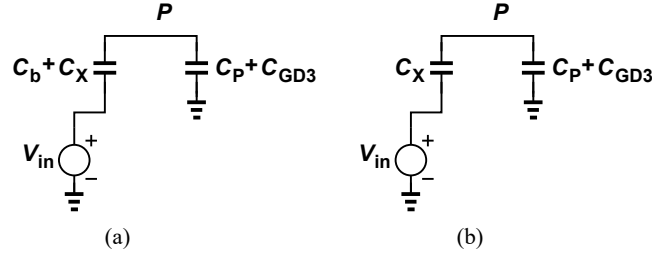


Figure 3.48 (a) Low-frequency, and (b) high-frequency models of bootstrapped path.

As expected, V_P tracks V_{in} well if $C_b \gg C_X + C_p + C_{GD3}$. At very high frequencies, on the other hand, C_X overwhelms the branch consisting of $R_{on2} + R_{on3}$ and C_b , yielding the situation depicted in Fig. 3.48(b), where

$$\frac{V_P}{V_{in}} = \frac{C_X}{C_X + C_p + C_{GD3}}. \quad (3.75)$$

Since C_X and $C_p + C_{GD3}$ are comparable, V_P does not track V_{in} accurately in this case.

Let us now return to Fig. 3.47(c) and compute the transfer function from V_{in} to V_P and then to $V_P - V_{in}$. Expressing the current through $C_p + C_{GD3}$ as $V_P(C_p + C_{GD3})s$ and that through C_X as $(V_{in} - V_P)C_X s$, we write a KCL at P to obtain the current through $R_{on2} + R_{on3}$ and C_b as $V_P(C_p + C_{GD3})s - (V_{in} - V_P)C_X s$. Next, we multiply this current by $R_{on2} + R_{on3} + 1/(C_b s)$ and equate the result to $V_{in} - V_P$:

$$[V_P(C_p + C_{GD3})s - (V_{in} - V_P)C_X s] \left(R_{on2} + R_{on3} + \frac{1}{C_b s} \right) = V_{in} - V_P. \quad (3.76)$$

It follows that

$$\frac{V_P}{V_{in}} = \frac{(R_{on2} + R_{on3})C_X C_b s + C_b + C_X}{(R_{on2} + R_{on3})(C_X + C_p + C_{GD3})C_b s + C_p + C_{GD3} + C_X + C_b}. \quad (3.77)$$

As $s \rightarrow 0$ or $s \rightarrow \infty$, we observe the same results as those predicted in Example 3.23.

The gate-source voltage of the sampling switch experiences dynamics given by

$$\frac{V_P - V_{in}}{V_{in}} = - \frac{(R_{on2} + R_{on3})(C_p + C_{GD3})C_b s + C_p + C_{GD3}}{(R_{on2} + R_{on3})(C_X + C_p + C_{GD3})C_b s + C_p + C_{GD3} + C_X + C_b}. \quad (3.78)$$

The transfer function exhibits a zero at

$$\omega_z = \frac{1}{C_b(R_{on2} + R_{on3})}. \quad (3.79)$$

If $C_{GD3} \ll C_X + C_p$ and $C_b \gg C_p + C_{GD3} + C_X$, we obtain a pole at

$$\omega_p \approx \frac{1}{(C_X + C_p)(R_{on2} + R_{on3})}. \quad (3.80)$$

Figure 3.49 plots the magnitude of the transfer function, revealing that, for $\omega < \omega_z$, it assumes a *small* value (why?). That is, $V_P \approx V_{in}$. For $\omega > \omega_p$, on the other hand, $|(V_P - V_{in})/V_{in}|$ can reach about 0.5 if C_X and $C_p + C_{GD3}$ are comparable. This means that changes in V_{in} do not propagate accurately to V_P .

The foregoing study implies that $R_{on2} + R_{on3}$ must be minimized so as to ensure the sampling switch remains linear at high input frequencies. Of course, as we increase W_2 and W_3 while C_b is kept constant, the capacitances of M_2 and M_3 eventually limit the performance.

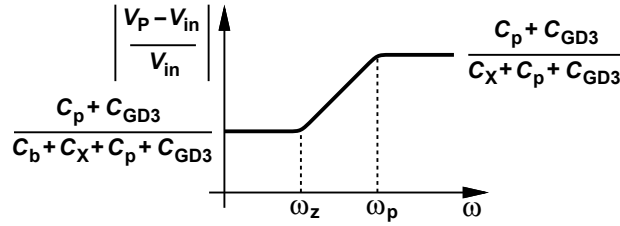


Figure 3.49 Magnitude response of the bootstrapped path.

3.9 Performance Parameters

Our study of sample-and-hold circuits has revealed that their performance is quantified in terms of speed, linearity, etc. In this section, we formally define these parameters.

3.9.1 Speed

The speed is primarily represented by the acquisition time, t_{acq} , which is defined as the amount of time the output takes to reach “sufficiently” close to the input. As illustrated in Fig. 3.50, when CK goes high and M_1 turns on, it takes t_{acq} seconds for $|V_{in} - V_{out}|$ to fall below an acceptable error, ΔV . The value of this error depends on the overall ADC resolution and is typically chosen less than 0.5 LSB.

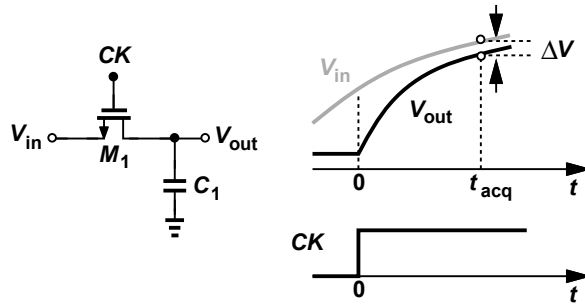


Figure 3.50 Definition of acquisition time.

Example 3.24

In the circuit of Fig. 3.51, $V_{out} = 0$ before $t = 0$. (Bootstrapping is not shown.) If V_{in} is constant and equal to the ADC full scale, V_{FS} , compute the acquisition time. Assume a resolution of N bits and model M_1 by a constant resistance, R_{on1} .

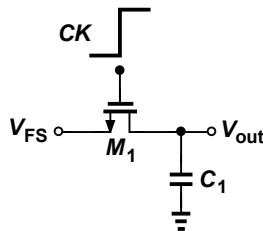


Figure 3.51 Simple circuit for calculating acquisition time.

Solution

The LSB size is given by $V_{FS}/2^N$. We wish V_{out} to reach $V_{FS} - 0.5$ LSB:

$$V_{FS} \left(1 - \exp \frac{-t_{acq}}{R_{on1}C_1} \right) = V_{FS} - \frac{V_{FS}}{2^{N+1}}. \quad (3.81)$$

It follows that

$$t_{acq} = R_{on1}C_1 \ln 2^{N+1} \quad (3.82)$$

$$= (N + 1)R_{on1}C_1 \ln 2 \quad (3.83)$$

$$\approx 0.69(N + 1)R_{on1}C_1. \quad (3.84)$$

For example, an 8-bit system requires an acquisition time equal to 6.2 time constants. We observe that $t_{acq} \propto N$ for $N \gg 1$.

3.9.2 Linearity

The linearity of sampling circuits is quantified by the output harmonic content. We apply a full-scale sinusoid to the input and allow the circuit to sample and hold the signal over many cycles [Fig. 3.52(a)]. We then measure the harmonics in the output waveform. A key point here is that the quantizer that would follow the

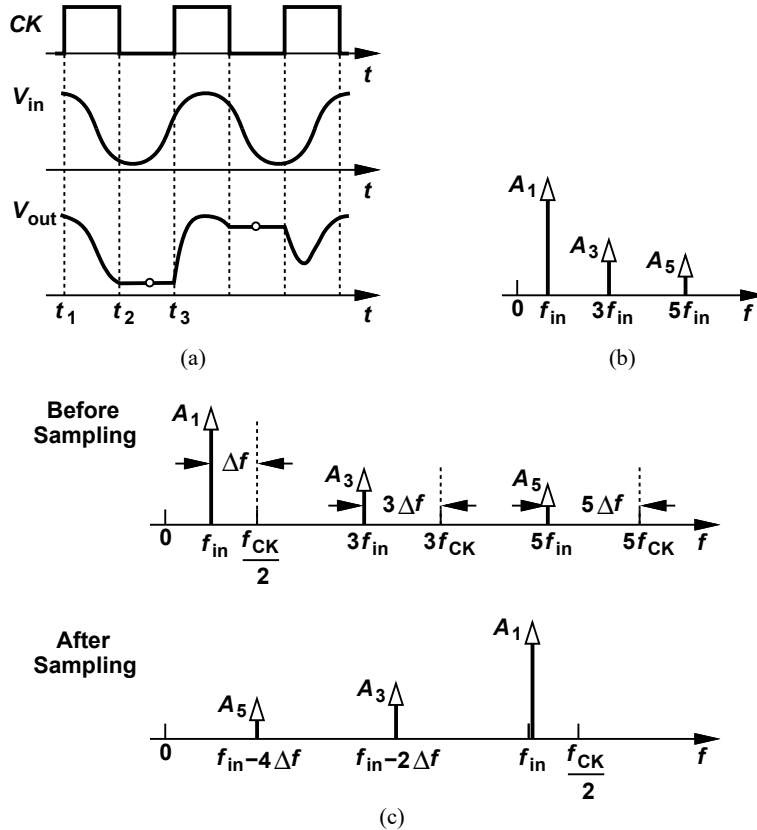


Figure 3.52 (a) Sampling circuit waveforms, (b) resulting spectrum, and (c) spectrum near Nyquist rate.

sampler digitizes only the *held* values in V_{out} , e.g., between t_2 and t_3 , and ignores the tracking parts of V_{out} , e.g., between t_1 and t_2 . For this reason, the output harmonic computation must consider only the sampled values, as indicated by the dots in Fig. 3.52(a). The analysis proceeds as follows. We perform a transient simulation and take the fast Fourier transform (FFT) of the sampled values, obtaining the corresponding spectrum, Fig. 3.52(b), where f_{in} denotes the input sinusoid frequency. In a fully-differential implementation without mismatches, the output contains only odd harmonics. We then find the total harmonic distortion as

$$HD_{tot} = \frac{A_3^2 + A_5^2 + \dots}{A_1^2}. \quad (3.85)$$

We have seen in the previous sections that the distortion tends to rise at higher input frequencies. The worst case therefore occurs at the Nyquist rate, where f_{in} is slightly below $f_{CK}/2$, i.e., $f_{in} = f_{CK}/2 - \Delta f$. This sampling scenario yields a more complex output spectrum, which we can derive by making the following observations. (1) In a differential circuit, an input sinusoid at $f_{in} = f_{CK}/2 - \Delta f$ produces harmonics at $3f_{in} = 3f_{CK}/2 - 3\Delta f$, $5f_{in} = 5f_{CK}/2 - 5\Delta f$, etc. [Fig. 3.52(c)]. (2) All of these components are sampled at a rate of f_{CK} , shifting up and down by f_{CK} , $2f_{CK}$, etc. Since $3f_{in}$, $5f_{in}$, etc., are *greater* than $f_{CK}/2$, they experience aliasing and are “folded” back to the first Nyquist zone. These aliased components are obtained as follows. The third harmonic, $3f_{CK}/2 - 3\Delta f$, is shifted down by f_{CK} , landing at $f_{CK}/2 - 3\Delta f = f_{in} - 2\Delta f$. Similarly, the fifth harmonic, $5f_{CK}/2 - 5\Delta f$, slides down by $2f_{CK}$, translating to $f_{CK}/2 - 5\Delta f = f_{in} - 4\Delta f$. We again compute the distortion according to Eq. (3.85).

3.9.3 Static and Dynamic Nonlinearity

The harmonic distortion test prescribed in the previous section generally reveals greater nonlinearity at higher input frequencies. This trend encourages us to distinguish between “static” nonlinearity and “dynamic” nonlinearity.

Sampling circuits suffer from input-dependent charge injection and clock feedthrough, exhibiting distortion that is relatively independent of the input frequency. We call this mechanism “static.” At high input frequencies, on the other hand, the variation of the sampling switch on-resistance leads to input-dependent phase shift and additional distortion. Similarly, the finite on-resistance of the switches in the bootstrap path causes V_{GS} variation at high frequencies. We call these mechanisms “dynamic.”

In the design of samplers, we first examine the static linearity by measuring the output harmonics at low input frequencies and adjust the value of the sampling capacitor to suppress them. This still requires transient simulations followed by an FFT. We then turn to dynamic linearity, run the circuit at high frequencies, measure the distortion, and see whether further adjustments are necessary. For example, we can make the sampling switch wider so as to lower its resistance. Of course, such an adjustment may degrade the static linearity (why?). Thus, some iteration between static and dynamic tests is necessary.

3.10 Effect of Clock Jitter

The clock that drives a sampling circuit is not perfectly periodic. Owing to the noise sources within the clock generator, the period varies randomly [Fig. 3.53(a)]. From another perspective, the clock transitions are not uniformly spaced in time. We say the clock suffers from “jitter.”

Recall from Section 3.1.2 that the cascade of the track-and-hold circuit and the quantizer acts as an impulse sampler. Random displacements of the clock edges thus translate to “nonuniform” sampling of the analog input. Illustrated in Fig. 3.53(b), this effect introduces errors in the sampled voltages. We observe that the error is large for high input slew rates. More specifically, since $dV_{in}/dt \approx \Delta V_{in}/\Delta t$, a time displacement of ΔT [Fig. 3.53(c)] yields a voltage error given by

$$\Delta V \approx \Delta T \frac{dV_{in}}{dt}. \quad (3.86)$$

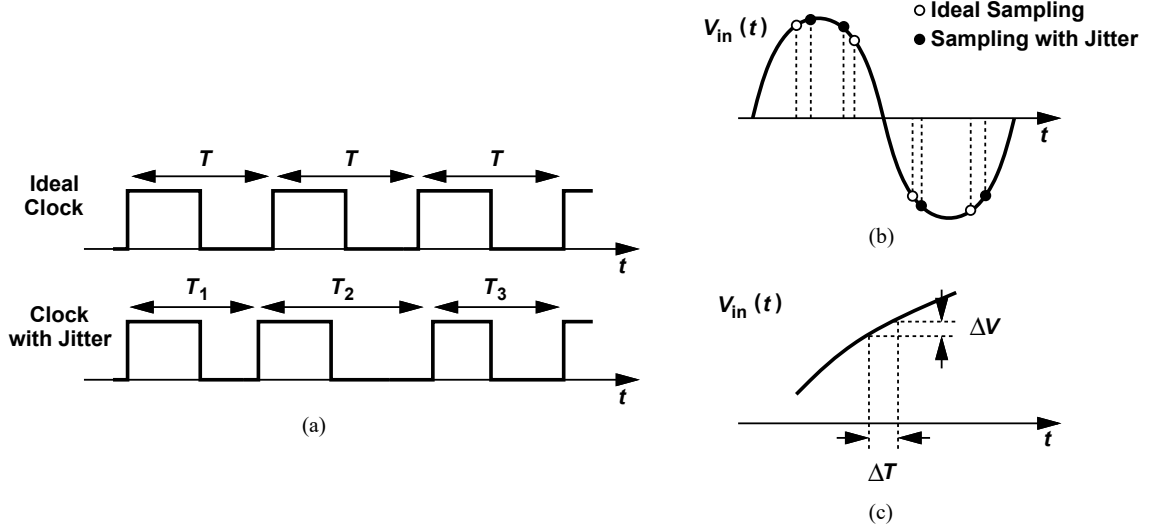


Figure 3.53 (a) Illustration of clock jitter, (b) effect of jitter on sampled signals, and (c) resulting voltage error.

Writing $V_{in}(t) = V_1 \sin \omega_1 t$, we have

$$\Delta V \approx \Delta T (V_1 \omega_1) \cos \omega_1 t. \quad (3.87)$$

Since ΔT is a random quantity, so is ΔV , thereby introducing *noise* in the sampled signal. We hereafter denote ΔV by a noise voltage, V_n .

Let us compute the output rms noise caused by the clock jitter. We first obtain the average power of the noise voltage, V_n , by recognizing that ΔT and $(V_1 \omega_1) \cos \omega_1 t$ in (3.87) are uncorrelated. We can then write the average power as the product of the average powers of ΔT and $(V_1 \omega_1) \cos \omega_1 t$. For the jitter component, the average power, P_j , is known from the clock generator design. The jitter power (also called the jitter “variance”) is equal to the square of the rms jitter, σ_j . The average power of $(V_1 \omega_1) \cos \omega_1 t$ is simply equal to $(V_1 \omega_1)^2/2$ (why?). It follows that

$$P_{V_n} = P_j \frac{(V_1 \omega_1)^2}{2}, \quad (3.88)$$

where P_{V_n} denotes the average power of V_n . We can also find the signal-to-noise ratio if we divide the signal power, $P_{sig} = V_1^2/2$, by P_{V_n} :

$$\text{SNR} = \frac{P_{sig}}{P_{V_n}} \quad (3.89)$$

$$= \frac{1}{\sigma_j^2 \omega_1^2}. \quad (3.90)$$

Note that the SNR does not depend directly on the clock frequency or the signal amplitude.

Example 3.25

A sampling circuit must achieve an SNR of 60 dB at an input frequency of 5 GHz. If the clock jitter is the only source of noise, how much jitter is tolerable?

Solution

Since $60 \text{ dB} \equiv 10^6$ for ratio of power quantities, we must have

$$\sigma_j^2 = \frac{1}{\text{SNR} \cdot \omega_1^2} \quad (3.91)$$

$$= 1.01 \times 10^{-27} \text{ s}^2, \quad (3.92)$$

which translates to

$$\sigma_j = 32 \text{ fs}. \quad (3.93)$$

This very small value underscores the severity of clock jitter in high-resolution, high-speed ADC design.

Example 3.26

The circuit in the previous example operates with a $0.5 V_{pp}$ analog input and incorporates a 1-pF sampling capacitor. Compute the overall SNR for the clock jitter found above. Assume a single-ended circuit.

Solution

We must first compute the total noise power, including both P_{V_n} from (3.88) and kT/C noise. With $P_j = \sigma_j^2 = 1.01 \times 10^{-27} \text{ s}^2$ from the previous example and $V_1 = 0.25 \text{ V}$, we have $P_{V_n} = 3.12 \times 10^{-8} \text{ V}^2$. Also, $kT/C = 4.14 \times 10^{-9} \text{ V}^2$. It follows that

$$\text{SNR} = \frac{V_1^2/2}{P_{V_n} + kT/C} \quad (3.94)$$

$$= 8.84 \times 10^5 \quad (3.95)$$

$$\equiv 59.5 \text{ dB}. \quad (3.96)$$

How much clock jitter is acceptable? We postulate that the jitter-induced output noise voltage, V_n , should have an rms value well below the LSB of the overall ADC. We formulate this condition after deriving the quantization noise power in Chapter 12.

References

- [1]. M. Dessouky and A. Kaiser, "Input switch configuration suitable for rail-to-rail operation of switched op amp circuits," *Electronics Letters*, vol. 35, pp. 8-10, Jan. 1999.
- [2]. B. Razavi, *Design of Analog CMOS Integrated Circuits*, Second Ed., McGraw-Hill, New York, 2017.

Problems

- 3.1. Consider the circuit in Fig. 3.5(a) and, for simplicity, assume that $t_1 = 0$ while $V_{in} = V_0 \cos \omega_1 t$. Obtain an expression for $V_{out}(t)$. If given enough time, does V_{out} come arbitrarily close to V_{in} ? Explain.
- 3.2. Solve Eq. (3.22) by factoring V_{out} on the left-hand side and using partial fractions. Note that $V_{out}(0) = V_{DD}$.
- 3.3. Suppose we double the switch width in Fig. 3.13.
 - (a) What happens to the R_{on} plot in Fig. 3.14?
 - (b) What happens to HD_2 and HD_3 ?
- 3.4. Construct the plots in Fig. 3.18 if $\mu_p(W/L)_P < \mu_n(W/L)_N$.
- 3.5. Referring to Fig. 3.22, let us assume $V_{in} = V_{DD}/2 + (V_{DD}/2) \cos \omega_1 t$ and express the time-varying capacitance $C_1 + C_{DB1} + C_{DB2}$ as $C_a + C_b \cos \omega_1 t$. The output voltage can be approximated by

$$V_{out} \approx \frac{V_{DD}}{2} + \frac{V_{DD}}{2} \cos[\omega_1 t - R_{on}(C_a + C_b \cos \omega_1 t)\omega_1], \quad (3.97)$$

where amplitude attenuation is neglected. Expand this result and compute HD_2 and HD_3 .

- 3.6. Let us return to Fig. 3.26 and write $\Delta V = Q_{ch}/C_1$ and $\tau = R_{on}C_1$. Express both Q_{ch} and R_{on} in terms of $\mu_n C_{ox} W/L$ and $V_{GS} - V_{TH}$ and prove that

$$\Delta V = \frac{L^2}{\mu_n \tau}. \quad (3.98)$$

This relation exemplifies a precision-speed trade-off.

- 3.7. In the circuit of Fig. 3.26, we double the transistor width and the value of C_1 . What happens to ΔV and kT/C noise?
- 3.8. In Fig. 3.28(a), the voltage across C_1 must have a slew rate of at least S_0 . How much is the minimum acceptable value of I_1 ? If we allow I_{D1} to drop to zero while this voltage is falling, does the signal experience distortion?
- 3.9. In the differential sampling circuit of Fig. 3.29, we double the widths of M_1 and M_2 . Assume $V_{in1} = V_{in2} = V_{DD}/2$.
 - (a) What happens to the clock feedthrough error in $V_{out1} - V_{out2}$?
 - (b) What happens to the channel charge injection error in $V_{out1} - V_{out2}$?
 - (c) What happens to the kT/C noise in $V_{out1} - V_{out2}$?
- 3.10. In the bootstrapped circuit of Fig. 3.35, we choose $C_b = 5C_p$. What fraction of V_{in} appears at node A ?
- 3.11. A student accidentally omits switch M_4 in Fig. 3.37. Explain what happens to the circuit's operation. Assume C_b begins with a zero initial condition.
- 3.12. Repeat the previous problem if the student omits only M_5 .
- 3.13. Explain how the circuit of Fig. 3.37 behaves if (a) M_6 is omitted, (b) M_3 is omitted, or (c) M_2 is omitted.
- 3.14. Consider the simplified circuit shown in Fig. 3.39(b). If $V_{in} = V_{DD}/2 + (V_{DD}/2) \cos \omega_1 t$, sketch the waveforms at Q , P , and A . Assume C_b holds a voltage equal to V_{DD} . What is the maximum voltage at A ?
- 3.15. In the circuit of Fig. 3.37, it is possible that the switches do not turn on or off simultaneously. Suppose, for example, that, at the end of the sampling mode, M_2 can turn off slightly before or after M_4 turns on. Which scenario is preferred?
- 3.16. We select the width of M_2 in Fig. 3.43 to be one-fifth of that of M_1 . If C_b is initially charged to V_{DD} and then connected to M_1 and M_2 , express its voltage in terms of W , L , C_{ox} , and C_{ov} . Derive a condition if C_b must not lose more than 10% of its charge.
- 3.17. Explain qualitatively how the kT/C noise deposited by M_5 and M_4 affects the operation of the circuit in Fig. 3.37 after it enters the sampling mode. This point reveals that the value of C_b is dictated by charge sharing rather than by kT/C noise.
- 3.18. Suppose we wish to keep the gate of M_5 in Fig. 3.40 connected to V_{DD} . What is the maximum tolerable voltage at P ? How does this limit the voltage swing at the main input?
- 3.19. In the bootstrapped circuit of Fig. 3.33, the source voltage of M_1 varies with the input level. Sketch the threshold voltage of M_1 versus V_{in} and explain why the circuit still suffers from input-dependent channel charge injection.

- 3.20.** We wish to apply the bootstrapping method of Fig. 3.41(a) to the differential sampler of Fig. 3.29. Can any of the devices be shared between the two bootstrapping networks?

Design Study of a Sampling Circuit

The previous chapter has laid the foundation for the design of sampling circuits and, in particular, the benefits of bootstrapping. We have seen that the bootstrapping action reduces the distortion and improves the speed with minimal power penalty. In this chapter, we present the step-by-step design of this circuit [1].

4.1 General Considerations

We have developed the topology shown in Fig. 4.1. In the sampling mode, M_2 and M_3 are on, and in the hold mode, M_4 - M_6 are on, grounding the gate of M_1 and charging C_B to V_{DD} .

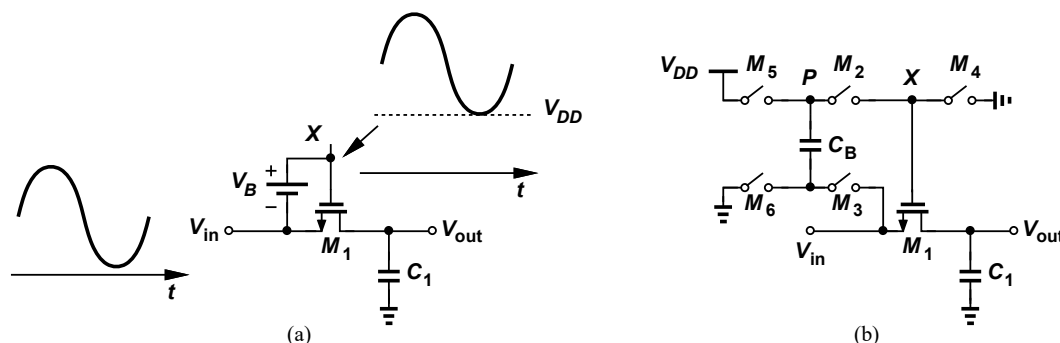


Figure 4.1 Bootstrapped sampling circuit.

We wish to design a differential sampler for the front end of a Nyquist-rate ADC with a resolution of 10 bits and a sampling rate of 5 GHz. Of the clock period of $T_{CK} = 200$ ps, we allocate one half to the sampling mode and the other to the hold mode. The design proceeds in a 28-nm CMOS process in the slow-slow (SS) corner at a temperature of 75°C with a worst-case supply of $1\text{ V} - 5\% = 0.95\text{ V}$. We assume a single-ended input range from 0.5 V to 0.75 V, which translates to a least-significant-bit (LSB) size of $1\text{ V}/1024 \approx 1\text{ mV}$ for the differential ADC. The bootstrapped sampler accommodates greater swings, but the stage preceding it—typically a buffer—may not.

For an N -bit ADC sensing a signal of the form $A \cos \omega_{in} t$, the ideal signal-to-noise ratio (SNR) is given by $(A^2/2)/(\Delta^2/12)$, where Δ is the LSB and equal to $2A/2^N$. It follows that $\text{SNR} = 6.02N + 1.76\text{ dB}$, which amounts to 62 dB for $N = 10$. In our differential system, $A = 0.5\text{ V}$.

The ADC specifications readily dictate certain values for the sampler. First, the harmonic distortion (HD) must remain well below -62 dB . Our ambition is to achieve an HD of about -65 dB in the worst case, namely, when $A = 0.5\text{ V}$ and $f_{in} = \omega_{in}/(2\pi)$ is near 2.5 GHz (i.e., at the Nyquist rate).

Second, the noise introduced by the sampler must negligibly degrade the overall SNR. Thus, the kT/C noise associated with M_1 and C_1 in Fig. 4.1(b) must be sufficiently small. We express the SNR of the sampler-ADC cascade as

$$\text{SNR} = \frac{\frac{A^2}{2}}{\frac{\Delta^2}{12} + \frac{2kT}{C_1}}, \quad (4.1)$$

where the factor of 2 in the denominator accounts for the noise in a differential sampling configuration. Let us target a 1-dB penalty due to kT/C noise, divide the SNR in Eq. (4.1) by the ideal SNR equation, take $10 \log$ of the result, and equate it to -1 dB. With $\Delta = 1$ mV and $T = 75^\circ\text{C} = 348$ K, we obtain $C_1 = 445$ fF, which we round up to 0.5 pF.

Third, the low-pass filter formed by M_1 and C_1 in Fig. 4.1(b) must negligibly attenuate the input signal in the track mode. The output amplitude is equal to $A/\sqrt{1 + R_{on1}^2 C_1^2 \omega_{in}^2}$, where R_{on1} denotes the on-resistance of M_1 . We aim for an attenuation of less than 0.5 dB at the Nyquist rate, arriving at $R_{on} = 44.5 \Omega$. We should select M_1 wide enough to ensure such a low resistance with $V_{GS1} \approx V_{DD} = 0.95$ V. This constraint yields $W_1 \approx 10 \mu\text{m}$ if $L_1 = 28$ nm.

4.2 Design Procedure

In order to understand the effect of device nonidealities, we follow an incremental design procedure. Specifically, we proceed in five steps: (1) we keep the battery in Fig. 4.1(a) and examine the distortion in only the track mode; (2) we allow the circuit to act as a sample-and-hold stage but, except for M_1 in Fig. 4.1(b), use ideal switches; (3) we change M_2 and M_4 to MOS devices; (4) we change the remaining switches to MOS-FETs; and (5) we replace the battery with a capacitor. In each case, we apply the full-scale input amplitude and measure the harmonic content of the differential output at a moderate input frequency and at the Nyquist rate. The incremental approach also permits us to optimize the design in every step.

We prefer to select minimum widths and lengths for the transistors unless there is a compelling reason not to do so. In this chapter, all channel lengths are equal to 28 nm.

4.3 Simulation Issues

In the design of sampling circuits, the clock and input frequencies must be chosen carefully. If f_{CK} is an integer multiple of f_{in} , then the circuit samples only certain points on the input waveform [Fig. 4.2(a)], failing to reveal its true performance. Even if we select $f_{CK}/f_{in} = P/Q$, where P and Q are integers,

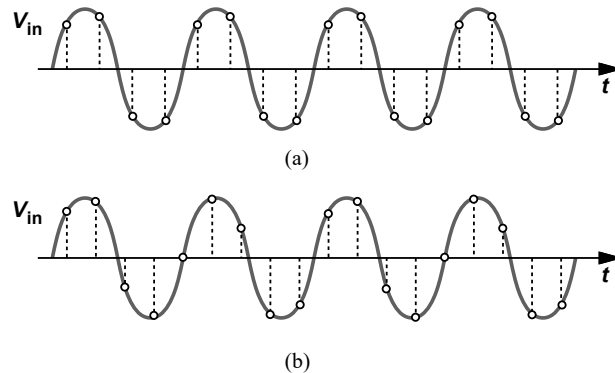


Figure 4.2 Sampling of the input with (a) an integer f_{CK}/f_{in} ratio, and (b) an irrational f_{CK}/f_{in} ratio.

we still have $QT_{in} = PT_{CK}$, where $T_{in} = 1/f_{in}$. This means that every Q cycles of the input exactly coincide with P cycles of the clock; thus, only Q values of the input are periodically sampled. For this reason, we should select f_{CK} and f_{in} such that the samples gradually slide along the input sinusoid and eventually assume all (or most) of the input values [Fig. 4.2(b)]. This is accomplished if f_{CK}/f_{in} is an irrational number. For simulations, we simply guarantee that P/Q yields a long periodicity. For example, with $f_{CK}/f_{in} = 5 \text{ GHz}/570 \text{ MHz} = 500/57$, the sampled points repeat themselves after every 500 clock cycles; i.e., 500 distinct points of the input voltage are collected.

Another issue relates to how we process the output time-domain waveform in Fig. 4.1(b) to go to the frequency domain. Since the ADC following the sampler senses only the held values on C_1 , the time points taken by fast Fourier transform (FFT) must be confined to the hold mode [Fig. 4.3(a)].

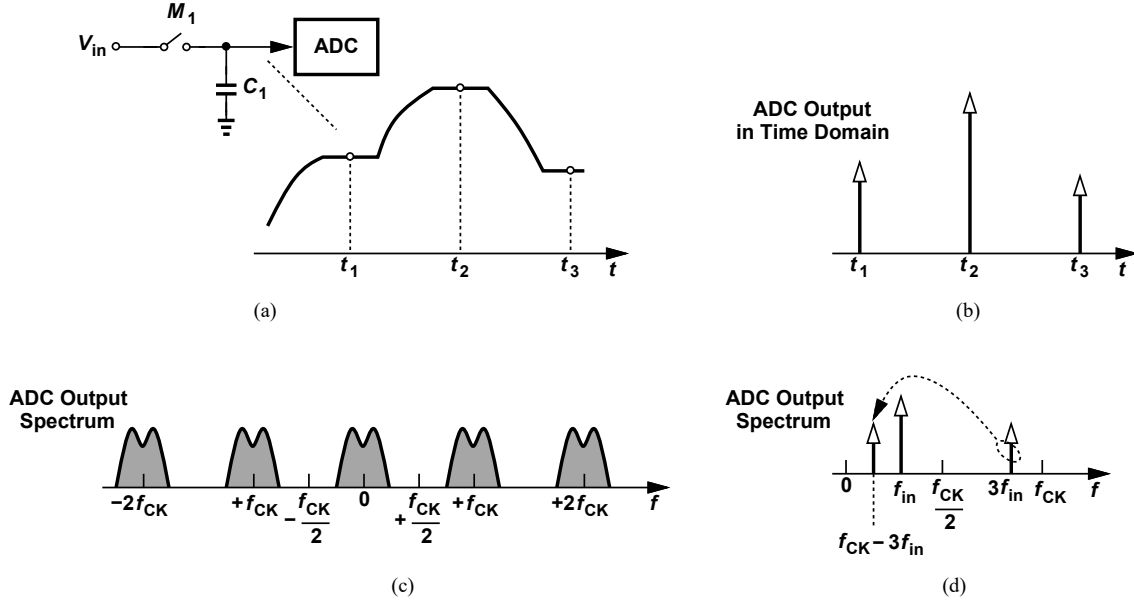


Figure 4.3 (a) Sensing of held values by the ADC, (b) equivalent ADC output in the time domain, (c) the corresponding spectrum, and (d) aliasing of the third input harmonic.

That the ADC only digitizes the held values also suggests that the sampler-ADC cascade equivalently multiplies the analog input by a train of impulses in the time domain [Fig. 4.3(b)] and hence convolves the input spectrum with a train of impulses in the frequency domain [Fig. 4.3(c)]. We call the range between $-f_{CK}/2$ and $+f_{CK}/2$ the “first Nyquist zone,” and generally confine our inspection of unwanted components to this range.

If the FFT senses only the values at t_1 , t_2 , etc., the resulting spectrum is given by the convolution of the input spectrum and a train of impulses. This point proves useful in understanding the FFT results. For example, if the sampler introduces third-order distortion, at $3f_{in}$, and this component lies above $f_{CK}/2$, then it is aliased to the first Nyquist zone. Depicted in Fig. 4.3(d), this phenomenon causes the third harmonic to land at $f_{CK} - 3f_{in}$.

4.4 Track-Mode Distortion

We begin by simulating two instances of Fig. 4.1(a) with differential inputs and $f_{in} = 570 \text{ MHz}$. As mentioned above, $W_1 = 10 \mu\text{m}$ for now. We surmise that this test yields a lower bound for the distortion because higher input frequencies and the sample-and-hold action tend to introduce greater nonlinearity. Figure 4.4 plots the FFT of the differential output. As expected, even harmonics are absent. The third and fifth harmonics are 79 dB and 97 dB below the fundamental, respectively. We denote the relative level of the former by HD_3 and that of the latter by HD_5 .

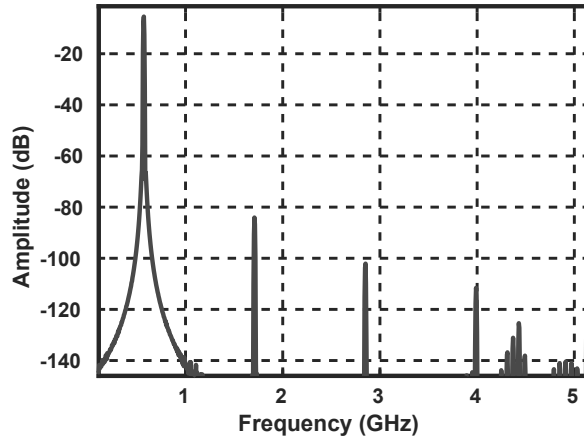


Figure 4.4 The track-mode output spectrum for $f_{in} = 570$ MHz.

Next, we increase f_{in} to 2.47 GHz and repeat the simulation. Figure 4.5(a) shows the time-domain outputs, revealing a peak-to-peak single-ended swing of 483 mV, which satisfies our attenuation constraint of 0.5 dB. Plotted in Fig. 4.5(b) is the output spectrum, exhibiting a third harmonic at -56 dB. This high level of distortion occurs because R_{on1} varies significantly due to the body effect. As explained in [1], the output phase then varies considerably with the input voltage. We must therefore reduce R_{on1} . Changing W_1 from $10\text{ }\mu\text{m}$ to $20\text{ }\mu\text{m}$ yields the spectrum shown in Fig. 4.5(c), achieving an HD_3 of -66 dB and an HD_5 far below this level. We hereafter assume $W_1 = 20\text{ }\mu\text{m}$.

4.5 Basic Sampler

In the next step of our design effort, we add two ideal switches to Fig. 4.1(a) so that M_1 is turned on and off by the clock [Fig. 4.6(a)]. To create some similarity to the actual circuit, we define an on-resistance of $50\text{ }\Omega$ for M_2 and M_4 .

Figure 4.6(b) plots the output spectrum for $f_{in} = 570$ MHz. The third harmonic is at 1.71 GHz and has a relative level of -83 dB. The fifth harmonic, on the other hand, is aliased to $f_{CK} - 5f_{in} = 2.15$ GHz, exhibiting a normalized level of -110 dB. Next, we raise f_{in} to 2.47 GHz and note that its third harmonic appears at $f_{CK} - 3f_{in} = 2.41$ GHz. Depicted in Fig. 4.6(c), the output spectrum yields $\text{HD}_3 = -64.4$ dB, 1.3 dB higher than the track-mode distortion found above. This can be attributed to the channel charge injected by M_1 onto C_1 in Fig. 4.6(a); even though V_{GS1} is constant, the transistor's threshold is not, modulating the charge as V_{in} varies.

4.6 Sampler with MOS Switches

At this point, we wish to implement M_2 and M_4 in Fig. 4.6(a) by MOS devices. We recognize that M_4 must be an NMOS device because it pulls X to ground, whereas M_2 must be a PMOS device as it ties X to a high potential (Fig. 4.7). But we must also decide to which node the n -well of M_2 should be connected. If attached to X , the n -well is drawn toward the ground by M_4 when M_2 is off, thereby forward-biasing the junctions between the n -well and the source and drain of M_2 . To avoid this issue, we tie the n -well to node P .

The minimum widths of M_2 and M_4 are dictated by the following constraints. We note that R_{on2} and the total capacitance at X —primarily the gate-source capacitance of M_2 —form a low-pass filter, attenuating the input swing as it reaches X . Thus, R_{on2} must be so chosen as to provide a bandwidth far beyond 2.5 GHz. This is readily possible with a W_2 of a few microns. We select $W_2 = 2.5\text{ }\mu\text{m}$. Moreover, M_4 must pull X down at a high slew rate so as to turn M_1 off rapidly. If it does not, the on-resistance of M_1 increases slowly,

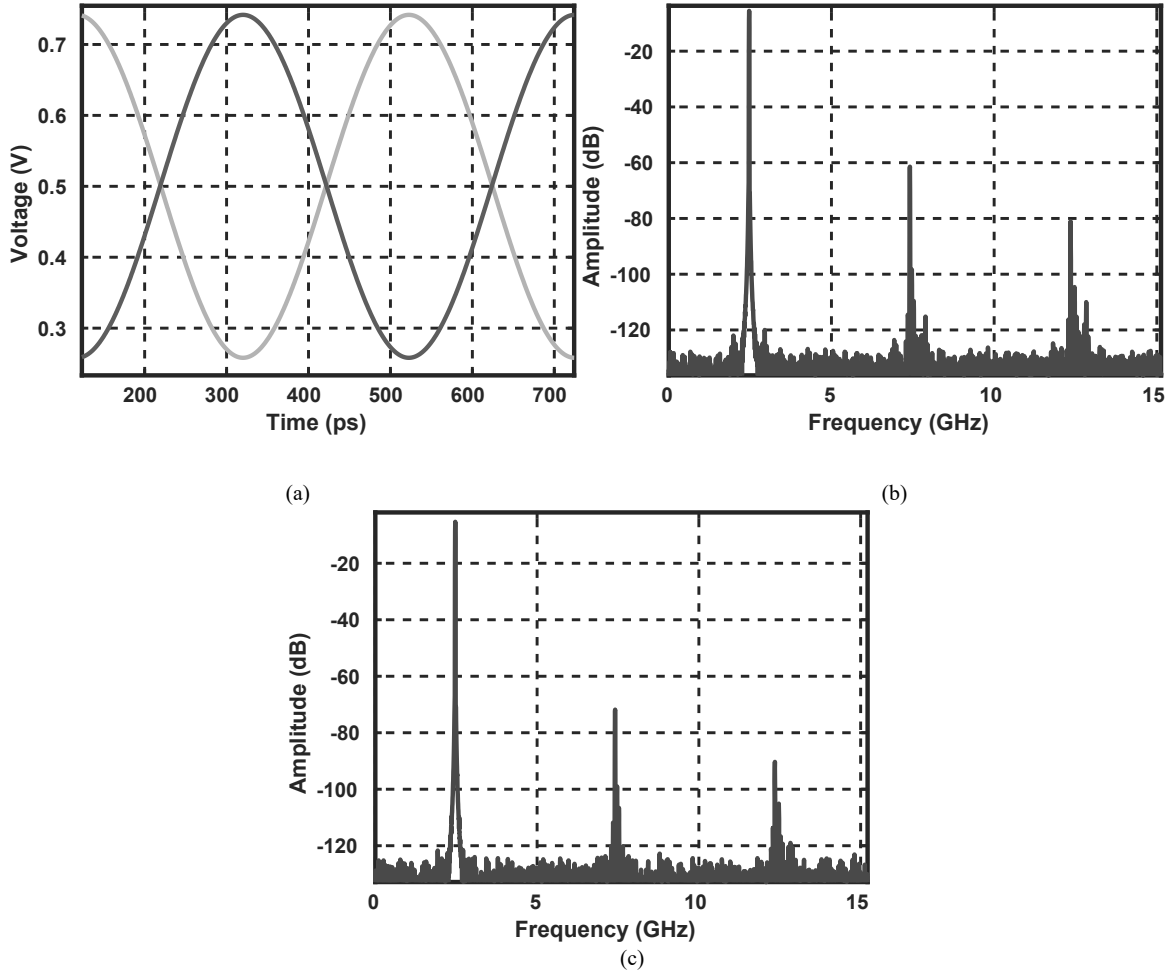


Figure 4.5 (a) The track-mode output waveforms for $f_{in} = 2.47$ GHz, (b) the corresponding spectrum, and (c) reduced distortion as a result of doubling W_1 .

causing distortion. We select $W_4 = 2.5 \mu\text{m}$ for now. For the clock driving these two switches, we choose 10-ps rise and fall times, as a representative in 28-nm technology.

Upon simulating the topology of Fig. 4.7 with $f_{in} = 570$ MHz and 2.47 GHz, we obtain an HD_3 of -68 dB and -58 dB, respectively. Why is the linearity degraded by such a large amount? This is because M_2 fails to turn off for part of the input swing: if $V_{in} > 0.5$ V, then $V_P > 1.5$ V, keeping M_2 on even if its gate is raised to 1 V. Consequently, M_4 cannot pull X to zero, and M_1 does not completely turn off.

To resolve this issue, we must remove the bootstrapping action in the hold mode. That is, the battery must be disconnected from the analog input. We thus add switches M_3 and M_6 [Fig. 4.8(a)]. Now, when CK is high, V_B generates $V_P = V_{DD}$, allowing M_2 to turn off. The gate of M_3 could not be driven by the rail-to-rail clock because the on-resistance of M_3 would vary considerably in the track mode. For this reason, we bootstrap its gate to V_{in} in a manner similar to M_1 . The on-resistance of M_3 should be comparable to that of M_2 and not limit the bandwidth at node X in the track mode. We tentatively choose $W_3 = 2.5 \mu\text{m}$ and $W_6 = 1 \mu\text{m}$.

Simulations indicate an HD_3 of -84 dB and -60 dB for $f_{in} = 570$ MHz and $f_{in} = 2.47$ GHz, respectively. The latter case is shown in Fig. 4.8(b). The circuit performs satisfactorily at this stage of the design.

In the last step, we replace the bootstrapping battery with a capacitor, as indicated in Fig. 4.1(b). This requires that M_5 be added so that C_B can be charged to V_{DD} in the hold mode. Several questions arise. First,

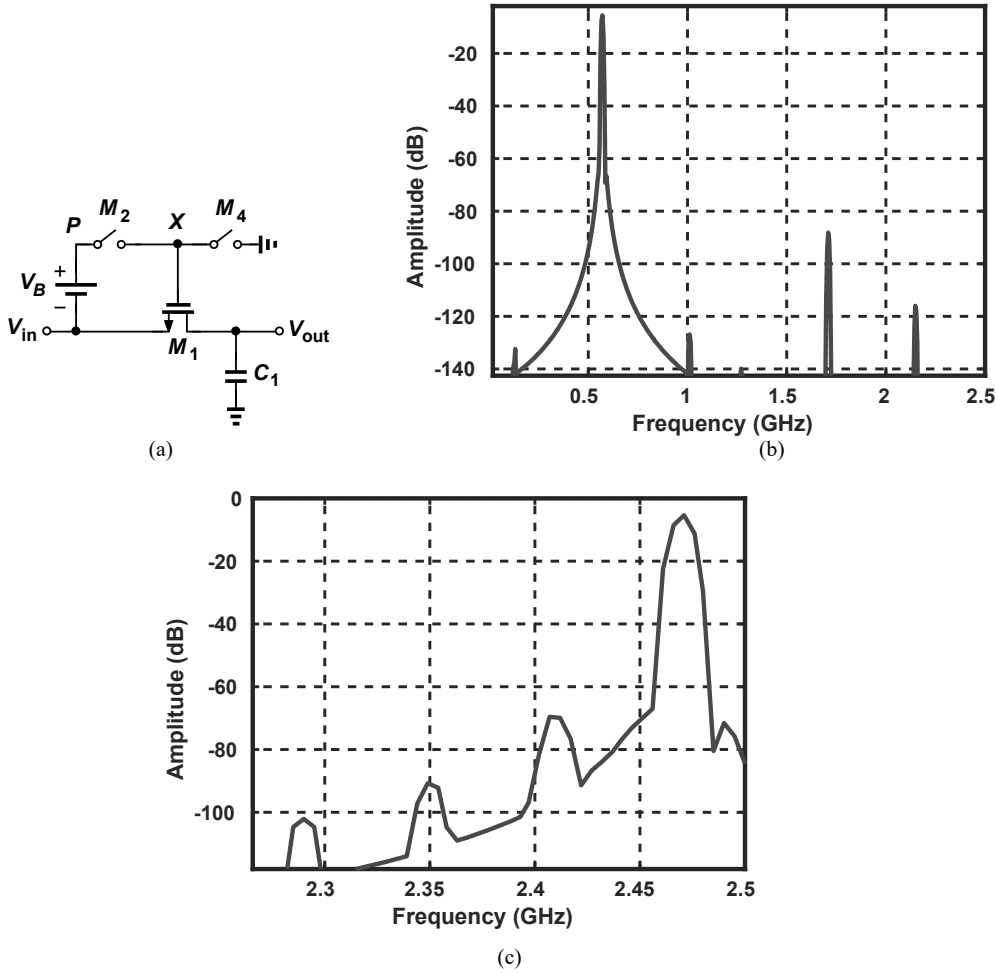


Figure 4.6 (a) Sampler using ideal switches M_2 and M_4 , (b) the output spectrum for $f_{in} = 570$ MHz, and (c) the output spectrum for $f_{in} = 2.47$ GHz.

to which node should the gate of M_5 be connected? If driven as in Fig. 4.9(a), M_5 fails to turn off when $\overline{CK}$ is high because $V_P = V_{in} + V_{DD}$ can reach 1.75 V. We therefore surmise that the gate of M_5 must be bootstrapped to V_{in} as well [Fig. 4.9(b)]. Second, to which node should the n -well of M_5 be connected? For the same reason mentioned above for M_2 , this n -well must be tied to node P .

The third question relates to the minimum acceptable value of C_B . Two effects play a role here. (1) When C_B is switched to X , it experiences charge sharing with the total parasitic capacitance at this node, thereby providing a bootstrapping voltage less than V_{DD} . The parasitics at X include the gate capacitances of M_1 , M_3 , and M_5 (the first two conducting and the third remaining off), and the drain capacitances of M_2 , M_4 , and M_5 . Thus, C_B must be large enough to minimize the voltage loss. (2) For C_B to fully charge to V_{DD} during the hold mode, the series combination of M_5 , C_B , and M_6 must exhibit a time constant below half of the clock period. As an example, with $T_{CK}/2 = 100$ ps and $C_B = 0.25$ pF, the total on-resistance of M_5 and M_6 must remain less than 400 Ω .

We select $C_B = 0.25$ pF and $W_5 = 2.5$ μm in Fig. 4.9(b), and increase W_6 from 1 μm to 2.5 μm . The simulation yields an HD_3 of -63 dB at $f_{in} = 2.47$ GHz. To assess the bootstrapping ability of C_B , we plot its voltage as a function of time [Fig. 4.9(c)]. We observe that C_B charges to only 0.895 V (rather than to $V_{DD} = 0.95$ V) due to the long time constant. Moreover, the bootstrapping voltage is less than 0.83 V due to charge sharing. From the difference between 0.895 V and 0.83 V, we estimate a parasitic capacitance of

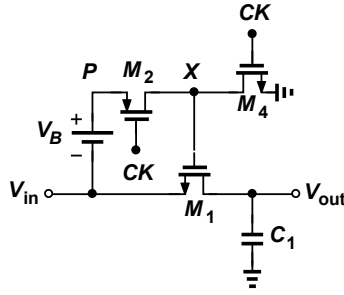


Figure 4.7 Realization of M_2 and M_4 .

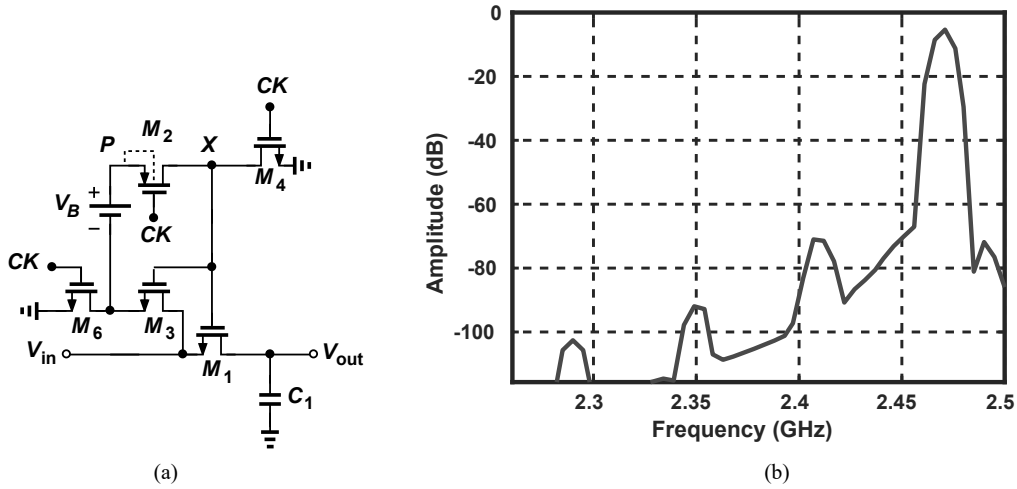


Figure 4.8 (a) Addition of switches to disengage V_B in the hold mode, and (b) output spectrum for $f_{in} = 2.47$ GHz.

20 fF at node X .

To improve the charge replenishment of C_B , we must further widen M_5 and M_6 , and to remedy charge sharing, we must increase the value of C_B . Of course, an excessively wide M_5 raises the parasitic at X and exacerbates the problem of charge sharing. Let us then choose $C_B = 500$ fF, $W_5 = 5 \mu\text{m}$, and $W_6 = 5 \mu\text{m}$. Simulations still suggest an HD_3 of -63 dB at $f_{in} = 2.47$ GHz. We face diminishing returns, partially because the large parasitic at X now creates a long time constant with M_4 , slowing down the turn-off transition of M_1 . This is alleviated by increasing W_4 to $5 \mu\text{m}$. The circuit now exhibits an HD_3 of -65 dB at 2.47 GHz.

4.7 Device Stress Issues

The topology of Fig. 4.9(b) applies a drain-source or gate-source voltage well above 1 V to two of the MOSFETs in the track mode. Called “device stress,” this effect degrades the transistors’ performance over time. We recognize that, when it is off, M_4 experiences a peak drain-source voltage of $V_{in} + V_{DD} \approx 1.75$ V. Similarly, M_2 sees the same voltage difference between its gate and source. To protect the former, we place a cascode device in series with it. As depicted in Fig. 4.10(a), M_8 shields M_4 , ensuring $V_{DS4} < V_{DD}$; M_8 itself sustains a maximum V_{DS} of $1.75 \text{ V} - (V_{DD} - V_{TH4}) \approx 1.05$ V. The series action of these two devices demands that they both be $10 \mu\text{m}$ wide.

The solution for reducing the stress on M_2 in Fig. 4.9(b) is more complex. We note that the only possibility is to allow the gate voltage of M_2 to change in unison with V_X and hence with V_{in} . That is, this gate must be

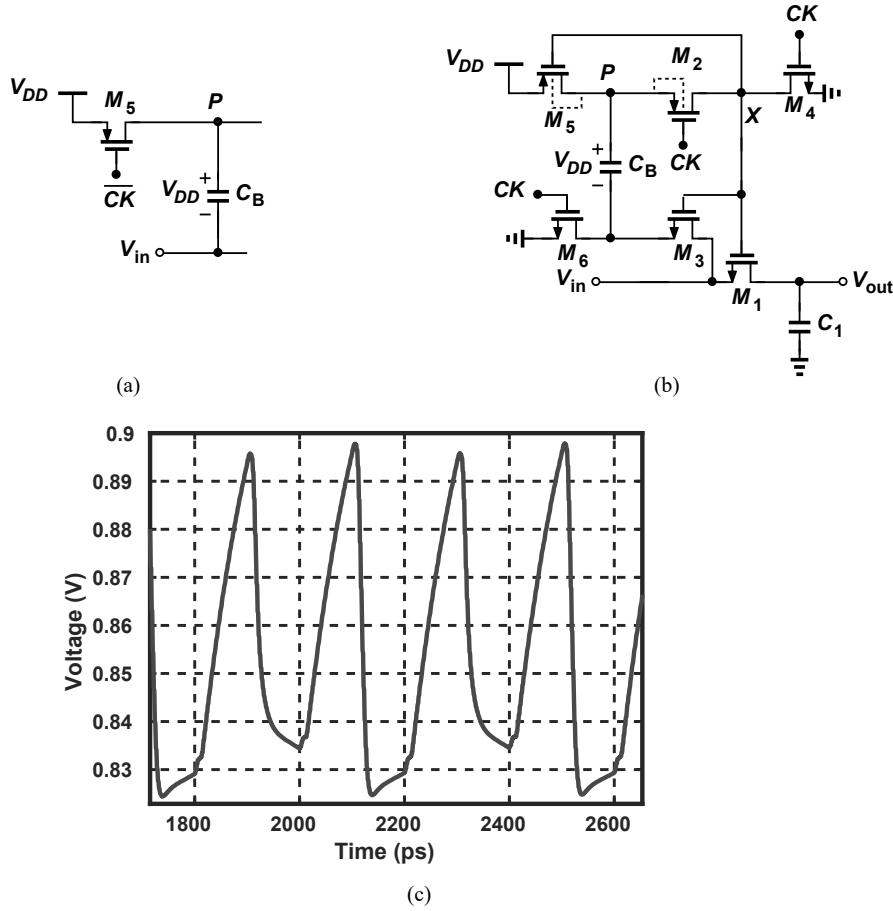


Figure 4.9 (a) A PMOS pull-up device driven by $\overline{CK}$, (b) bootstrapping the gate of M_5 to V_{in} through V_X , and (c) the voltage across C_B .

connected to V_{in} in the track mode and to V_{DD} in the hold mode. Shown in Fig. 4.10(b) is an implementation example where M_a and M_b perform these tasks, respectively. However, if $\overline{CK}$ reaches only V_{DD} , then M_a turns off as V_{in} approaches its peak value of 0.75 V. This issue is resolved as depicted in Fig. 4.10(c), where M_c , with its gate-source voltage bootstrapped to V_{in} , provides a low resistance in series with the gate of M_2 . While appearing redundant, M_a is still necessary for proper startup of the circuit. We select $W_a = 0.25 \mu\text{m}$, $W_b = 1 \mu\text{m}$, and $W_c = 2.5 \mu\text{m}$, the last one wide enough not limit the bandwidth at the gate of M_2 .

Simulation of the modified circuit yields an HD_3 of -63.4 dB . We surmise that the bandwidth at the gate of M_2 is still limited by the on-resistance of M_c . We then double W_c . The final design is shown in Fig. 4.11(a) and its output spectrum in Fig. 4.11(b). The HD_3 is -64.3 dB , the HD_5 is about 20 dB lower, and the total power consumption is around 1 mW. The principal factor limiting the linearity is that the clock pulses arriving at node X are only about 80 ps wide, providing insufficient track time.

4.8 Input Current Issues

Recall that we initially selected an on-resistance of about 45Ω for the sampling switch in Fig. 4.1(a) to ensure an attenuation less than 0.5 dB at $f_{in} = 2.5 \text{ GHz}$. We later doubled the width of M_1 so as to meet the linearity requirement. With $R_{on1} \approx 23 \Omega$, the circuit draws a large input current when operating near the Nyquist rate. This is because the previous sample on C_1 and the present input value at the beginning of the track mode can differ by as much as the entire single-ended swing, about 0.5 V. Thus, M_1 carries an initial current of $0.5 \text{ V}/R_{on1} \approx 22 \text{ mA}$. This current must be provided by the stage preceding the sampler, typically a buffer.

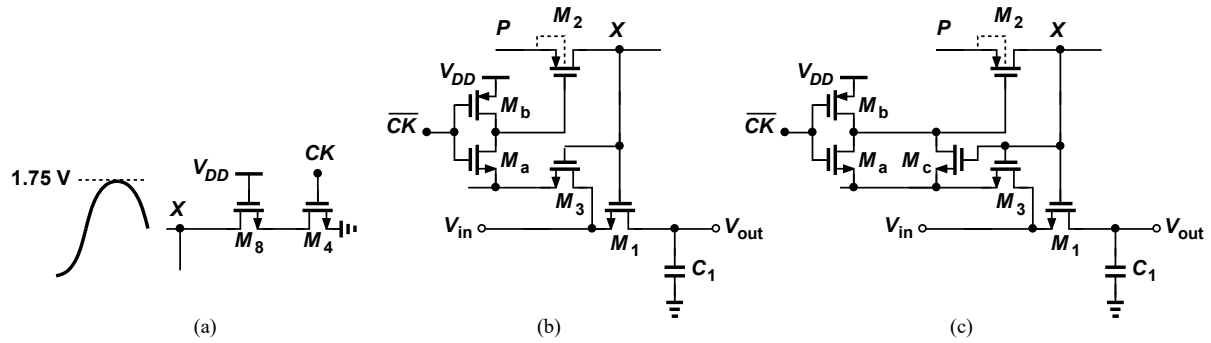


Figure 4.10 (a) Use of M_8 to shield M_4 , (b) addition of M_a and M_b to bootstrap the gate of M_2 , and (c) addition of M_c to provide a low resistance in series with the gate of M_2 .

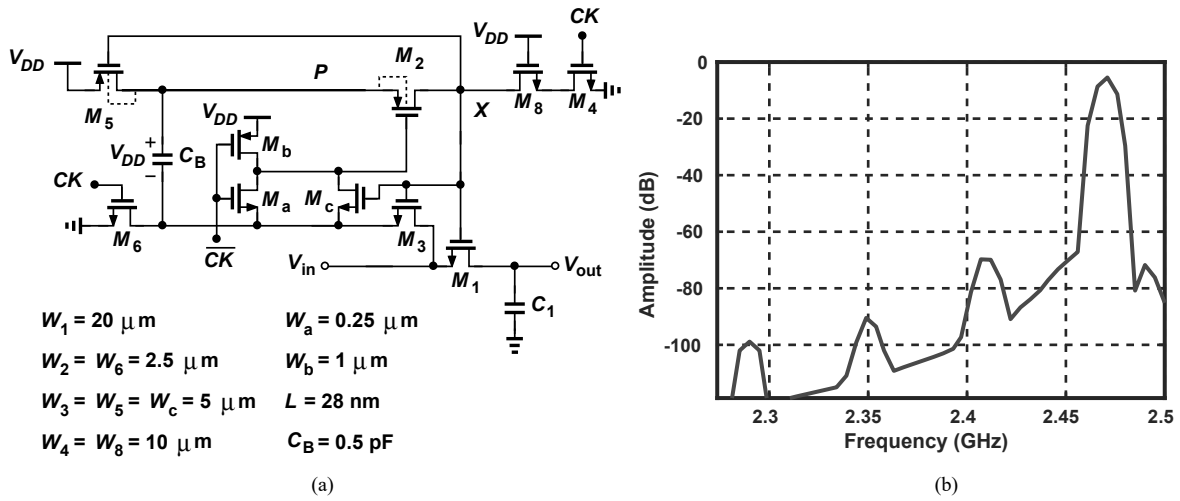


Figure 4.11 (a) The final design, and (b) its output spectrum for $f_{in} = 2.47$ GHz.

From another perspective, if the buffer must not limit the sampling speed, its output impedance must be well below $23\ \Omega$, a great challenge in circuit design.

References

- [1]. B. Razavi, "The bootstrapped switch," *IEEE Solid-State Circuits Magazine*, pp. 12-15, Summer 2015.

Problems

- 4.1. An engineer mistakenly replaces switch M_3 in Fig. 4.1(b) with a short circuit. Explain the issue(s) arising from this error.
- 4.2. Repeat the previous problem if the engineer replaces M_2 with a short circuit.
- 4.3. We wish to limit the kT/C noise penalty given by Eq. (4.1) to 2 dB. Compute the necessary value of the sampling capacitor.
- 4.4. With the capacitor value found in the previous problem, how wide should M_1 in Fig. 4.1(b) be for a 0.5-dB signal attenuation at the Nyquist rate?
- 4.5. A sampler introduces fifth-order nonlinearity. Repeat the analysis of Fig. 4.3(d) for this case if (a) $f_{in} < f_{CK}/5$, or (b) $f_{CK}/5 < f_{in} < 3f_{CK}/10$.
- 4.6. Consider a bootstrapped sampler in the track mode. Is it possible for the circuit to introduce distortion in this mode for a low-frequency analog input? Explain.
- 4.7. The circuit of Fig. 4.1(b) exhibits a third harmonic at -66 dB. With a sampling capacitance of 0.5 pF, estimate from Chapter 3 the peak-to-peak variation of the sampling switch on-resistance.
- 4.8. Suppose the distortion observed in Fig. 4.6(a) arises entirely from the switch on-resistance variation studied in Chapter 3. If we halve the value of C_1 , how much improvement do we expect in HD_3 and HD_5 ?
- 4.9. Suppose node X in Fig. 4.7 incurs some capacitance, C_X , to ground. Assume CK rises at $t = 0$. Viewing M_4 as a current source for simplicity, determine at what time M_1 turns off. Note that the turn-off time of M_1 depends on the value of V_{in0} . This effect produces distortion.
- 4.10. A student decides to replace M_2 in Fig. 4.7 with an NMOS device and drive its gate with $\overline{CK}$. Explain which aspect(s) of the performance may degrade.
- 4.11. A student decides to connect the gate of M_6 in Fig. 4.8(a) to node X rather than to CK . Explain whether the circuit operates properly.
- 4.12. Determine HD_5 and HD_7 in Fig. 4.8(b).
- 4.13. If V_{in} in Fig. 4.8(a) swings between V_1 and V_2 , determine the maximum and minimum gate-source voltages of M_3 in the track mode and in the hold mode.
- 4.14. Suppose the clock applied to the gates of M_2 and M_4 in Fig. 4.9(b) exhibits a slow rising edge, during which both transistors are turned on. Explain which aspects of the performance degrade.
- 4.15. If V_{in} in Fig. 4.9(b) swings between V_1 and V_2 , determine the maximum and minimum drain-source voltages of M_6 and M_3 in the track mode and in the hold mode.
- 4.16. A student decides to change M_1 in Fig. 4.9(b) to a PMOS device. Draw the entire bootstrapping network for this carefully.
- 4.17. In the circuit of Fig. 4.10(a), does M_8 turn off or on more slowly? (Hint: consider how M_8 behaves when M_4 turns on or turns off.)
- 4.18. In Fig. 4.10(b), explain in which region of operation M_1 , M_2 , and M_3 reside when $\overline{CK}$ is high or low.
- 4.19. A bootstrapped sampling circuit based on the topology of Fig. 4.10(c) operates with a clock frequency of 5 GHz. Sketch the gate voltage of M_2 versus time as the circuit goes through track and hold modes. Assume V_{in} is a slow sinusoid swinging between V_1 and V_2 .
- 4.20. Suppose V_{in} in Fig. 4.11(a) varies between V_1 and V_2 . If the bootstrap capacitor holds a voltage equal to V_{DD} , determine the maximum and minimum gate voltages of M_C in the track mode and in the hold mode.
- 4.21. Repeat the previous problem for M_5 .
- 4.22. The on-resistance of M_1 in Fig. 4.11(a) is about $23\ \Omega$. We have seen that the circuit can draw a peak current of 22 mA from the input. But suppose the preceding stage is modeled by a sinusoidal Thevenin voltage and an output resistance of $10\ \Omega$. Sketch the waveform of V_{in} as the circuit enters the track mode if C_1 has a zero initial voltage. Explain which aspect(s) of the performance may degrade.

Comparator Design Principles

All analog-to-digital converters employ comparators. For this reason, we deal with comparator design before turning to ADC architectures. We begin with a brief study of offsets and then present some basic concepts. Next, we describe a number of comparator topologies. We also investigate such design aspects as speed and noise. Our studies culminate in a popular structure called the “StrongArm” comparator.

5.1 A Brief Note on Offsets

Offset voltages prove problematic in ADC design and merit a detailed treatment (Section 5.7.1). However, a basic understanding of offsets is necessary even for our initial study of comparators. In this section, we briefly look at this phenomenon. Let us begin with a simple example.

Example 5.1

Prove that the arrangement in Figs. 5.1(a) and 5.1(b) are identical.

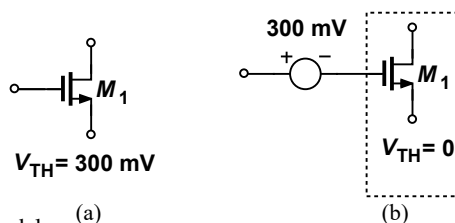


Figure 5.1 Two MOS device models.

Solution

We know that some of the transistor equations involve V_{TH} . But these equations are always a function of the overdrive voltage, $V_{GS} - V_{TH}$. Thus, we artificially change V_{TH} and compensate for it by inserting a voltage source in series with the gate so that the overdrive remains constant. This partitioning is arbitrary: the voltage source can have a value of, say, 200 mV while the transistor threshold is assumed to be 100 mV. The reader is encouraged to determine whether the voltage source can instead be placed in series with the source terminal of the transistor.

We now consider the simple differential pair shown in Fig. 5.2(a). If M_1 and M_2 are perfectly “matched,” their drain currents vary symmetrically around $V_{in1} - V_{in2} = 0$, assuming a value of $I_{SS}/2$ at this point. In reality, however, the two transistors suffer from random *asymmetries* as they are manufactured. For example, in one of the chips, we can have $V_{TH1} = 305$ mV and $V_{TH2} = 300$ mV [Fig. 5.2(b)]. How does this

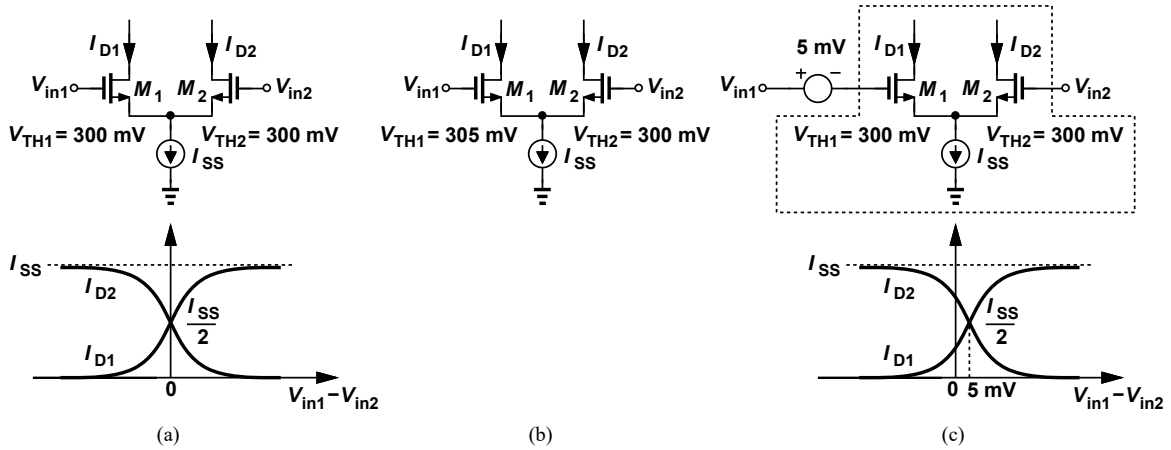


Figure 5.2 (a) Differential pair with no offset and its characteristic, (b) inclusion of threshold mismatch, and (c) mismatch model and resulting characteristic.

“threshold mismatch” affect the input-output characteristic of the differential pair? From Example 5.1, we note that the 5-mV change in V_{TH1} can be modeled by a voltage source in series with the gate of M_1 [Fig. 5.2(c)]. The circuit in the dashed box is now symmetric, and the net voltage sensed by M_1 and M_2 in this model is equal to $V_{in1} - 5 \text{ mV} - V_{in2} = V_{in1} - V_{in2} - 5 \text{ mV}$. We then recognize that $V_{in1} - V_{in2}$ must reach +5 mV for $I_{D1} = I_{D2}$, observing a shift in the characteristic to the right by this amount. We call $\Delta V_{TH} = V_{TH1} - V_{TH2}$ the “input-referred offset voltage,” V_{OS} , of the circuit.

Example 5.2

Sketch the input-output characteristic of the differential pair shown in Fig. 5.3(a) if $V_{TH1} = V_{TH2}$ or $V_{TH1} - V_{TH2} = \Delta V_{TH}$.

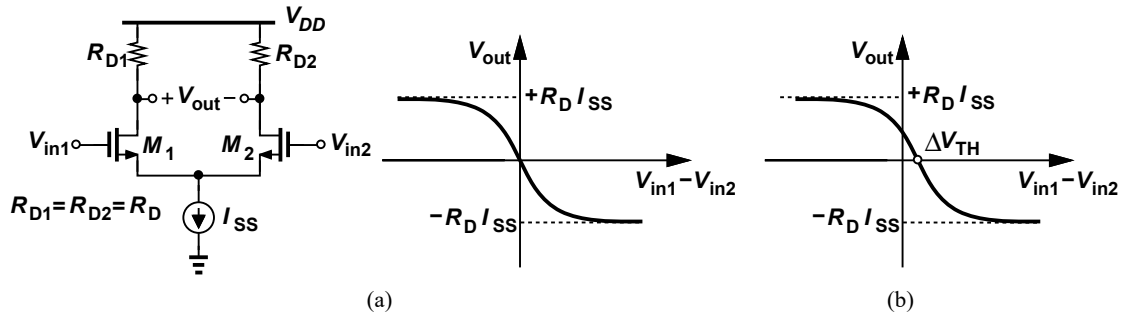


Figure 5.3 Simple differential pair and its characteristic for (a) matched devices, and (b) mismatched devices.

Solution

Without mismatches, V_{out} is equal to zero for $V_{in1} = V_{in2}$ and equal to $+R_D I_{SS}$ and $-R_D I_{SS}$ for a sufficiently negative or positive $V_{in1} - V_{in2}$, respectively [Fig. 5.3(b)] (why?). In the presence of threshold mismatch, $V_{in1} - V_{in2}$ must reach ΔV_{TH} for $I_{D1} = I_{D2}$ and $V_{out} = 0$. That is, the characteristic is shifted to the right by an amount equal to ΔV_{TH} [Fig. 5.3(c)].

Two key points emerge from this brief study. First, a nominally-symmetric differential circuit in reality suffers from random asymmetries. As a result, the circuit’s characteristic is shifted to the right or to the left.

Second, the effect of asymmetries can be modeled by an input-referred offset voltage. Since the offset has a random sign, it can be in series with either input.

For more complex situations, we need a methodical approach to calculating the input offset, V_{OS} . For example, if R_{D1} and R_{D2} in Fig. 5.3(a) suffer from mismatch, how do we obtain the resulting V_{OS} ?

A simple and intuitive approach to offset calculation is as follows. We begin with the actual “mismatched” circuit [Fig. 5.4(a)]. Due to asymmetries, this circuit does not generate $V_{out1} - V_{out2} = 0$ unless

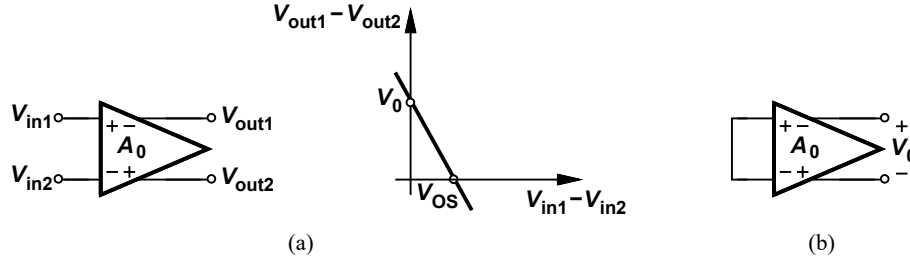


Figure 5.4 (a) Circuit with offset, and (b) effect on output with zero input.

$V_{in1} - V_{in2} = V_{OS}$. We can then say V_{OS} is the “external” voltage that must be applied to the input so that the output goes to zero at $V_{in1} - V_{in2} = 0$. But a different perspective can simplify our task. Recognizing that (a) at $V_{in1} - V_{in2} = 0$, we have $V_{out1} - V_{out2} = V_0$, and (b) V_0/V_{OS} is equal to the slope of the characteristic and hence the small-signal gain, A_0 , we can write

$$V_{OS} = \frac{V_0}{A_0}. \quad (5.1)$$

The key point here is that we set the input difference to *zero* [Fig. 5.4(b)], obtain the output difference, V_0 , (also called the “output offset”), and divide it by the gain to obtain V_{OS} . The reader may notice the similarity between this procedure and that for input-referred noise calculation.

5.2 Basic Concepts

Suppose we wish to determine whether an analog input, V_{in} , is less or greater than zero. Our objective is to generate a logical output, V_{out} , that conveys the result of this comparison, e.g.,

$$V_{out} = \text{Low if } V_{in} < 0 \quad (5.2)$$

$$= \text{High if } V_{in} > 0. \quad (5.3)$$

This function can be approximated by a high-gain amplifier. As depicted in Fig. 5.5(a), if $|V_{in}|$ is sufficiently large, the output “saturates” and provides an unambiguous low or high output. This means that A_0 must be large enough to allow the comparator to resolve small input levels.

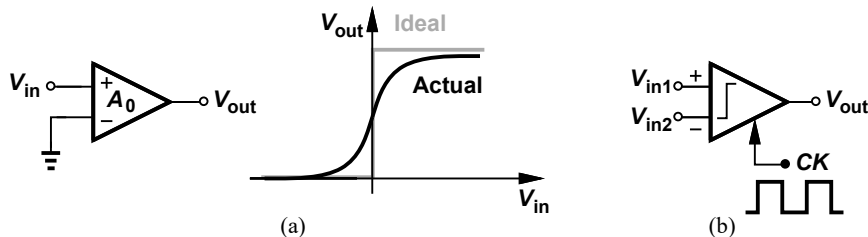


Figure 5.5 (a) Use of a high-gain amplifier as a comparator, (b) use of clocking to define comparison time.

In ADC design, the comparator must perform comparison at certain points in time, i.e., it is clocked. Moreover, it generally compares two non-zero voltages. We thus arrive at the conceptual topology shown in Fig. 5.5(b), where V_{out} is low if $V_{in1} < V_{in2}$ and vice versa. But how exactly does the clock control the comparator? In other words, how should V_{out} behave when CK is high and when it is low? We will answer these questions carefully after a detailed study of comparators; for now, we say the comparator “makes a decision” when CK goes high. This means that the comparator detects the *polarity* of $V_{in1} - V_{in2}$ and generates a logical high or low level at the output.

Example 5.3

Consider the arrangement depicted in Fig. 5.6(a), where V_{r1} is a constant voltage. Plot V_{out} as a function of V_{in} .

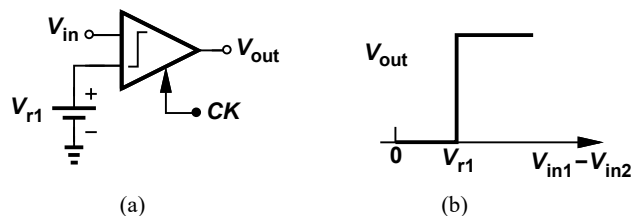


Figure 5.6 (a) Comparator sensing V_{in} and V_{r1} , and (b) its input-output characteristic.

Solution

Even though the circuit is clocked here, we seek an input-output characteristic similar to that in Fig. 5.5(a). That is, we select V_{in} less than V_{r1} , clock the comparator and find V_{out} . We then increment V_{in} and repeat the experiment. We note that V_{out} is low if $V_{in} < V_{r1}$ and high if $V_{in} > V_{r1}$, thus obtaining the characteristic shown in Fig. 5.6(b). The reader may wonder what happens if $V_{in} = V_{r1}$. We examine this situation (called “metastability”) in Chapter 6.

Example 5.4

In the circuit of Fig. 5.7(a), two comparators compare the main input, V_{in} , to two “reference” voltages, V_{r1} and V_{r2} . These voltages can be readily generated by resistors. Plot V_{out1} and V_{out2} as a function of V_{in} .

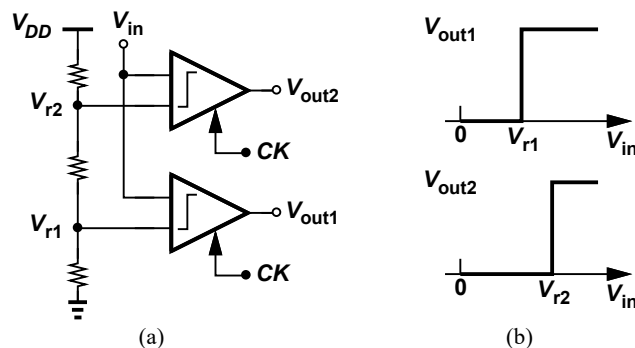


Figure 5.7 (a) Two comparators sensing different reference voltages, and (b) their input-output characteristics.

Solution

Following our analysis in the previous example, we recognize that V_{out1} changes as V_{in} crosses V_{r1} , and

V_{out2} changes as V_{in} crosses V_{r2} [Fig. 5.7(b)]. We denote the logical values by A_1 and A_2 , respectively, and write

$$A_1 A_2 = 00 \text{ if } V_{in} < V_{r1} \quad (5.4)$$

$$= 10 \text{ if } V_{r1} < V_{in} < V_{r2} \quad (5.5)$$

$$= 11 \text{ if } V_{r2} < V_{in}. \quad (5.6)$$

Comparator Offset The conceptual comparator in Fig. 5.5(b) must employ a differential pair at its input to sense $V_{in1} - V_{in2}$. As explained in the previous section, this pair suffers from random asymmetries. Also, the comparator may incorporate other differential stages beyond the input pair, exhibiting additional mismatches. We model the effect of all of the asymmetries by an input-referred offset, as shown in Fig. 5.8(a).

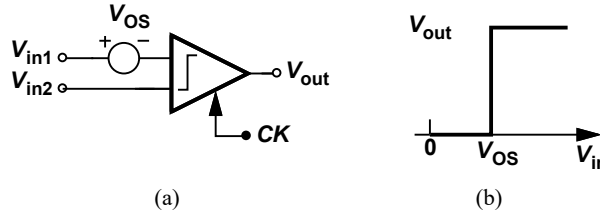


Figure 5.8 (a) Comparator with offset, and (b) its characteristic.

The comparator offset has significant implications in some ADC architectures. Depicted in Fig. 5.8(b), the circuit's input-output characteristic is shifted horizontally by V_{OS} , leading to an *incorrect* decision if $0 < V_{in1} - V_{in2} < V_{OS}$. We may design comparators such that their offset is a small fraction of the ADC's LSB, e.g., in the range of 0.1 LSB to 0.2 LSB.

5.3 Amplification by Positive Feedback

Our preliminary look at comparators suggests that they must achieve a high gain so as to produce a valid logical output. In fact, the ideal, abrupt characteristics in Figs. 5.6(b) and 5.7(b) imply an infinite slope and hence an infinite gain. But, as a more realistic example, suppose the comparator must deliver a high output equal to 1 V while sensing an input difference of 1 mV. How do we realize a gain of 1,000? An op amp is *not* the answer here as it would consume high power and suffer from a low speed. We must therefore seek other methods of amplification.

An important technique that comes to our aid here is amplification by positive feedback. Let us take a differential pair and connect its outputs to its inputs in a positive-feedback loop [Fig. 5.9(a)]. Here, C_L denotes the total capacitance from X or Y to ac ground. Redrawing this circuit as the “cross-coupled” pair shown in Fig. 5.9(b), we wish to examine its behavior even though it appears to have no input! If the circuit begins with $V_X = V_Y$, then each transistor operates in saturation (why?) and carries half of the tail current. That is, V_X and V_Y reside at a common-mode (CM) level equal to $V_{DD} - R_D I_{SS}/2$. The cross-coupled pair seems to retain this state indefinitely. We analyze the circuit in both frequency and time domains.

Frequency-Domain Perspective In this analysis, we proceed in three steps. First, we know that a linear, time-invariant system having a transfer function $H(s)$ and sensing an input of the form $e^{s_1 t}$ produces an output equal to $H(s_1)e^{s_1 t}$. (The reader is encouraged to prove this.) In particular, if $H(s_1) = 1$, then the input exponential travels through the system unchanged.

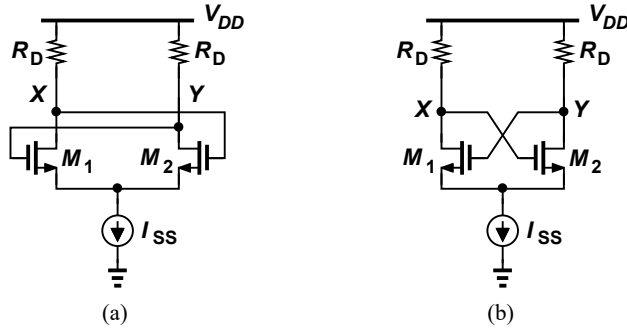


Figure 5.9 (a) Differential pair in positive-feedback loop, and (b) redrawn circuit.

Second, consider the positive-feedback loop¹ shown in Fig. 5.10(a), noting that

$$\frac{Y}{X} = \frac{H(s)}{1 - H(s)}. \quad (5.7)$$

If $H(s = s_1) = 1$, then $Y/X \rightarrow \infty$, i.e., the loop provides an infinite gain. This means that a signal of the form $e^{s_1 t}$ circulates around the loop indefinitely because it experiences a gain of unity through $H(s)$ [Fig. 5.10(b)]. The exponential continues to grow if s_1 is real and positive.

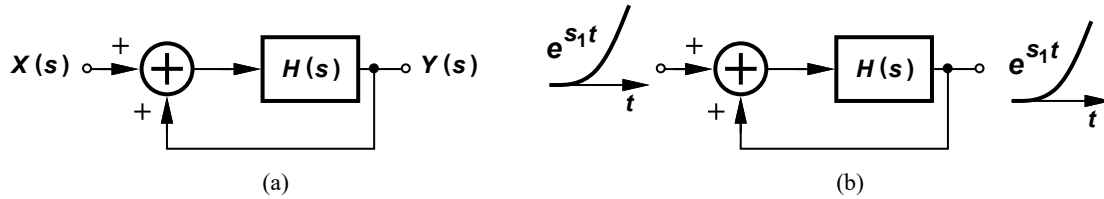


Figure 5.10 (a) System with positive feedback, and (b) growing exponential circulating around the loop.

Third, if $H(s) = A_0/(1 + s/\omega_0)$, then, s_1 is obtained by writing

$$\frac{A_0}{1 + \frac{s_1}{\omega_0}} = 1 \quad (5.8)$$

and hence

$$s_1 = (A_0 - 1)\omega_0. \quad (5.9)$$

For the differential pair, we have $A_0 = g_m R_D$ and $\omega_0 = 1/(R_D C_L)$, where C_L denotes the capacitance at X and Y to ground. It follows that

$$s_1 = \frac{g_m R_D - 1}{R_D C_L}. \quad (5.10)$$

Thus, the loop in Fig. 5.9(a) can sustain a growing exponential if $s_1 > 0$, i.e., if $g_m R_D > 1$.

¹By positive feedback, we mean the loop contains no net inversion at low frequencies.

Time-Domain Perspective In the next step of our analysis, we wish to see what happens in the time domain if the cross-coupled pair begins with a small imbalance. But let us first consider a simpler situation: suppose the differential pair shown in Fig. 5.11 senses a small input difference of the form $V_{in1} = V_{CM} + \Delta V$ and $V_{in2} = V_{CM} - \Delta V$, where V_{CM} denotes the input common-mode level. We note that

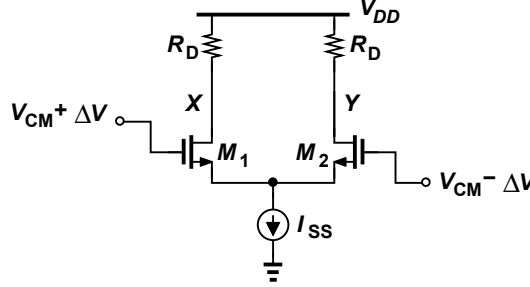


Figure 5.11 Simple differential pair.

$V_X = V_{DD} - R_D I_{SS}/2 - g_m R_D \Delta V$ and $V_Y = V_{DD} - R_D I_{SS}/2 + g_m R_D \Delta V$. The key point here is that if the gate voltages of M_1 and M_2 depart from their CM level by $+\Delta V$ and $-\Delta V$, respectively, then $V_X - V_Y = -2g_m R_D \Delta V$.

We now turn to the cross-coupled pair illustrated in Fig. 5.12, assuming $R_{D1} = R_{D2} = R_D$. For reasons that become clear later, we include the parasitic capacitances of the circuit as C_{L1} and C_{L2} and

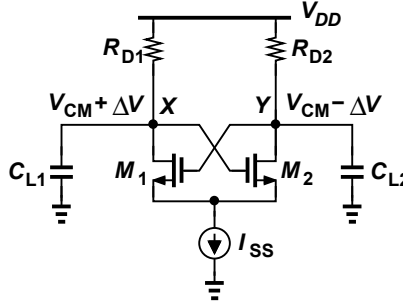


Figure 5.12 Positive-feedback pair with load capacitances.

impose initial conditions on C_{L1} and C_{L2} equal to $V_{CM} + \Delta V$ and $V_{CM} - \Delta V$, respectively, where $V_{CM} = V_{DD} - R_D I_{SS}/2$. Can the circuit retain this state indefinitely? In Fig. 5.11, a difference of $2\Delta V$ between the gate voltages of M_1 and M_2 leads to an imbalance of $-2g_m R_D \Delta V$ between their drain voltages, but here the imbalance is $+2\Delta V$. Thus, the cross-coupled pair cannot remain in this state.

Example 5.5

Can we predict in which directions V_X and V_Y will change in Fig. 5.12?

Solution

Yes, we can. Let us consider the currents flowing into and out of X at $t = 0$, assuming that $R_{D1} = R_{D2} = R_D$ and $g_{m1} = g_{m2} = g_m$. We observe that the gate voltage of M_1 is ΔV volts lower than V_{CM} , yielding a drain current equal to

$$I_{D1} = \frac{I_{SS}}{2} - g_{m1} \Delta V. \quad (5.11)$$

Similarly,

$$I_{D2} = \frac{I_{SS}}{2} + g_{m2} \Delta V. \quad (5.12)$$

Moreover, the current through R_{D1} is reduced to $[V_{DD} - (V_{CM} + \Delta V)]/R_{D1} = I_{SS}/2 - \Delta V/R_{D1}$ (Fig. 5.13). This means that the current through R_{D1} is *not* equal to I_{D1} unless $g_{m1}R_D = 1$. Suppose $g_{m1}R_D > 1$. Then, a net current flows *from* R_{D1} *to* C_{L1} . Considering the direction of the current flowing through C_{L1} , we conclude that V_X rises (and V_Y falls). We also recognize that, in the absence of C_{L1} and C_{L2} , the drain and resistor current differences would have nowhere to go, rendering the circuit meaningless with our assumed

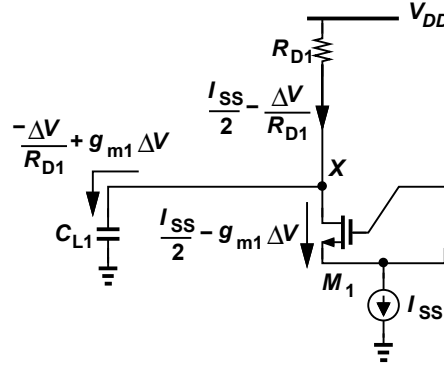


Figure 5.13 Effect of imbalance on one side of cross-coupled pair.

initial conditions.²

How do V_X and V_Y in Fig. 5.13 evolve with time? Suppose V_Y falls by some amount. Then V_{GS1} and I_{D1} decrease, helping V_X rise. This rise now increases I_{D2} , speeding up the discharge at Y . Observing that the positive feedback around the loop tends to accelerate the changes in V_X and V_Y , we predict an exponential behavior for these voltages (Fig. 5.14). We say the circuit “regenerates.”

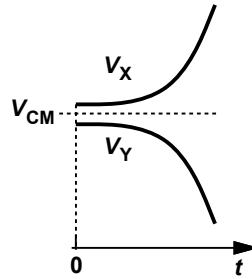


Figure 5.14 Exponential growth due to positive feedback.

Example 5.5 provides the framework for quantifying the behavior of V_X and V_Y . Alternatively, we can perform a small-signal analysis in the time domain. Shown in Fig. 5.15, where $C_{L1} = C_{L2} = C_L$, the small-signal model represents how the circuit evolves if it begins with an initial imbalance, e.g., $V_X = V_{X0}$, $V_Y = V_{Y0}$. Writing a KCL at X gives

$$C_L \frac{dV_X}{dt} + \frac{V_X}{R_D} + g_m V_Y = 0. \quad (5.13)$$

²This would be similar to imposing an initial condition at the output of a simple resistive divider. The situation is meaningless if this condition is not equal to divider’s output voltage.

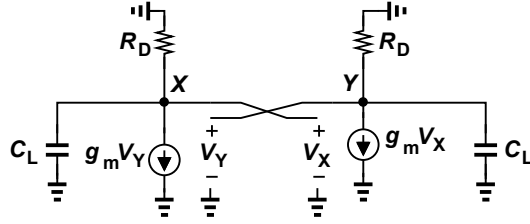


Figure 5.15 Small-signal model of cross-coupled pair.

Similarly, we have at Y

$$C_L \frac{dV_Y}{dt} + \frac{V_Y}{R_D} + g_m V_X = 0. \quad (5.14)$$

Our analysis is simplified if we consider only the differential output, $V_{XY} = V_X - V_Y$. Subtracting (5.14) from (5.13) yields

$$C_L \frac{dV_{XY}}{dt} + \frac{V_{XY}}{R_D} - g_m V_{XY} = 0, \quad (5.15)$$

and hence

$$C_L \frac{dV_{XY}}{dt} = \frac{g_m R_D - 1}{R_D} V_{XY}. \quad (5.16)$$

It follows that

$$\int_{V_{XY0}}^{V_{XY}} \frac{R_D C_L}{g_m R_D - 1} \frac{dV_{XY}}{V_{XY}} = \int_0^t dt. \quad (5.17)$$

We thus have

$$V_{XY} = V_{XY0} \exp \frac{t}{\frac{R_D}{g_m R_D - 1} C_L}. \quad (5.18)$$

This result provides several insights into the behavior of the cross-coupled pair. First, the exponential has a *positive* exponent if $g_m R_D > 1$, growing rapidly with time. This point stands in contrast to *decaying* exponentials that we observe in simple first-order systems. Second, the “regeneration time constant” is given by

$$\tau_{reg} = \frac{R_D}{g_m R_D - 1} C_L. \quad (5.19)$$

Interestingly, this quantity is less than the natural time constant, $R_D C_L$, by a factor of $g_m R_D - 1$. For example, if $R_D = 1 \text{ k}\Omega$, $C_L = 5 \text{ fF}$, and $g_m = 1/(100 \text{ }\Omega)$, we have $\tau_{reg} = 0.56 \text{ ps}$. That is, V_{XY} grows by a factor of $e = 2.718$ every 0.56 ps . If $g_m R_D \gg 1$, then $\tau_{reg} \approx C_L/g_m$. Third, by virtue of positive feedback, the circuit can provide an infinite gain if given infinite time. In other words, for an arbitrarily small V_{XY0} , we can simply wait for the exponential to grow to a desired value. By contrast, amplifiers studied in basic analog design do not provide an infinite gain. Fourth, if $g_m R_D < 1$, the exponential *decays* and there is no amplification, an undesirable condition.

Example 5.6

Identify the components of C_{L1} and C_{L2} in Fig. 5.12.

Solution

As illustrated in Fig. 5.16(a), each transistor presents its C_{GS} from one node to ground and its C_{DB} from the

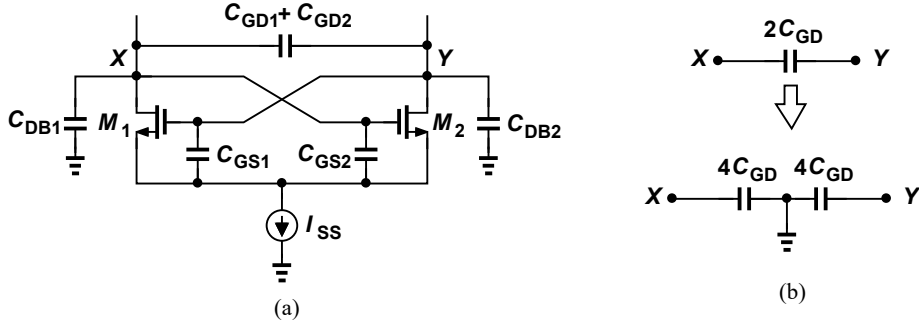


Figure 5.16 (a) Illustration of transistor capacitances, and (b) equivalent circuit for C_{GD} .

other node to ground. In addition, C_{GD1} and C_{GD2} appear between X and Y . Noting that V_X and V_Y change differentially, we can consider $C_{GD1} + C_{GD2}$ as the series combination of two capacitances, each equal to $2(C_{GD1} + C_{GD2})$ [Fig. 5.16(b)]. That is, a capacitance of $4C_{GD}$ equivalently appears from X to ground and from Y to ground.³ We then have

$$C_{L1} = C_{L2} = C_{GS} + C_{DB} + 4C_{GD}. \quad (5.20)$$

How far do V_X and V_Y in Fig. 5.12 travel? As the regeneration continues, M_1 begins to turn off, allowing M_2 to carry all of I_{SS} . Eventually, $I_{D1} = 0$ and $V_X = V_{DD}$, while $I_{D2} = I_{SS}$ and $V_Y = V_{DD} - R_D I_{SS}$. That is, the circuit delivers a differential output equal to $R_D I_{SS}$. The reader is encouraged to reexamine Eq. (5.18) for this case.

It is instructive to study the trade-offs in the design of the cross-coupled pair shown in Fig. 5.12. To minimize τ_{reg} , we pursue one of the following approaches. (1) We can *increase* R_D because this would allow more of the drain currents to flow through the capacitances; however, a larger R_D translates to greater voltage headroom consumption, a serious issue in low-voltage design. In older technologies and with higher supply voltages, we could ensure $g_m R_D \gg 1$ and hence $\tau_{reg} \approx C_L / g_m$. (2) We can increase g_m by widening M_1 and M_2 , in which case their capacitances rise proportionally, i.e., τ_{reg} does not change much in this case.⁴ (3) We can increase g_m by raising I_{SS} , but the greater drops across R_{D1} and R_{D2} and the larger V_{GS1} and V_{GS2} prohibit operation with a low supply voltage.

Example 5.7

It is possible to incorporate both cross-coupled NMOS and PMOS transistors in a positive-feedback loop, as shown in Fig. 5.17. Here, the load resistors are replaced with M_3 and M_4 . Determine the regeneration time constant of the circuit. Use an N or P subscript for g_m and r_O of the NMOS and PMOS devices, respectively.

³This equivalence can also be seen from the Miller effect.

⁴In fact, τ_{reg} increases because $g_m \propto \sqrt{W}$ whereas $C_L \propto W$.

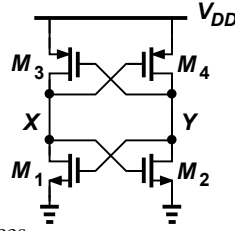


Figure 5.17 Cross-coupled complementary devices.

Solution

The small-signal model of this regenerative pair is identical to that in Fig. 5.15 if we make these substitutions: (1) $R_D = r_{ON} || r_{OP}$, and (2) $g_m = g_{mN} + g_{mP}$. This is because M_1 and M_3 operate in parallel, and so do M_2 and M_4 . It follows from Eq. (5.19) that

$$\tau_{reg} = \frac{r_{ON} || r_{OP}}{(g_{mN} + g_{mP})(r_{ON} || r_{OP}) - 1} C_L. \quad (5.21)$$

The circuit can also be viewed as two cross-coupled inverters (why?).

Of course, we still do not know where the *input* is. But patience is a virtue.

5.4 Controlled Positive Feedback

The cross-coupled pair of Fig. 5.12 is attractive for comparator design as it can provide a high gain, producing a valid logical output from a small input difference. However, the circuit cannot be used in this simple form. To understand why, let us assume that (1) M_1 and M_2 have a threshold mismatch of ΔV_{TH} ; (2) the pair is otherwise symmetric; and (3) the circuit begins with $V_X = V_Y$. Does the loop remain in this state? Drawing the small-signal model as shown in Fig. 5.18,⁵ we observe that the drain currents of M_1 and M_2 are not equal even though V_X and V_Y are. If $\Delta V_{TH} > 0$, M_1 draws a greater current from C_{L1} , causing V_X to fall. As a result, I_{D2} drops, helping V_Y rise. The regeneration thus continues and the circuit rapidly amplifies its own offset, making a logical decision even in the absence of an imbalance between V_X and V_Y .

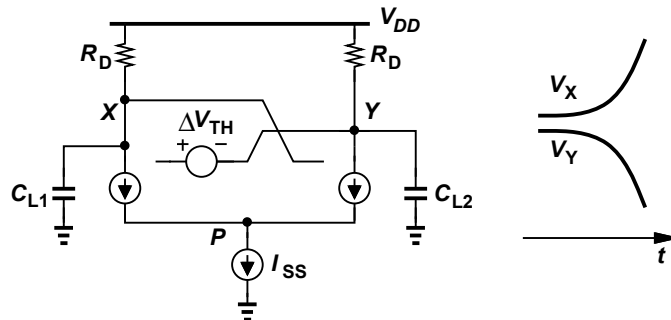


Figure 5.18 Cross-coupled pair with offset.

The foregoing study reveals that positive feedback is an “untamed” effect and must come into action only when necessary. For example, we can keep the tail current in Fig. 5.12 off while an imbalance is impressed on X and Y and then turn it on to amplify this imbalance. We will expand upon the notion of “controlled” positive feedback as we study various comparator topologies in this chapter.

⁵Note that the sources of M_1 and M_2 are *not* grounded because ΔV_{TH} makes the circuit asymmetric.

Example 5.8

Determine the output offset voltage, V_{OS} , of the circuit shown in Fig. 5.18, i.e., $V_X - V_Y$ due to ΔV_{TH} before regeneration begins.

Solution

Ignoring the capacitors, the reader can prove that

$$V_X - V_Y = \frac{g_m R_D}{g_m R_D - 1} \Delta V_{TH}. \quad (5.22)$$

Thus, if we impose an initial imbalance on C_{L1} and C_{L2} equal to the opposite of this value before the tail current turns on, the circuit remains balanced after regeneration begins.

However, if we first allow the circuit to amplify its own offset for some time and *then* attempt to apply an external imbalance, the necessary value is larger—as Eq. (5.18) predicts. We say the offset of the circuit *grows* with time.

5.5 A Simple Comparator

Suppose a differential stage provides outputs with a difference of 10 mV and a common-mode level of 0.8 V. We wish to construct a comparator that senses this difference and regeneratively delivers a differential output equal to 0.6 V.⁶ Our thoughts in the previous sections suggest that (1) the cross-coupled pair in Fig. 5.12 must remain off before we sense the input difference, and (2) we must ensure $g_m R_D > 1$ to obtain amplification. We then assemble the circuit shown in Fig. 5.19(a). Here, the main inputs reach X and Y through switches S_2 and S_1 , respectively. Also, these switches and the cross-coupled pair's tail device are controlled by a rail-to-rail clock, CK . Let us see how this comparator operates, assuming a threshold voltage of about +0.3 V for M_1 - M_3 and -0.3 V for S_1 and S_2 .

When CK is low, S_1 and S_2 are on and M_3 is off. As shown in Fig. 5.19(b), $V_X \approx V_{in2}$ and $V_Y \approx V_{in1}$ if the switch on-resistances are small. When CK rises, S_1 and S_2 turn off, M_3 turns on, and the cross-coupled pair comes into action. Now, the difference between V_X and V_Y is amplified until M_2 turns off and M_1 carries all of I_{SS} [Fig. 5.19(c)]. If we choose $R_D I_{SS} = 600$ mV, we satisfy the output swing requirement. The structure consisting of M_1 - M_3 and the load resistors is also called the “latch” as it stores the state.

The circuit of Fig. 5.19(a) is a realization of the clocked comparator concept shown in Fig. 5.5(b). We recognize that the output does *not* carry a valid logical level at all times. We say the circuit is in the “sense mode” when CK is low [before t_1 in Fig. 5.19(c)], and in the “regeneration mode” (or the “latch mode”) when CK is high (after t_1). Only at the end of the regeneration mode do we obtain a valid logical output—at least for the conditions that we have in this example.

Example 5.9

For the parameter values chosen in the circuit of Fig. 5.19(a), do V_X and V_Y vary symmetrically during regeneration? Assume $V_{DD} = 1$ V.

Solution

No, they do not. With $R_D I_{SS} = 0.6$ V, V_X falls from 0.795 mV to 0.4 V, whereas V_Y rises from 0.805 V to 1 V. The reader can repeat the regeneration analysis for this case.

⁶It is assumed that this swing can be interpreted as a valid logical level. If not, we can amplify it further.

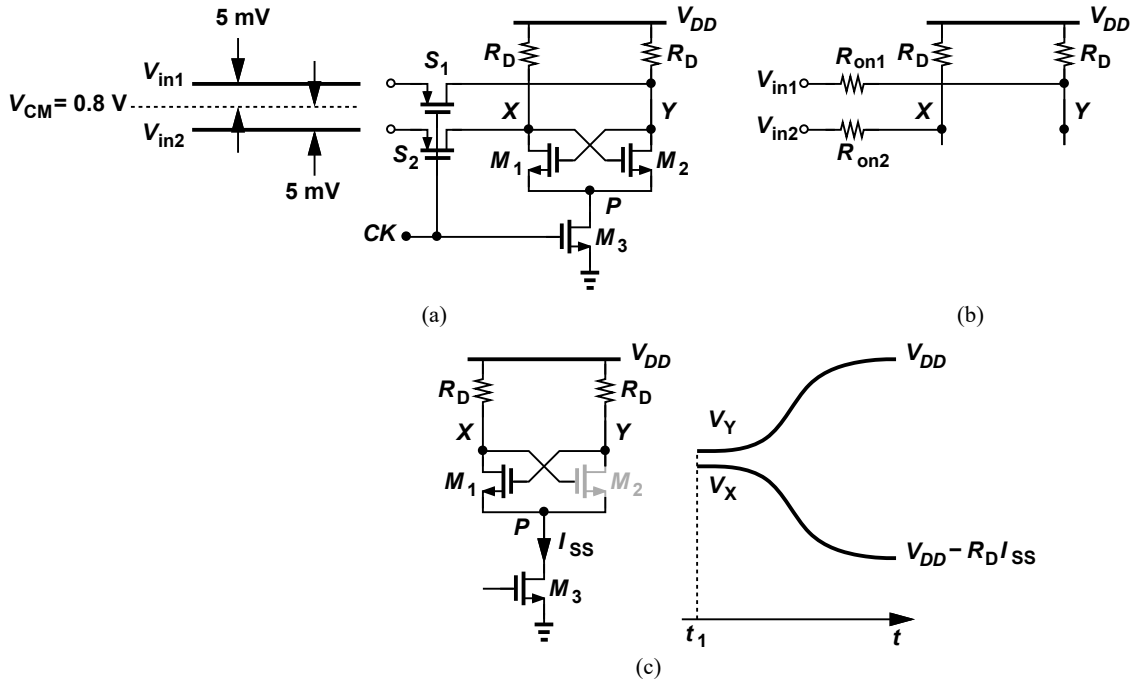


Figure 5.19 (a) Clocked cross-coupled pair with input sense switches, (b) equivalent input path, and (c) regeneration of outputs.

Role of S_1 and S_2 The reader may wonder why we have added S_1 and S_2 in series with the inputs in Fig. 5.19(a). Suppose they are replaced with wires. Then, the preceding stage interferes with the cross-coupled pair's operation. To understand this point, consider the situation depicted in Fig. 5.20(a), where S_1 and S_2 are omitted and the stage driving the comparator is modeled by output resistances R_1 and R_2 . We observe

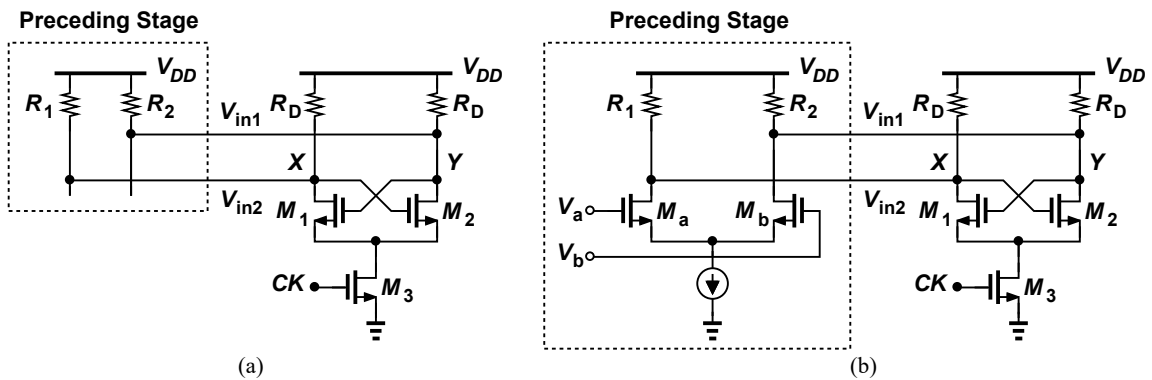


Figure 5.20 (a) Cross-coupled pair including output resistance of preceding stage, and (b) example of preceding stage.

that the regeneration is affected by these resistors as they appear in parallel with the R_D resistors. But a more serious phenomenon can also occur. Let us turn to Fig. 5.20(b). Suppose $V_a - V_b$ and hence $V_{in1} - V_{in2}$ have a certain value when CK goes high. The cross-coupled pair then attempts to amplify this initial difference. But what happens if, soon after CK rises, $V_a - V_b$ changes? Does the comparator make a correct decision? We do not readily know; nevertheless, we predict that an input change can interfere with the regeneration. Thus, we generally prefer to disconnect the input in the latch mode.

The simple comparator of Fig. 5.19(a) entails some drawbacks. First, the bias current of M_3 is poorly-defined and a function of PVT (why?). Second, to accommodate an output swing of 0.6 V, M_3 enters the triode region; we have, in Fig. 5.19(c), $V_P = V_{DD} - R_D I_{SS} - V_{DS1}$ while the gate of M_3 is at V_{DD} . As a result, the common-mode rejection of the circuit suffers. Third, the circuit provides no voltage gain before regeneration begins because M_1 and M_2 are off, a point that becomes more meaningful in the next section.

5.6 A Better Comparator

In the comparator of Fig. 5.19(a), the latch is disengaged from the input by means of simple passive switches, S_1 and S_2 . We can alternatively employ a switched differential pair for this purpose. Illustrated in Fig. 5.21(a), the idea is to *amplify* $V_{in1} - V_{in2}$ in the sense mode and apply the result to X and Y . Here, when CK is

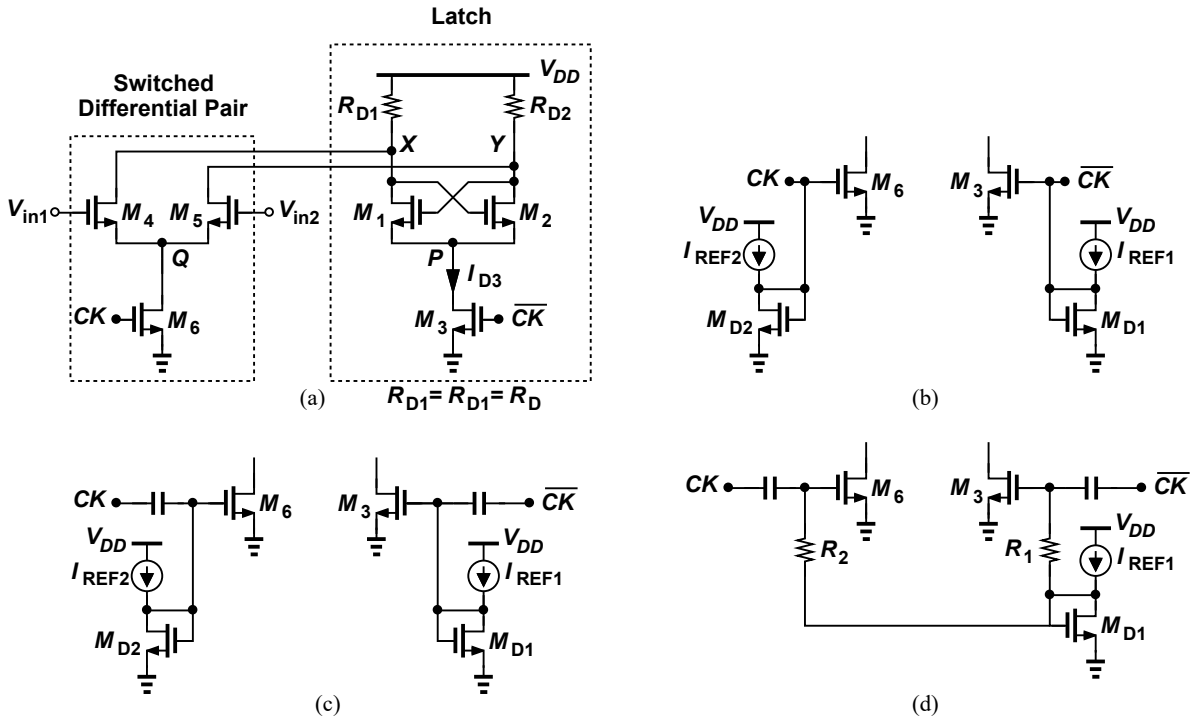


Figure 5.21 (a) Cross-coupled pair preceded by switched differential pair, (b) simple implementation of clock path, (c) proper biasing, and (d) sharing the bias path.

high, M_4 and M_5 are enabled, and M_1 and M_2 are disabled. The sense-mode voltage gain is thus equal to $g_{m4,5} R_D$ if channel-length modulation is neglected. In the transition to the regeneration mode, CK goes low, $\overline{CK}$ goes high, and the difference between V_X and V_Y is amplified by M_1 and M_2 . Note that the input pair and the cross-coupled pair share the same load resistors for their amplification function.

The topology of Fig. 5.21(a) is called a “current-steering” (or “current-mode”) comparator and finds application only at very high speeds because it consumes high power. But we study this structure for the intuition that it offers. We must address the other two drawbacks outlined in the previous section, namely, the PVT dependence of the bias current of M_3 (and M_6), and the triode-region operation of these devices. To resolve the former, we incorporate a current-mirror arrangement, as shown in Fig. 5.21(b), where the bias currents of M_3 and M_6 are copied from I_{REF1} and I_{REF2} , respectively. But how do we apply CK and $\overline{CK}$ without interfering with the bias network? This is accomplished by capacitor coupling [Fig. 5.21(c)]. In the absence of clock swings, M_3 and M_6 are still properly biased. When the clocks are applied, the gate voltages of these two transistors swing up and down, turning them on and off. Nevertheless, the coupling capacitors and the diode-connected transistors form a *high-pass* filter (why?), possibly attenuating the clock swings. For this

reason, we place relatively large resistors in series with M_{D1} and M_{D2} so as to lower the cut-off frequency. Finally, we merge the reference currents and the diode-connected device, arriving at the topology in Fig. 5.21(d).

In order to avoid driving M_3 and M_6 into the triode region, we opt for only moderate clock and output swings. For example, if the CK swing is $0.6 V_{pp}$ and V_P and V_Q do not fall below $0.6 V - V_{TH}$, then M_3 and M_6 remain in saturation.

The current-steering comparator of Fig. 5.21 entails three issues. First, the bias currents of M_3 and M_6 translate to static power consumption, a serious issue if a large number of comparators are used in an ADC. Second, the output swings are not rail-to-rail,⁷ requiring further amplification before they can drive rail-to-rail (CMOS) logic. Third, the coupling capacitors occupy a large area,⁸ again a severe difficulty if a large number of comparators are necessary. For these reasons, we rarely resort to this topology—unless very high speeds are targeted.

5.7 Comparator Design Parameters

Having studied basic comparator concepts and topologies, we are now ready to define the design parameters of comparators. We shall refer to the topology of Fig. 5.21(a) as an example.

5.7.1 Input Offset

As explained in Section 5.1, random asymmetries within a comparator shift its characteristic horizontally, an effect that can be modeled by inserting a voltage source in series with either of the comparator inputs. Called the input-referred offset or simply the input offset, this imperfection limits the accuracy with which a comparator can detect small differences.

In comparator design, we expend a great deal of effort on computing and managing the offset. We must therefore be able to identify sources of asymmetry and quantify their contributions. The following example serves as a starting point.

Example 5.10

In the differential pair shown in Fig. 5.22(a), M_4 and M_5 exhibit a threshold mismatch of ΔV_{TH} and R_{D1} and R_{D2} a mismatch of ΔR_D . That is, $R_{D1} = R_D$ and $R_{D2} = R_D + \Delta R_D$. Determine the input offset if $\lambda = 0$.

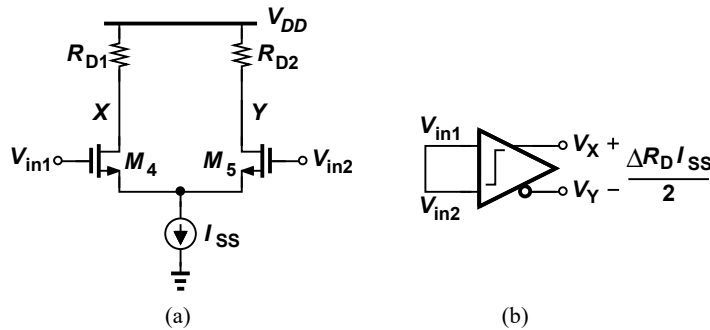


Figure 5.22 (a) Differential pair with resistor mismatch, and (b) output offset due to resistor mismatch.

Solution

Since the transistor and resistor mismatches are uncorrelated, we can calculate their contributions by superposition. With only ΔV_{TH} included, we simply have $V_{OS} = \Delta V_{TH}$. Next, we only consider ΔR_D , noting

⁷We state without proof that the moderate voltage swings here translate to high-speed operation.

⁸These capacitors should be about 5 to 10 times the gate capacitance of M_6 and M_3 if the clock swing attenuation is an issue.

that, for $V_{in1} - V_{in2} = 0$, we have $I_{D4} = I_{D5} = I_{SS}/2$ and hence $V_X - V_Y = (V_{DD} - R_{D1}I_{SS}/2) - (V_{DD} - R_{D2}I_{SS}/2) = \Delta R_D I_{SS}/2$. That is, the circuit displays an “output offset” equal to $\Delta R_D I_{SS}/2$ [Fig. 5.22(b)]. As explained in Section 5.1, we divide this quantity by the small-signal voltage gain, $g_m R_D$, to refer it to the input: $\Delta R_D I_{SS}/(2g_m R_D)$.

In the last step of our analysis, we must combine the offsets due to ΔV_{TH} and ΔR_D . Recall from basic statistics that the variance of the sum of two uncorrelated quantities, a and b , is given by the sum of their variances:

$$\sigma_{a+b}^2 = \sigma_a^2 + \sigma_b^2. \quad (5.23)$$

(This is similar to adding the *powers* of uncorrelated signals.) The overall input offset thus emerges as

$$V_{OS}^2 = \Delta V_{TH}^2 + \left(\frac{I_{SS}}{2g_m R_D} \right)^2 \Delta R_D^2. \quad (5.24)$$

It is instructive to compute the input-referred offset of the comparator shown in Fig. 5.21(a). We denote the threshold mismatch between M_1 and M_2 by ΔV_{TH1} and that between M_4 and M_5 by ΔV_{TH4} . We also assume a load resistor mismatch of ΔR_D . When the circuit is in the sense mode, the input differential pair amplifies $V_{in1} - V_{in2}$, and the circuit exhibits an input-referred offset given by Eq. (5.24):

$$V_{OS, sen}^2 = \Delta V_{TH4}^2 + \left(\frac{I_{SS}}{2g_{m4,5} R_D} \right)^2 \Delta R_D^2. \quad (5.25)$$

In the latch mode, $V_X - V_Y$ is applied to the regenerative pair. Recall from Example 5.8 that an imbalance of $V_X - V_Y = \Delta V_{TH1}$ is necessary to cancel the threshold mismatch between M_1 and M_2 . To refer this effect to the comparator input, we divide Eq. (5.22) by $g_{m4,5} R_D$. The total input offset, $V_{OS, tot}$, then consists of three uncorrelated terms, namely, ΔV_{TH4} , $I_{SS} \Delta R_D / (2g_{m4,5} R_D)$, and $[(g_{m1,2}/g_{m4,5}) / (g_{m1,2} R_D - 1)] \Delta V_{TH1}$. It follows that

$$V_{OS, tot}^2 = \Delta V_{TH4}^2 + \left(\frac{I_{SS}}{2g_{m4,5} R_D} \right)^2 \Delta R_D^2 + \left(\frac{g_{m1,2} \Delta V_{TH1}}{g_{m4,5} (g_{m1,2} R_D - 1)} \right)^2. \quad (5.26)$$

It is desirable to maximize $g_{m3} R_D$ so as to minimize the last two contributions.

Example 5.11

A student reasons that the resistor mismatch also manifests itself in the regeneration mode, contributing an additional term to Eq. (5.26). Is the student correct?

Solution

No, the student is not correct. Suppose only resistor mismatch is considered, and $V_{in1} - V_{in2} = 0$. As the circuit switches from the sense mode to the latch mode, $V_X - V_Y$ goes from $\Delta R_D I_{D6}/2$ to $\Delta R_D I_{D3}/2$ (why?). If $I_{D6} \approx I_{D3}$, then the output imbalance due to ΔR_D remains the same in the sense and latch modes.

Given that the sense and latch modes occur at different times, the reader may wonder why we add their offsets. This point is better understood by noting that the voltage difference inherited by the latch is already corrupted by the sense-mode offset.

5.7.2 Speed

A comparator used in an ADC must satisfy certain speed requirements. We wish to quantify the speed carefully.

Suppose a comparator is to operate at a clock rate of 1 GHz. We expect the circuit to deliver a correct decision in every clock cycle. Let us model the comparator of Fig. 5.21(a) as shown in Fig. 5.23(a), and apply a small difference between its inputs. When CK is high, the circuit is in the sense mode and $V_X - V_Y =$

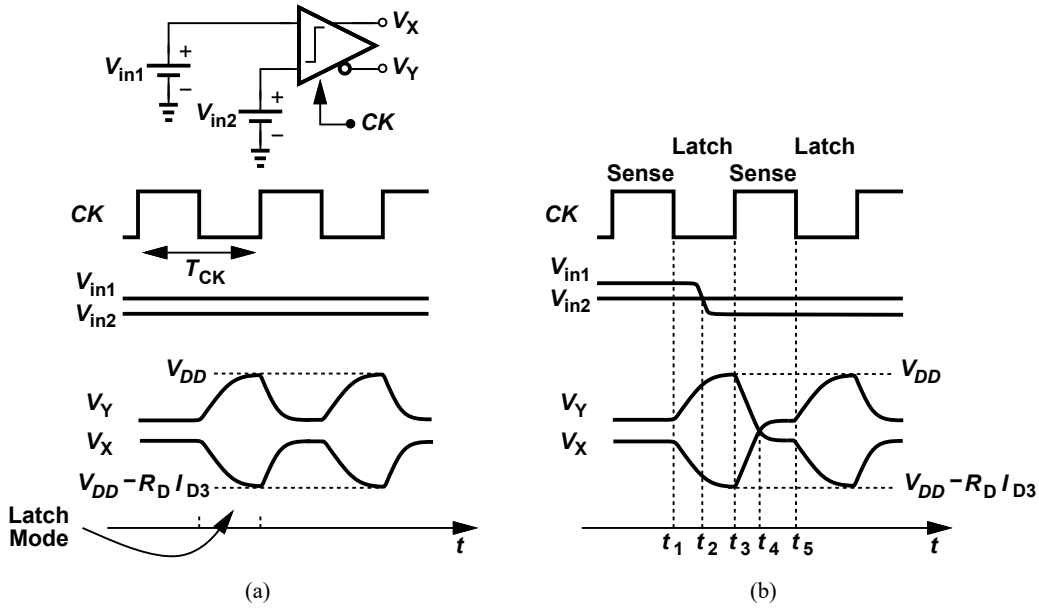


Figure 5.23 Comparator sensing (a) constant, or (b) time-varying inputs.

$-g_{m4,5}R_D(V_{in1} - V_{in2})$. When CK goes low, V_X and V_Y regeneratively reach their extreme values. We may then say that the comparator meets the 1-GHz speed requirement if V_X and V_Y can approach V_{DD} and $V_{DD} - R_D I_{D3}$ in about 500 ps.

The arrangement shown in Fig. 5.23(a) exemplifies a “static” test in that it assumes a *constant* input difference. In an ADC environment, on the other hand, this difference changes randomly with time, demanding a “dynamic” test. We illustrate this point with the aid of the waveforms shown in Fig. 5.23(b). Before $t = t_1$, the circuit is in the sense mode, $V_{in1} - V_{in2} > 0$, and hence $V_X - V_Y < 0$. At $t = t_1$, the latch turns on, and V_X and V_Y travel toward their extreme levels. Suppose, in this mode, at $t = t_2$, V_{in1} changes and $V_{in1} - V_{in2}$ becomes *negative*. Since the input differential pair is off, this change has no effect yet. At $t = t_3$, the circuit enters the sense mode again, which means $V_X - V_Y$ must change until it becomes equal to $-g_{m4,5}R_D(V_{in1} - V_{in2})$. Thus, V_X and V_Y begin from their extreme levels, cross each other at $t = t_4$, and approach their desired values, preferably before $t = t_5$. After this time, the comparator regenerates, taking V_X and V_Y to V_{DD} and $V_{DD} - R_D I_{D3}$, respectively.

Example 5.12

Suppose V_X and V_Y in Fig. 5.23(b) do not change fast enough between $t = t_3$ and $t = t_5$. Explain what happens.

Solution

As shown in Fig. 5.24, $V_X - V_Y$ has the wrong *polarity* at $t = t_5$, causing the regeneration to go in the wrong direction as well. That is, the comparator decision between $t = t_5$ and $t = t_6$ is incorrect.

These thoughts indicate that the dynamic test “stresses” the comparator to a greater extent, revealing its speed limitations more clearly. But how do we choose the value of $V_{in1} - V_{in2}$ in Fig. 5.23(b) before $t = t_1$ and after $t = t_2$? The input difference is typically chosen equal to 0.5 LSB of the ADC for a general assessment of the comparator’s speed. For a more severe test, we allow $V_{in1} - V_{in2}$ to be small and positive

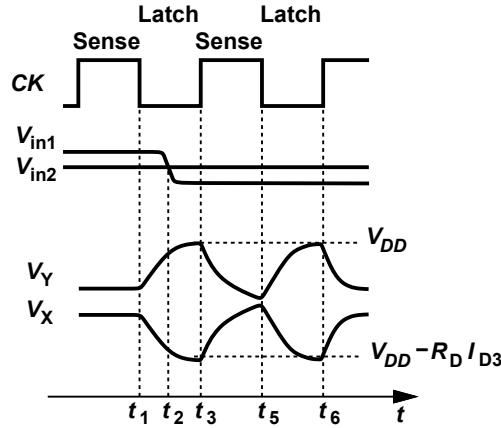


Figure 5.24 Incorrect comparator decision.

in one cycle and large and negative in the next. Since $V_{in1} - V_{in2}$ is *random*, it can sometimes be very small. The resulting comparator behavior must then be studied in the context of “metastability” (Section 6.2).

5.7.3 Electronic Noise

The devices comprising a comparator introduce thermal and flicker noise in the signal path, causing the circuit to occasionally make an incorrect decision if the input difference is small. In Fig. 5.21(a), for example, M_4 , M_5 , R_{D1} , and R_{D2} add noise to $V_X - V_Y$ in the sense mode, corrupting the desired output, $-g_{m4,5}R_D(V_{in1} - V_{in2})$. Recall from basic analog design that the differential output noise spectrum can be expressed as

$$\overline{V_{n,out}^2} = 2(4kTR_D + 4kT\gamma g_{m4,5}R_D^2). \quad (5.27)$$

This quantity represents the noise power in a 1-Hz bandwidth. The total noise corrupting the signal arises from all of the noise frequency components within the bandwidth available at the output nodes. We must therefore assume a *finite* output bandwidth, given by R_D and the total capacitance seen from X or Y to ground, C_L . As illustrated in Fig. 5.25, the noise expressed by (5.27) experiences the low-pass action of R_D

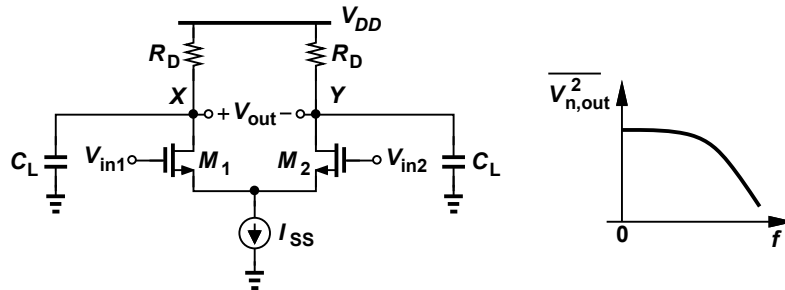


Figure 5.25 Differential pair output noise spectrum.

and C_L , and must be integrated from $f = 0$ to $f = \infty$ to yield the total noise. The reader can show that the integral is equal to $\overline{V_{n,out}^2}(\pi/2)/(2\pi R_D C_L) = 2(1 + \gamma g_{m4,5}R_D)kT/C_L$. The rms value of the noise is given by $\sqrt{\overline{V_{n,out}^2}(\pi/2)/(2\pi R_D C_L)}$.

What happens to the noise of M_4 and M_5 in Fig. 5.21(a) as the circuit enters the latch mode? These transistors turn off but their sense mode noise is sampled at X and Y along with the desired output. The sampling effect complicates the noise analysis but we rely on the total value found above as an approximation. In the latch mode, the noise of the cross-coupled pair further corrupts the desired $V_X - V_Y$. At the beginning

of regeneration, the noise of each can be modeled by a gate-referred voltage source whose spectrum is equal to $4kT\gamma/g_{m1,2}$. This resembles the threshold mismatch studied in Example 5.8, yielding an output noise voltage spectrum given by $(4kT\gamma/g_{m1,2})(g_{m1,2}R_D)^2/(g_{m1,2}R_D - 1)^2$. This quantity must be added to $V_{n,out}^2$ in (5.27). A more detailed analysis is presented in [1].

Having computed the total output noise, we take its square root to obtain the rms value and design the comparator such that this value is well below $g_{m3,4}R_D|V_{in1} - V_{in2}| = g_{m1,2}R_D \times 1 \text{ LSB}$. Alternatively, we divide the output rms noise by $g_{m3,4}R_D$ and select its value to be well below the ADC's LSB, e.g., around 0.25 LSB.

5.7.4 Kickback Noise

The switching actions within a comparator cause transient currents to flow through its input terminals. Called the “kickback noise,” such currents can interfere with the operation of the comparator or other circuits in an ADC. We begin our study with an example.

Example 5.13

The gate voltage of M_6 in Fig. 5.26 jumps from zero to a high value at $t = t_1$. Explain how the circuit behaves.

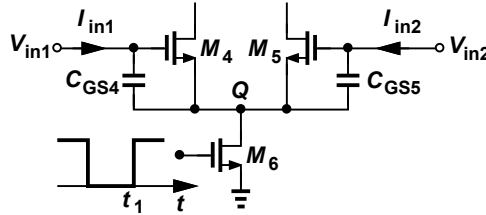


Figure 5.26 Kickback noise at input of switched pair.

Solution

Before $t = t_1$, all three transistors are off. When the gate voltage of M_6 rises, I_{D6} initially does *not* flow through M_4 and M_5 because they are off. Thus, this current must be provided by C_{GS4} and C_{GS5} until V_Q falls and M_4 and M_5 turn on. The inputs therefore experience transient currents. In other words, every time the comparator enters the sense mode, it draws “kickback” currents from V_{in1} and V_{in2} .

Why is the kickback noise undesirable? Suppose the comparator senses an analog signal at V_{in1} and a reference voltage at V_{in2} (Fig. 5.27). Even if I_{in1} and I_{in2} are equal, they flow through unequal impedances, introducing an error voltage at the comparator input.

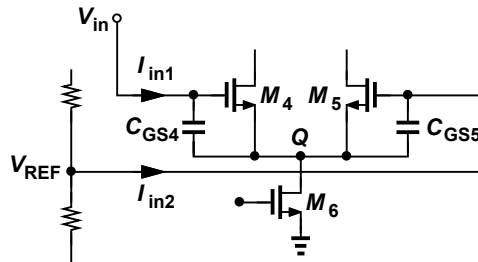


Figure 5.27 Effect of kickback noise.

We should make two additional remarks about kickback noise. First, the circuit depicted in Fig. 5.21(a) entails other kickback noise mechanisms as well. For example, when CK rises, it couples through C_{GD6} to node Q and then through C_{GS4} and C_{GS5} to the inputs. The same type of coupling occurs when CK falls. Second, to reduce kickback noise, we must precede the comparator with *unswitched* stages. As illustrated in Fig. 5.28, a continuously-biased differential pair, M_a - M_b , creates no kickback. The kickback from M_4 and M_5 must now propagate through C_{GDa} and C_{GDb} before reaching the main inputs, and hence is suppressed.

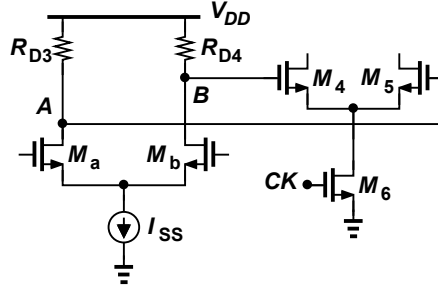


Figure 5.28 Kickback noise reduction.

Example 5.14

Does the additional differential pair in Fig. 5.28 also lower the input offset?

Solution

Yes, but not by a large factor. Denoting the voltage gain of this stage by A_1 , we recognize that Eq. (5.26) is now divided by A_1^2 . However, we must also remark that (a) in low-voltage design, and with channel-length modulation of M_a and M_b included, A_1 is in the range of 2 to 3, and (b) the new differential pair contributes its *own* offset. Specifically, mismatches between M_a and M_b and between R_{D3} and R_{D4} must also be taken into account.

5.8 The StrongArm Comparator

The most popular comparator topology in today's ADC design is based on the "StrongArm latch." Originally conceived to serve as a "sense amplifier" for memory design, this circuit offers a number of advantages over the configurations studied in previous sections. However, the structure and operation of the StrongArm comparator are more complex. We suggest that the reader study this section patiently.

5.8.1 Differential Pairs with Capacitive Loads

In previous sections, we have seen that a continuously-biased differential pair with resistive loads provides a voltage gain of $g_m R_D$. We wish to explore the possibility of replacing the resistors with capacitors. Consider the arrangement shown in Fig. 5.29(a), assuming the initial value of V_X and V_Y is equal to V_{DD} . Suppose V_{in1} is slightly higher than V_{in2} , allowing M_1 to carry a greater current than does M_2 . Also, assume V_{in1} and V_{in2} are constant. How does the circuit evolve? We observe that V_X falls faster than V_Y because $I_{D1} > I_{D2}$. Thus, $|V_X - V_Y|$ gradually increases with time. It is then possible that at some point in time, $t = t_1$, $|V_X - V_Y|/|V_{in1} - V_{in2}|$ is greater than unity, i.e., the circuit acts as an amplifier.

We compute the voltage gain at $t = t_1$ in Fig. 5.29(a) as follows. We note that, at $t = 0$, M_1 and M_2 are in saturation because their drain voltages are equal to V_{DD} , and V_{in1} and V_{in2} are less than V_{DD} . Let us assume that the transistors are still in saturation at $t = t_1$, i.e., $V_X > V_{in1} - V_{TH}$ and $V_Y > V_{in2} - V_{TH}$. Since $I_{D1} - I_{D2} = g_m(V_{in1} - V_{in2})$, and since this current difference is constant, the differential output voltage

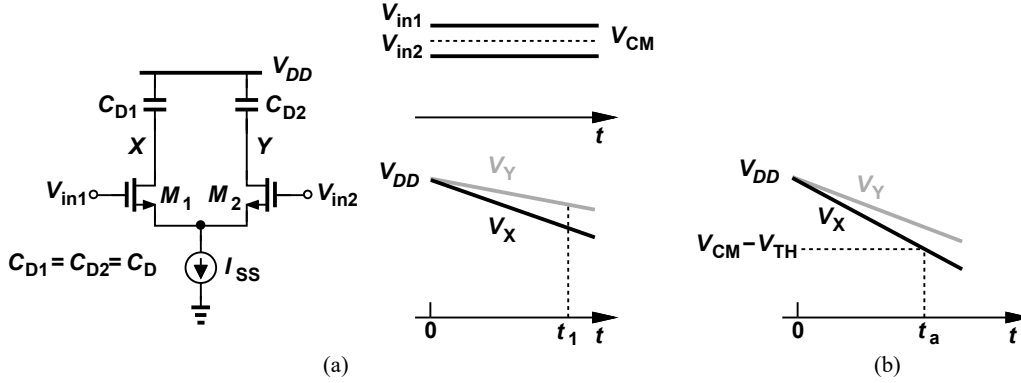


Figure 5.29 (a) Differential pair with capacitive loads, and (b) evolution of V_X and V_Y with time.

varies linearly with time:

$$V_X - V_Y = -g_m(V_{in1} - V_{in2})\frac{1}{C_D}t. \quad (5.28)$$

It follows that,

$$\frac{V_X - V_Y}{V_{in1} - V_{in2}} = -\frac{g_m}{C_D}t. \quad (5.29)$$

The waveforms in Fig. 5.29(a) reveal that, as time progresses, (a) a differential component appears between V_X and V_Y , and (b) V_X and V_Y also fall together, i.e., their common-mode level drops. The former is desirable but not the latter. If V_X or V_Y falls so much as to drive its corresponding transistor into the triode region, then g_m also drops, providing less gain. For this reason, we typically consider the circuit an “amplifier” only if M_1 and M_2 are in saturation. The voltage gain at the end of this regime is readily obtained if we assume V_{in1} and V_{in2} are close to each other and are approximately equal to a common-mode level, V_{CM} . When either V_X or V_Y falls to $V_{CM} - V_{TH}$ [Fig. 5.29(b)], one device is at the edge of saturation. From $t = 0$ to $t = t_a$, the current discharging C_{D1} is equal to $I_{SS}/2 + g_m(V_{in1} - V_{in2})/2$ (why?), which can be approximated as $I_{SS}/2$ if $V_{in1} - V_{in2}$ is small. Thus,

$$V_X(t_a) = V_{DD} - \frac{I_{SS}}{2C_D}t_a \quad (5.30)$$

$$= V_{CM} - V_{TH}. \quad (5.31)$$

It follows that,

$$t_a = \frac{2C_D}{I_{SS}}[V_{DD} - (V_{CM} - V_{TH})]. \quad (5.32)$$

The differential voltage gain at t_a is obtained from (5.29) as:

$$\frac{V_X - V_Y}{V_{in1} - V_{in2}} = \frac{-2g_m}{I_{SS}}[V_{DD} - (V_{CM} - V_{TH})], \quad (5.33)$$

a quantity independent of C_D . For M_1 and M_2 , we can write $g_m = I_{SS}/(V_{GS} - V_{TH})$, arriving at

$$\left| \frac{V_X - V_Y}{V_{in1} - V_{in2}} \right| = 2 \frac{V_{DD} - (V_{CM} - V_{TH})}{V_{GS} - V_{TH}}. \quad (5.34)$$

For example, if $V_{DD} = 1$ V, $V_{CM} = 0.6$ V, $V_{TH} = 0.3$ V, and $V_{GS} - V_{TH} = 0.2$ V, we obtain a gain of 7. Note, however, that we have neglected channel-length modulation here.

Example 5.15

What happens to V_X and V_Y in Fig. 5.29(a) at $t = \infty$?

Solution

As M_1 and M_2 draw current from their load capacitors, V_X and V_Y continue to fall, eventually reaching zero (Fig. 5.30). The tail current source enters the deep triode region, experiencing a zero V_{DS} and providing a zero drain current. The circuit is therefore completely “dead.”

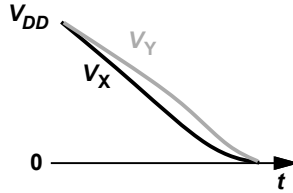


Figure 5.30 Output waveforms showing final values.

If V_X and V_Y collapse to zero, how can the circuit of Fig. 5.29(a) amplify again? We must reset (precharge) X and Y to V_{DD} before we expect amplification. Illustrated in Fig. 5.31, this point implies that the differential pair must have a “precharge mode” (when S_1 and S_2 are on) followed by an “amplification mode” (when they

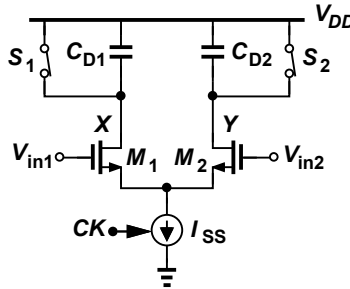


Figure 5.31 Addition of precharge.

are off). But what should we do with I_{SS} when X and Y are precharged? We can simply turn it off to save power.⁹ That is, when S_1 and S_2 are on, I_{SS} is off and vice versa. The reader should also recognize that C_{D1} and C_{D2} need not be tied to V_{DD} ; they can instead be connected to ground.

5.8.2 Addition of Positive Feedback

As explained in Section 5.4, we incorporate (controlled) positive feedback in comparators so as to produce logical levels at their output. In this spirit, let us add a regenerative PMOS pair to the circuit of Fig. 5.31, arriving at the structure shown in Fig. 5.32(a). Note that M_3 or M_4 turns on only if its gate voltage is less than $V_{DD} - |V_{THP}|$. In the precharge mode, S_1 and S_2 are on and $V_X = V_Y = V_{DD}$ while I_{SS} is off. In the amplification mode, S_1 and S_2 turn off, I_{SS} turns on, and V_X and V_Y begin to fall from V_{DD} ; but M_3 and M_4 are initially off. The circuit therefore behaves as that in Fig. 5.29(a), generating $V_X - V_Y = -(g_{mN}/C_D)t(V_{in1} - V_{in2})$, where g_{mN} denotes the transconductance of M_1 and M_2 . However,

⁹And also allow V_X and V_Y to precharge to exactly V_{DD} .

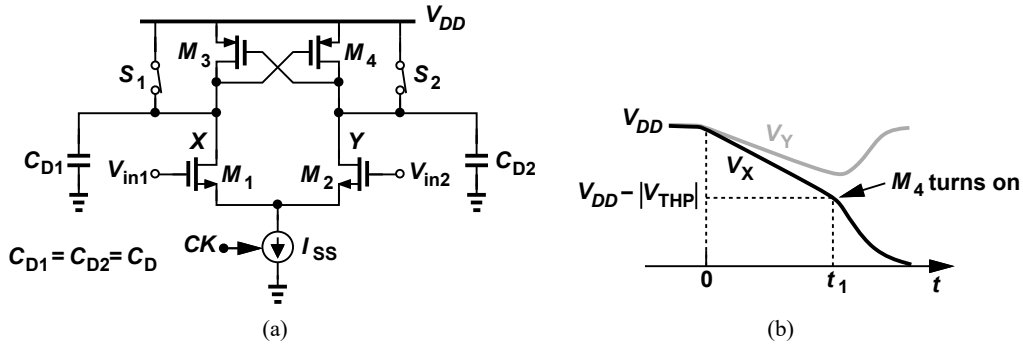


Figure 5.32 (a) Addition of positive feedback, and (b) corresponding waveforms.

as V_X drops down to $V_{DD} - |V_{THP}|$, M_4 turns on, *opposing* the fall in V_Y [Fig. 5.32(b)]. Thus, at some point, V_Y begins to *rise*, and $|V_X - V_Y|$ becomes substantially larger.

Example 5.16

Does M_3 turn on in Fig. 5.32(a)?

Solution

If the difference between V_X and V_Y is relatively large at $t = t_1$ in Fig. 5.32(b), then M_3 remains off because V_Y does not have a chance to drop to $V_{DD} - |V_{THP}|$. On the other hand, for a small difference, both M_3 and M_4 turn on, some regeneration occurs, and then M_4 prevails, pulling V_Y toward V_{DD} (Fig. 5.33) and turning M_3 off.

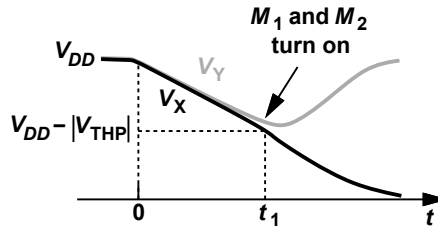


Figure 5.33 Output waveforms for a small initial difference.

Do we consider the structure in Fig. 5.32(a) a comparator? Yes, we do; the circuit does generate a large output in response to $V_{in1} - V_{in2}$. However, this topology's modes are different from those of the comparator in Fig. 5.21(a): the latter operates in the sense and latch modes whereas the former, in the precharge and amplification/latch modes. That is, the “sense” and “latch” functions occur while CK is high in Fig. 5.32(a).

The circuit of Fig. 5.32(a) displays a number of interesting properties. First, its positive-feedback action is automatically “controlled”: M_3 and M_4 do not begin regeneration until V_X or V_Y has fallen by $|V_{THP}|$. Second, the circuit can provide a voltage gain *before* M_3 and M_4 turn on, thereby suppressing the effect of their mismatches. To obtain the voltage gain at $t = t_1$ in Fig. 5.32(b), we return to Eq. (5.28) and write

$$V_X(t_1) = V_{DD} - \frac{I_{SS}}{2C_D} t_1 \quad (5.35)$$

$$= V_{DD} - |V_{THP}|. \quad (5.36)$$

That is,

$$t_1 = \frac{2C_D}{I_{SS}} |V_{THP}|. \quad (5.37)$$

Substituting in (5.29) yields

$$\frac{V_X - V_Y}{V_{in1} - V_{in2}}(t = t_1) = \frac{-2g_{mN}}{I_{SS}} |V_{THP}|. \quad (5.38)$$

Note that we have assumed M_1 and M_2 are in saturation. The overall input-referred offset is then equal to

$$V_{OS}^2 = \Delta V_{TH1,2}^2 + \frac{\Delta V_{TH3,4}^2}{\frac{4g_{mN}^2}{I_{SS}^2} |V_{THP}|^2}, \quad (5.39)$$

where $\Delta V_{TH1,2}$ and $\Delta V_{TH3,4}$ denote the NMOS and PMOS threshold mismatches, respectively. The reader can show that the mismatch between C_{D1} and C_{D2} also contributes offset.

Dynamic Offset In addition to the “static” offset computed above, the circuit of Fig. 5.32(a) also suffers from a “dynamic” component. Suppose V_X and V_Y have respectively reached zero and V_{DD} in the amplification mode. Now, the circuit enters the precharge mode but does not find enough time to pull V_X to exactly V_{DD} (Fig. 5.34). That is, the circuit inherits a difference of ΔV between V_Y and V_X as it goes to the next amplification mode. Called dynamic offset, this error adds to $\Delta V_{TH3,4}$.

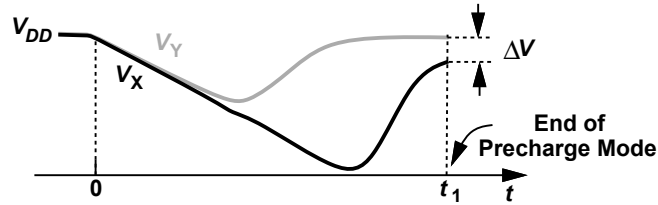


Figure 5.34 Problem of incomplete precharge.

We have seen that the offset of differential pairs is a function of their tail current. If the input CM level of the StrongArm topology changes with *time* (as occurs in some ADCs), then so does the offset.

Example 5.17

A student decides to replace the two precharge switches in Fig. 5.32(a) with an “equalizing” switch as shown in Fig. 5.35. Here, when the tail current is off, nodes X and Y are pulled together. Explain the pros and cons of this idea.

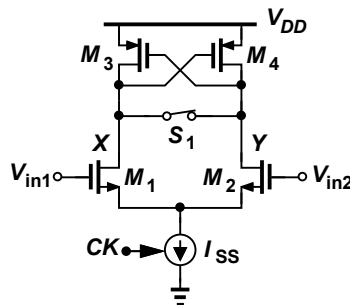


Figure 5.35 Use of an equalizing switch.

Solution

The equalizing switch offers two benefits. First, it suppresses the dynamic offset by ensuring $V_X = V_Y$. Second, when I_{SS} is off, V_X and V_Y remain around $V_{DD} - |V_{THP}|$ because M_3 and M_4 are nearly off and cannot pull these nodes to V_{DD} in a limited amount of time. Thus, in the next sense/latch mode, M_3 and M_4 readily come into action and begin to regenerate. That is, the circuit is slightly faster. However, V_X and V_Y no longer start from V_{DD} , and the initial voltage gain given by (5.38) is absent, accentuating the offset due to M_3 and M_4 . For this reason, we prefer the topology in Fig. 5.32(a).

The third property of the topology shown in Fig. 5.32(a) is that it draws static power in the amplification mode. This can be seen by noting that, when M_3 turns off, M_1 pulls V_X low but eventually carries no current. The circuit thus reduces to that in Fig. 5.36, where all of I_{SS} flows from the right side. Another drawback is that this current flow prohibits V_Y from reaching V_{DD} (why?), yielding a degraded logical level. The StrongArm topology resolves this issue, as explained next.

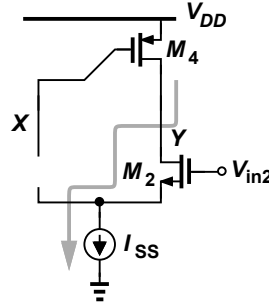


Figure 5.36 Problem of static current.

5.8.3 Complete StrongArm Comparator

We wish to modify the comparator of Fig. 5.32(a) such that it does not draw static power. To this end, we must cut off the path between M_4 and M_2 in Fig. 5.36 as the circuit makes a decision. Let us insert an NMOS device in this path and control its gate by V_X [Fig. 5.37(a)]. As V_X falls, this transistor turns off, forcing the current in the right branch to zero. We then apply the same idea to the left branch, obtaining the structure

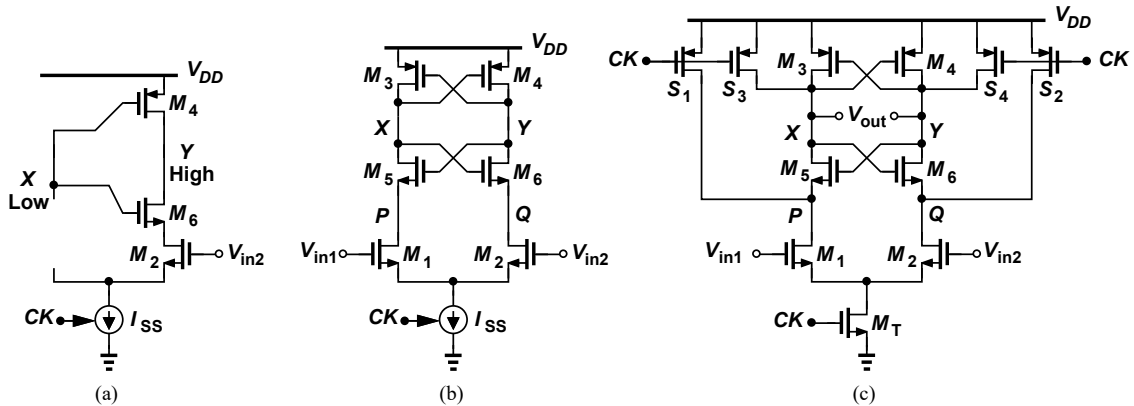


Figure 5.37 (a) Addition of M_6 to avoid static current, (b) StrongArm comparator core, and (c) addition of precharge switches.

shown in Fig. 5.37(b). The original StrongArm latch places a reset switch between P and Q [2, 3], but as

explained in Example 5.17, we prefer the modified topology in Fig. 5.37(c), where P and Q are precharged to V_{DD} so as to provide the initial voltage gain expressed by (5.38) [4].

Our incremental development of the StrongArm comparator has provided a reasonable understanding of its operation, but let us analyze the circuit carefully. As seen below, the operation consists of *four* modes. We assume V_{in1} is slightly greater than V_{in2} in Fig. 5.37(c).

In the first (precharge) mode, CK is low and nodes P , Q , X , and Y are tied to V_{DD} by their respective switches. We note that these switches must be realized by PMOS devices (why?) and are also controlled by CK . When CK goes high, the switches turn off, M_T begins to draw a current from M_1 and M_2 , and M_3 - M_6 are initially off. In this second mode, the circuit reduces to that shown in Fig. 5.38(a), with V_P and V_Q experiencing both a differential change and a common-mode change. Here, C_P and C_Q denote the total

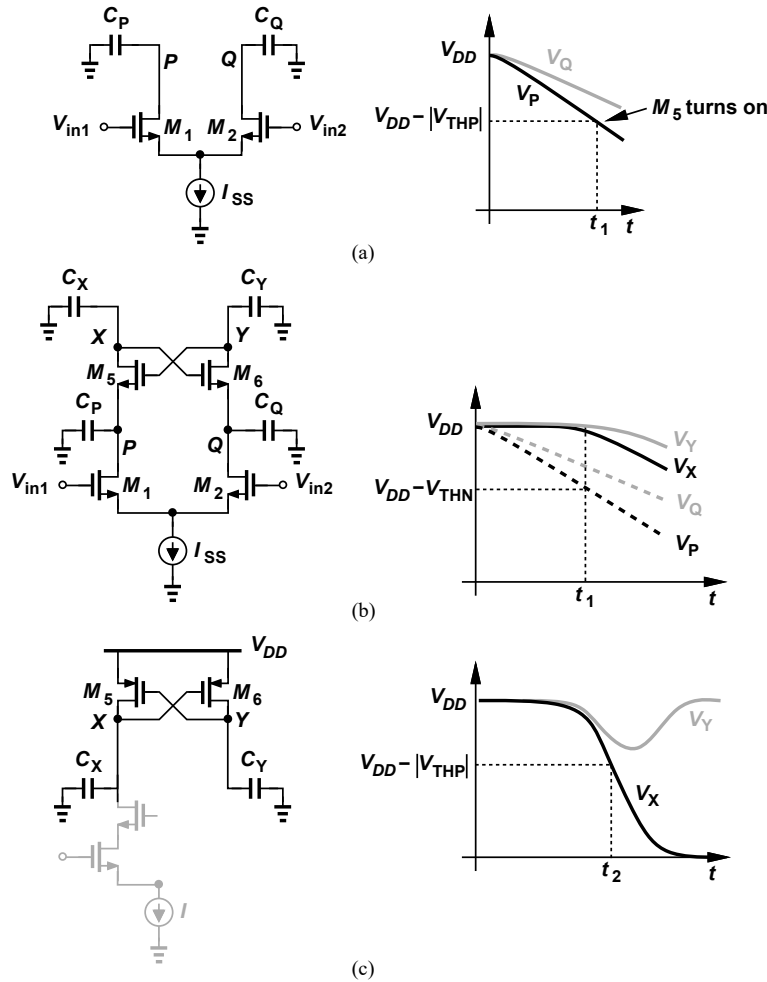


Figure 5.38 (a) StrongArm comparator in the initial amplification mode, (b) illustration of the third mode, and (c) positive feedback in the fourth mode.

capacitance at each node. But when do M_5 and M_6 turn on? Since their gate voltages are equal to V_{DD} , their sources must fall to $V_{DD} - V_{THN}$. The voltage gain provided by M_1 and M_2 at P and Q in this mode can be obtained from (5.38) by simply replacing $|V_{THP}|$ with V_{THN} :

$$\frac{V_P - V_Q}{V_{in1} - V_{in2}}(t = t_1) = \frac{-2g_{m1,2}}{I_{SS}} V_{THN}. \quad (5.40)$$

power because M_3 and M_6 or M_4 and M_5 cut the current path in either branch. Third, by virtue of the initial voltage gain provided by M_1 and M_2 , the input offset, V_{OS} , is dominated by only these two transistors. In a typical design, M_5 and M_6 contribute about 25%, and M_3 and M_4 about 10%, to V_{OS} , the latter because M_3 and M_4 turn on only after V_X and V_Y have fallen to $V_{DD} - |V_{THP}|$. Fourth, the comparator requires a single clock phase (as opposed to complementary phases).

The power consumption of the StrongArm comparator can be estimated from its node capacitances. Recall from logic design that a gate driving a capacitance C_1 at a frequency f_1 dissipates a power of $f_1 C_1 V_{DD}^2$. Of the four nodes in Fig. 5.37(c), three of them fall from V_{DD} to ground in the amplification/regeneration mode and return to V_{DD} in the precharge mode. Thus, $P \approx f_{CK}(C_X + C_P + C_Q)V_{DD}^2$.

Despite its low average power consumption, the StrongArm latch can draw large transient currents from the supply. This occurs at the beginning of the precharge mode, when *three* of the PMOS switches momentarily operate in the saturation region. As illustrated in Fig. 5.40, the three transistors sustain $|V_{DS}| = |V_{GS}| = V_{DD}$, drawing a high current even though they are relatively narrow devices.

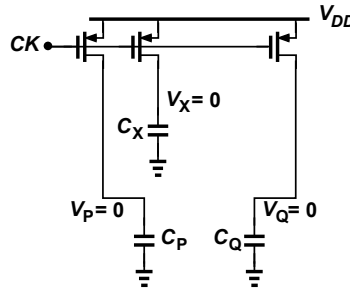


Figure 5.40 Illustration of supply transient current during precharge.

Example 5.20

A 4-bit ADC employs 16 StrongArm comparators. The capacitance at P , Q , X , and Y in Fig. 5.37(c) is about 5 fF. As shown in Fig. 5.41, the ADC is supplied from a voltage regulator. Since the regulator is not fast enough to provide charge during transitions, it is shunted by a bypass capacitor, C_b . Determine the minimum value of C_b that guarantees a drop of less than 10 mV on V_{DD} in the comparator precharge mode.

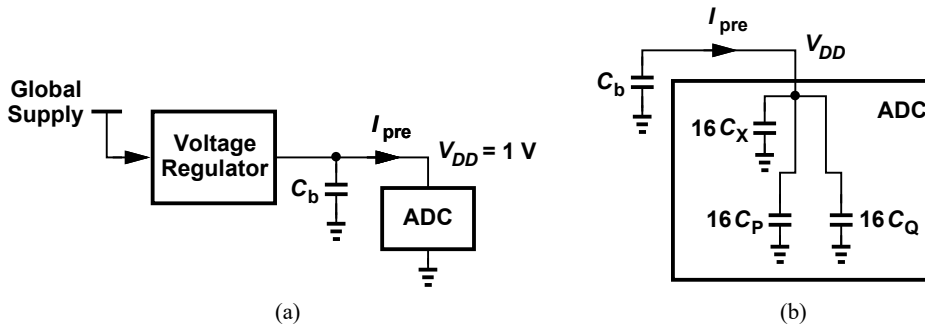


Figure 5.41 (a) Precharge current drawn from voltage regulator and bypass capacitor, and (b) simplified model.

Solution

We model the charge-sharing phenomenon as depicted in Fig. 5.41(b). Here, three capacitances within each comparator charge from zero to V_{DD} through C_b . For a total capacitance of $3 \times 5 \text{ fF} \times 16 = 240 \text{ fF}$, the necessary charge is equal to $240 \text{ fF} \times 1 \text{ V} = 240 \text{ fC}$. Capacitor C_b loses this charge while experiencing a voltage change of 10 mV. Thus, $C_b = 240 \text{ fC} / 10 \text{ mV} = 24 \text{ pF}$.

Owing to its rail-to-rail swings, the StrongArm comparator is slower than the current-steering topology of Fig. 5.21(a). This disadvantage manifests itself only in certain high-speed ADC architectures (Chapter 13).

The principal drawback of the StrongArm comparator is its high kickback noise. Recall from Section 5.7.4 that the switching action of the tail transistor creates input transient currents. In addition to coupling through gate-source capacitances, another significant component arises here as V_P and V_Q in Fig. 5.37(c) fall at unequal rates (because $V_{in1} \neq V_{in2}$) and couple to the inputs through C_{GD1} and C_{GD2} (Fig. 5.42). This effect becomes more pronounced as M_1 and M_2 enter the triode region and their gate-drain capacitances increase. As explained in Chapter 7, the input currents exhibit both differential and common-mode components.

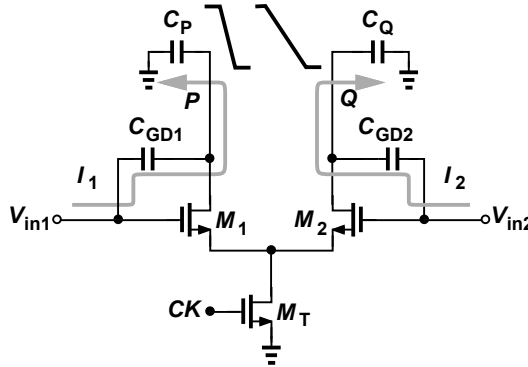


Figure 5.42 Illustration of kickback noise in StrongArm comparator.

In addition, the dependence of the tail current upon the input CM level degrades the speed if this level is not sufficiently high.

5.8.5 The Need for an RS Latch

In the precharge mode, both of the StrongArm comparator's outputs are high. How should the subsequent logic interpret this level? Since the circuit has not made a decision and this logical state is meaningless, the stage that follows must "ignore" the outputs in this mode. This is possible through the use of a reset-set (RS) latch. Let us first construct such a latch. Shown in Fig. 5.43(a) is an example; if R and S are low, M_r and M_s are off, and the cross-coupled inverters maintain a logical state, e.g., E is high and F is low indefinitely. If,

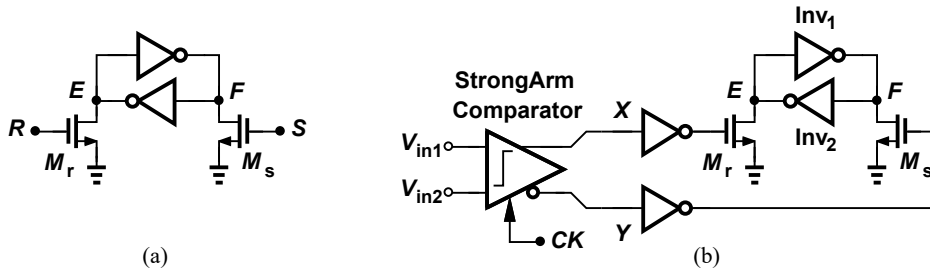


Figure 5.43 (a) An RS latch, and (b) its interface with StrongArm comparator.

for example, R goes high, E is "reset" to low and F goes high. That is, we write a new state onto the latch. The circuit preceding the latch must guarantee that R and S are not high simultaneously (why?).

We now wish to connect the RS latch to the StrongArm comparator. However, since the latter's outputs are both *high* during precharge, we must also insert two inverters between the two circuits, arriving at the arrangement in Fig. 5.43(b). The chain operates as follows. Assume $V_{in1} > V_{in2}$. During precharge, $V_X = V_Y = V_{DD}$, M_r and M_s are off, and Inv_1 and Inv_2 retain the logical state from the previous comparison, e.g., E is high and F is low. When CK goes high, V_X falls, M_r turns on, and E and F change

accordingly. The key point here is that E and F hold a correct logical state at all times—except during transitions—even though X and Y are not valid for half of the clock cycle.

We should remark that the foregoing RS latch incorporates “ratioed” logic, i.e., M_r and M_s must be strong enough to overcome the PMOS devices in Inv_2 and Inv_1 , respectively.¹⁰

References

- [1]. S. Ibrahim and B. Razavi, “Low-power CMOS equalizer design for 20-Gb/s systems,” *IEEE Journal of Solid-State Circuits*, vol. 46, pp. 1321-1336, June 2011.
- [2]. J. Montanaro et al, “A 160-MHz 32-b 0.5-W CMOS RISC microprocessor,” *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 1703-1714, Nov. 1996.
- [3]. T. Kobayashi et al, “A current-mode latch sense amplifier and a static power saving input buffer for low-power architecture,” *VLSI Circuits Symposium, Digest of Technical Papers*, pp. 28-29, June 1992.
- [4]. Y. T. Wang and B. Razavi, “An 8-bit 150-MHz CMOS A/D converter,” *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 308-317, March 2000.

¹⁰One could omit the inverters and use an RS latch with PMOS input devices. But, in such a structure, PMOS transistors must overcome the NMOS devices.

Problems

- 5.1. In the differential pair of Fig. 5.3(a), M_1 and M_2 are identical but $R_{D1} = R_D$ while $R_{D2} = R_D + \Delta R_D$. Determine the output and input offset voltages of the circuit.
- 5.2. From Fig. 5.8, determine what happens to the characteristic shown in Fig. 5.7 if the comparators exhibit random offsets.
- 5.3. Suppose in Fig. 5.12, $g_m R_D = 3$. We now double R_{D1} and R_{D2} . Does the circuit regenerate faster or more slowly?
- 5.4. Express the regeneration time constant of Fig. 5.12 if channel-length modulation is taken into account.
- 5.5. We double the widths of M_1 and M_2 in Fig. 5.16. If their transconductance is multiplied by $\sqrt{2}$, explain what happens to the regeneration time constant.
- 5.6. We double I_{SS} and the widths of M_1 and M_2 in Fig. 5.16. How does the regeneration time constant change? (Note that doubling the width and drain current of a MOS device always doubles its transconductance.)
- 5.7. As explained after Example 5.6, V_X and V_Y in Fig. 5.12 reach V_{DD} and $V_{DD} - R_D I_{SS}$, respectively, if the loop gain is greater than unity (so that the circuit regenerates and one transistor turns off). (a) Sketch V_X and V_Y as a function of time, noting that, inevitably, they begin to lose their exponential nature at some point. (b) From Eq. (5.19), explain what happens to τ_{reg} as one transistor turns off.
- 5.8. Suppose the circuit of Fig. 5.17 begins with $V_X \approx V_{DD}/2 + \Delta V$ and $V_Y \approx V_{DD}/2 - \Delta V$. Explain how V_X and V_Y vary with time and determine their final values. What are the final states of M_1 - M_4 ?
- 5.9. Compute the regeneration time constant of the circuit shown in Fig. 5.17 if channel-length modulation is neglected. How does τ_{reg} change if we double the transistor widths? (Hint: The bias currents and capacitances also double and the output resistances are halved.)
- 5.10. Show that, in Fig. 5.18, V_X rises and V_Y falls if the polarity of ΔV_{TH} is reversed.
- 5.11. Consider the latch of Fig. 5.19(b), assuming it begins with the initial conditions shown in Fig. 5.19(a). Using the small-signal model of Fig. 5.15, write the differential equations for V_X and V_Y and solve them with these initial conditions.
- 5.12. The current-mode comparator of Fig. 5.21(a) receives $V_{in1} - V_{in2} = \Delta V$ in the sense mode. Write an equation for the time evolution of $V_X - V_Y$ in the regeneration mode.
- 5.13. In the circuit of Fig. 5.22(a), we double the nominal value of R_{D1} and R_{D2} . It can be shown that, if the physical width of the resistors is not changed, their mismatch, ΔR_D , rises by a factor of $\sqrt{2}$. Explain how the input-referred offset changes.
- 5.14. Integrate the output noise of the circuit shown in Fig. 5.25 from $f = 0$ to $f = \infty$. You can simply model the circuit by a Thevenin voltage source having a flat noise spectrum of $V_{n,out}^2$, a Thevenin resistance of R_D , and a load capacitance of C_L .
- 5.15. If the input CM level of the circuit in Fig. 5.29 is equal to V_{DD} , how much voltage gain can the circuit provide before M_1 and M_2 enter the triode region?
- 5.16. Suppose switches S_1 and S_2 in Fig. 5.31 turn on at $t = 0$ with $V_X = V_Y = 0$, and precharge X and Y to V_{DD} . If S_1 and S_2 are realized by PMOS devices, sketch V_X and V_Y versus time and explain how these transistors' region of operation changes.
- 5.17. We simplify the circuit of Fig. 5.32(a) to that shown in Fig. 5.44. Assuming that at $t = 0$, $V_X = V_Y = V_a < V_{DD} - |V_{THP}|$. Use a small-signal model to write the differential equations for V_X and V_Y and express these voltages as a function of time. Neglect channel-length modulation.
- 5.18. In the circuit of Fig. 5.36, is it possible that the transistor acting as I_{SS} enters the triode region? Explain.
- 5.19. If we select a higher value for I_{SS} in Fig. 5.37(a), does the voltage gain in the initial amplification mode increase or decrease?
- 5.20. Simplifying the circuit of Fig. 5.38(b) to that shown in Fig. 5.45, write the differential equations for V_X and V_Y for small-signal operation. Assume that the initial conditions at P and Q are unequal but those at X and Y are equal. Next, solve the equations.
- 5.21. Suppose the StrongArm comparator of Fig. 5.37(c) has made a decision. In which region of operation does M_T reside?

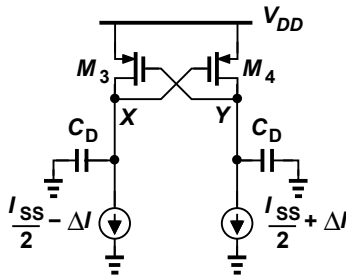


Figure 5.44 Regenerative pair.

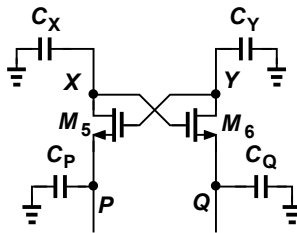


Figure 5.45 Another regenerative pair.

- 5.22.** The widths of all of the transistors in Fig. 5.37(c) are doubled. What happens to the power consumption of the circuit?
- 5.23.** In Fig. 5.43(b), $V_{in1} > V_{in2}$. Suppose CK rises at $t = 0$ and falls at $t = t_1$. Sketch V_X , V_Y , V_E , and V_F versus time.

Advanced Comparator Concepts

Having studied the basic design of comparators in the previous chapter, we now analyze a number of phenomena that impact their performance. Specifically, we deal with transistor mismatches, metastability, noise in the StrongArm topology, simulation of comparator noise, and offset cancelation. If eager to begin DAC or ADC studies, the reader can skip this chapter for now.

6.1 Analysis of Transistor Mismatches

In our study of offsets in Sections 5.1 and 5.7.1, we assumed that two nominally-identical MOSFETs exhibited a random threshold mismatch, ΔV_{TH} . In this section, we delve into the matter of transistor mismatches. We remind the reader from basic statistics that, if we add two uncorrelated Gaussian variables having standard deviations equal to σ_1 and σ_2 , then the standard deviation of the sum is given by

$$\sigma_{tot} = \sqrt{\sigma_1^2 + \sigma_2^2}. \quad (6.1)$$

For example, if two variables have the same standard deviation, σ , then that of their sum is $\sqrt{2}\sigma$.

6.1.1 Basic Concepts

Let us first identify all of the mismatches that can occur between two transistors. As shown in Fig. 6.1(a), the three transistor dimensions, W , L , and t_{ox} , incur mismatches. Moreover, the mobility, μ_n , may be slightly different. Also, the threshold voltages can have a mismatch: during fabrication, V_{TH} is adjusted by ion implantation [Fig. 6.1(b)], suffering from a mismatch if the number of ions received in one transistor's channel area is different from that in the other transistor. We now express the drain currents in the saturation region by a simple quadratic equation:

$$I_D = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})^2, \quad (6.2)$$

noting that mobility and dimension mismatches are reflected in $\mu_n C_{ox} W/L$. Also, recall that $C_{ox} = \epsilon_{ox}/t_{ox}$, where t_{ox} is the oxide thickness. In mismatch modeling, it is common practice to denote $\mu_n C_{ox} W/L$ by β and lump all of these four mismatches into one parameter, $\Delta\beta$. The transistor is thus characterized by only two mismatches, $\Delta\beta$ and ΔV_{TH} .

Example 6.1

Compute $\Delta\beta$ in terms of mobility and dimension mismatches.

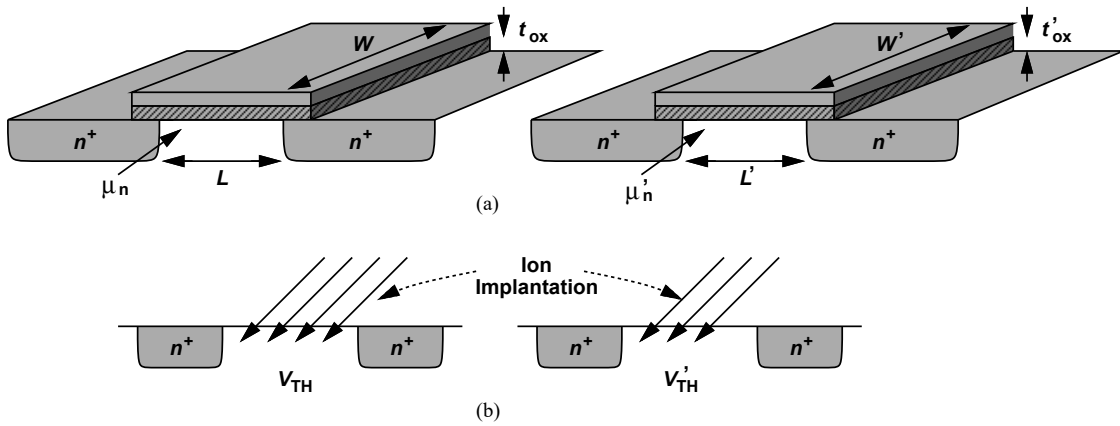


Figure 6.1 (a) Mismatch between two transistors, and (b) effect of implant mismatch.

Solution

Consider a function, y , of several variables, x_1 , x_2 , and x_3 . The total change in y (called the “total differential” in calculus) is given by

$$\Delta y = \Delta x_1 \frac{dy}{dx_1} + \Delta x_2 \frac{dy}{dx_2} + \Delta x_3 \frac{dy}{dx_3}. \quad (6.3)$$

This equation simply means that, to find Δy , we must scale Δx_j by the “sensitivity” of y to x_j and sum the results. Applying this principle to

$$\beta = \mu_n \frac{\epsilon_{ox}}{t_{ox}} \frac{W}{L}, \quad (6.4)$$

we have

$$\Delta \beta = \Delta \mu_n \frac{\epsilon_{ox}}{t_{ox}} \frac{W}{L} + \Delta t_{ox} \left(-\mu_n \frac{\epsilon_{ox}}{t_{ox}^2} \frac{W}{L} \right) + \Delta W \mu_n \frac{\epsilon_{ox}}{t_{ox}} \frac{1}{L} + \Delta L \left(-\mu_n \frac{\epsilon_{ox}}{t_{ox}} \frac{W}{L^2} \right). \quad (6.5)$$

We generally normalize $\Delta \beta$ to β to obtain the mismatch as a percentage:

$$\frac{\Delta \beta}{\beta} = \frac{\Delta \mu_n}{\mu_n} - \frac{\Delta t_{ox}}{t_{ox}} + \frac{\Delta W}{W} - \frac{\Delta L}{L}. \quad (6.6)$$

If the mismatches on the right-hand side are uncorrelated, we have from Eq. (6.1):

$$\left(\frac{\Delta \beta}{\beta} \right)^2 = \left(\frac{\Delta \mu_n}{\mu_n} \right)^2 + \left(\frac{\Delta t_{ox}}{t_{ox}} \right)^2 + \left(\frac{\Delta W}{W} \right)^2 + \left(\frac{\Delta L}{L} \right)^2. \quad (6.7)$$

We conclude that $\Delta \beta / \beta$ can be minimized by maximizing W and L . We examine this point more closely later and discover that the situation is more complex.

Example 6.2

Determine the input offset voltage, $V_{OS,in}$, of the differential pair shown in Fig. 6.2 in terms of β mismatch and threshold mismatch between its input transistors.

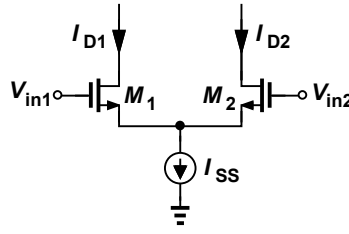


Figure 6.2 Differential pair for mismatch study.

Solution

We must obtain the output mismatch, $\Delta I_D = I_{D1} - I_{D2}$, while the input difference is zero, and then divide ΔI_D by the “gain” to refer it to the input. We have the following equations:

$$I_{D1} - I_{D2} = \Delta I_D, \quad (6.8)$$

$$I_{D1} + I_{D2} = I_{SS}, \quad (6.9)$$

$$V_{GS1} = V_{GS2}, \quad (6.10)$$

the last one because $V_{in1} = V_{in2}$. Let us assume $\beta_1 = \beta$, $\beta_2 = \beta + \Delta\beta$, $V_{TH1} = V_{TH}$, and $V_{TH2} = V_{TH} + \Delta V_{TH}$. Denoting V_{GS1} and V_{GS2} by V_{GS} , we express I_{D2} as

$$I_{D2} = \frac{1}{2}\beta_2(V_{GS} - V_{TH2})^2 \quad (6.11)$$

$$= \frac{1}{2}(\beta + \Delta\beta)(V_{GS} - V_{TH} - \Delta V_{TH})^2 \quad (6.12)$$

$$\approx \frac{1}{2}(\beta + \Delta\beta)(V_{GS} - V_{TH})^2 - \beta(V_{GS} - V_{TH})\Delta V_{TH}, \quad (6.13)$$

where the terms involving $\Delta\beta \cdot \Delta V_{TH}$ and ΔV_{TH}^2 have been neglected. Since $I_{D1} = (1/2)\beta(V_{GS} - V_{TH})^2$,

$$\Delta I_D = -\frac{\Delta\beta}{2}(V_{GS} - V_{TH})^2 + \beta(V_{GS} - V_{TH})\Delta V_{TH}. \quad (6.14)$$

Dividing ΔI_D by the gain, we refer the mismatch to the input. The gain is given by the differential pair’s transconductance (why?), $G_m \approx g_{m1} \approx g_{m2} \approx I_{SS}/(V_{GS} - V_{TH})$. We thus have

$$V_{OS,in} = \frac{\Delta I_D}{g_{m1}} \quad (6.15)$$

$$= \frac{-(\Delta\beta/2)(V_{GS} - V_{TH})^2 + \beta(V_{GS} - V_{TH})\Delta V_{TH}}{I_{SS}/(V_{GS} - V_{TH})}. \quad (6.16)$$

We also write $I_{D1} \approx I_{SS}/2$ and hence $\beta(V_{GS} - V_{TH})^2 \approx I_{SS}$. It follows that

$$V_{OS,in} = -\frac{\Delta\beta}{2\beta}(V_{GS} - V_{TH}) + \Delta V_{TH}. \quad (6.17)$$

Since $\Delta\beta$ and ΔV_{TH} are uncorrelated,

$$V_{OS,in}^2 = \frac{1}{4} \left(\frac{\Delta\beta}{\beta} \right)^2 (V_{GS} - V_{TH})^2 + \Delta V_{TH}^2. \quad (6.18)$$

Equation (6.18) merits two remarks. First, to reduce the input offset due to β mismatch, the differential pair's overdrive voltage must be minimized—by selecting wide transistors or a small bias current (why?). Second, this result shows why we generally normalize $\Delta\beta$ but not ΔV_{TH} .

6.1.2 Dependence of Mismatches on Channel Area

Recall from Eq. (6.7) that $\Delta\beta/\beta$ can be minimized by maximizing W and L . In reality, increasing the transistors' channel area, WL , provides greater benefits than this equation suggests.

Let us begin with length mismatch, neglecting other mismatches for now. To develop an intuitive understanding, we first consider only *one* transistor and examine random variations in its length. As shown in Fig. 6.3(a), random aberrations (“roughness”) on the gate edges cause the length to vary from one transistor to

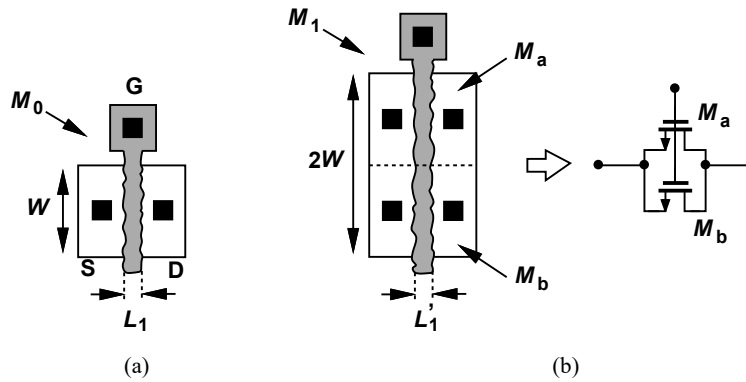


Figure 6.3 (a) Effect of edge “roughness,” and (b) reduction of mismatch due to a larger gate area.

another. What happens to this length variation if we double the *width* of the device [Fig. 6.3(b)]? Viewing M_1 as composed of units M_a and M_b , we intuitively note that random changes in the length of M_a and random changes in the length of M_b tend to average out to some extent. That is, the variation of the length of M_1 is *less* than that of M_0 . The statistical averaging occurring here suggests that the length variation falls by a factor of $\sqrt{2}$.

The foregoing prediction can also be supported by calculations. Considering M_a and M_b as two transistors in parallel (why?), we recognize that the factor $2W/L_1$ in transistor equations can be replaced with $W/L_a + W/L_b$. For example, the drain current in the saturation region is expressed as

$$I_{D1} = \frac{1}{2} \mu_n C_{ox} \left(\frac{W}{L_a} + \frac{W}{L_b} \right) (V_{GS} - V_{TH})^2, \quad (6.19)$$

where other variations are neglected. Similarly, the triode-region and on-resistance equations contain the factor $W/L_a + W/L_b$. Now, suppose the length of each unit transistor experiences random changes around a nominal (mean) value, L_0 . That is, $L_a = L_0 + \delta L_a$ and $L_b = L_0 + \delta L_b$, where random changes δL_a and δL_b are assumed uncorrelated Gaussian variables with a standard deviation of σ . We write

$$\frac{W}{L_a} + \frac{W}{L_b} = \frac{W}{L_0 \left(1 + \frac{\delta L_a}{L_0} \right)} + \frac{W}{L_0 \left(1 + \frac{\delta L_b}{L_0} \right)} \quad (6.20)$$

$$= \frac{W}{L_0} \frac{2 + \frac{\delta L_a}{L_0} + \frac{\delta L_b}{L_0}}{\left(1 + \frac{\delta L_a}{L_0} \right) \left(1 + \frac{\delta L_b}{L_0} \right)}. \quad (6.21)$$

Since δL_a and δL_b are much less than L_0 , the denominator of (6.21) reduces to $1 + \delta L_a/L_0 + \delta L_b/L_0$. Also, for $\epsilon \ll 1$, we have $(2 + \epsilon)/(1 + \epsilon) \approx (2 + \epsilon)(1 - \epsilon) \approx 2 - \epsilon$. That is,

$$\frac{W}{L_a} + \frac{W}{L_b} \approx \frac{W}{L_0} \left(2 - \frac{\delta L_a + \delta L_b}{L_0} \right). \quad (6.22)$$

Writing $1 - \epsilon \approx 1/(1 + \epsilon)$, we have

$$\frac{W}{L_a} + \frac{W}{L_b} \approx \frac{2W}{L_0 + \frac{\delta L_a + \delta L_b}{2}}. \quad (6.23)$$

As expected, M_1 has a nominal aspect ratio of $2W/L_0$. We can consider the denominator of (6.23) as the equivalent length of M_1 , recognizing that it has a variability given by $(\delta L_a + \delta L_b)/2$. If δL_a and δL_b follow a Gaussian distribution with a standard deviation of σ , then the standard deviation of $(\delta L_a + \delta L_b)/2$ is given by $\sqrt{2}\sigma/2 = \sigma/\sqrt{2}$. Thus, the overall *length* variation is reduced by a factor of $\sqrt{2}$ as a result of doubling the *width* of the transistor. In Problem 6.3, we show that doubling the length of a transistor reduces the width variability by a factor of $\sqrt{2}$.

In summary, we have observed that increasing one dimension of the transistor decreases the variability of the other dimension [1], an interesting departure from what Eq. (6.7) predicted. In the following example, we examine the effect on the threshold variation.

Example 6.3

The width of a transistor is doubled. How does the threshold standard deviation change?

Solution

Returning to Fig. 6.3(b), we postulate that the threshold variations of M_a and M_b experience some averaging. As an example, let us consider the saturation region and write:

$$I_{Da} + I_{Db} = \frac{1}{2}\mu_n C_{ox} \frac{W}{L} [(V_{GS} - V_{THa})^2 + (V_{GS} - V_{THb})^2]. \quad (6.24)$$

Expressing the thresholds as a nominal (mean) value plus a variation, we have $V_{THa} = V_{TH0} + \delta V_a$ and $V_{THb} = V_{TH0} + \delta V_b$. The standard deviation of δV_a and δV_b is denoted by σ_{VTH} . We expand the right-hand side of (6.24) and neglect δV_a^2 and δV_b^2 :

$$I_{Da} + I_{Db} \approx \frac{1}{2}\mu_n C_{ox} \frac{W}{L} [2(V_{GS} - V_{TH0})^2 - 2(V_{GS} - V_{TH0})(\delta V_a + \delta V_b)]. \quad (6.25)$$

Factoring out the coefficient 2 and adding the negligible term $(\delta V_a + \delta V_b)^2/4$ inside the square brackets, we have

$$I_{Da} + I_{Db} \approx \frac{1}{2}\mu_n C_{ox} \frac{2W}{L} \left[V_{GS} - V_{TH0} - \frac{\delta V_a + \delta V_b}{2} \right]^2. \quad (6.26)$$

The overall transistor thus exhibits a threshold standard deviation equal to $\sqrt{2}\sigma_{VTH}/2 = \sigma_{VTH}/\sqrt{2}$. In Problem 6.4, we prove that doubling the length has the same effect. That is, the threshold variation falls as the gate *area* of the transistor increases.

Our study points to a profound result: variations of the transistor parameters, including μ_n and C_{ox} , tend to average out as the channel area increases. This point is elegantly articulated by two simple mismatch

equations [1]:

$$\frac{\Delta\beta}{\beta} = \frac{A_\beta}{\sqrt{WL}}, \quad (6.27)$$

$$\Delta V_{TH} = \frac{A_{VTH}}{\sqrt{WL}}. \quad (6.28)$$

Here, A_β and A_{VTH} are relatively constant for a given CMOS process. For example, $A_\beta \approx 0.05 \mu\text{m}$ and $A_{VTH} \approx 2.5 \text{ mV} \cdot \mu\text{m}$ in typical 28-nm technologies. These two expressions represent another departure from Eq. (6.7).

Example 6.4

A differential pair employs $W/L = 2 \mu\text{m}/28 \text{ nm}$ for its input transistors with an overdrive voltage of 200 mV. Determine the input-referred offset voltage, $V_{OS,in}$. How does $V_{OS,in}$ change if W is doubled while I_D is constant?

Solution

From (6.27) and (6.28), we have $\Delta\beta/\beta = 21\%$ and $\Delta V_{TH} = 10.6 \text{ mV}$. It follows from (6.18) that

$$V_{OS,in}^2 = \frac{1}{4} \left(\frac{\Delta\beta}{\beta} \right)^2 (V_{GS} - V_{TH})^2 + \Delta V_{TH}^2 \quad (6.29)$$

$$= \frac{(42 \text{ mV})^2}{4} + (10.6 \text{ mV})^2 \quad (6.30)$$

$$\approx (23.5 \text{ mV})^2. \quad (6.31)$$

If W is doubled, then both $\Delta\beta/\beta$ and ΔV_{TH} fall by a factor of $\sqrt{2}$, and so does $V_{GS} - V_{TH}$ (for I_D to be constant). Thus, $V_{OS,in}$ drops to 13 mV.

6.2 Metastability

Recall from Section 5.3 that a regenerative comparator can provide an infinite gain if given infinite time. For a finite clock period, on the other hand, the regeneration may not yield an unambiguous logical output if the input difference is small. Thus, the subsequent logical circuit may fail to interpret this output correctly. We say the comparator has become “metastable” in this case.

6.2.1 Metastability in Current-Steering Comparators

Suppose the current-steering topology of Fig. 5.21(a) is employed as shown in Fig. 6.4, where V_{in} is a random signal swinging around V_{REF} . When the comparator is clocked at t_1 , the amplified input difference is large enough to produce a well-defined logical level between V_X and V_Y by t_a . However, clocked at t_2 , the comparator senses a small difference and is unable to amplify $V_X - V_Y$ sufficiently by t_b . We say the circuit has become metastable after t_2 .

Before formulating this phenomenon, we should remark that metastability is inevitable: since V_{in} in Fig. 6.4 is random, it can assume a value that is arbitrarily close to V_{REF} . Thus, metastable states occur with a finite probability.

Let us return to the regenerative current-steering pair and study its behavior for very small initial differences [Fig. 6.5(a)]. We examine the response for two cases, namely, when the initial difference is V_{XY0} or

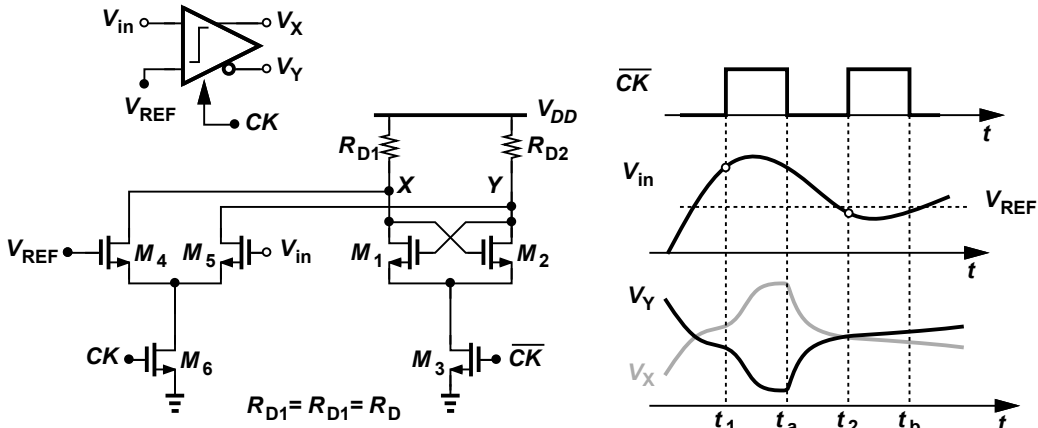


Figure 6.4 Metastability in a current-steering comparator.

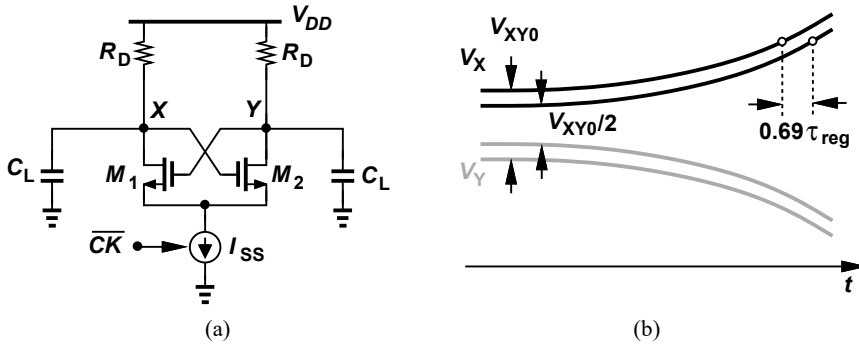


Figure 6.5 (a) Simple regenerative pair, and (b) metastability response for different initial conditions.

$V_{XY0}/2$. From Chapter 5, the former is expressed as

$$V_{XY1}(t) = V_{XY0} \exp \frac{t}{\tau_{reg}}, \quad (6.32)$$

and the latter as

$$V_{XY2}(t) = (V_{XY0}/2) \exp \frac{t}{\tau_{reg}}. \quad (6.33)$$

Noting that $1/2 \approx \exp(-0.69)$, we rewrite this response as

$$V_{XY2}(t) = V_{XY0} \exp \frac{t - 0.69\tau_{reg}}{\tau_{reg}}. \quad (6.34)$$

Interestingly, V_{XY2} is simply a shifted replica of V_{XY1} [Fig. 6.5(b)]. That is, every halving of the initial difference shifts the response to the right by an amount equal to $0.69\tau_{reg}$. We can also say the waveform is delayed by $0.69\tau_{reg}$, but the growth rate does not change.

Example 6.5

Explain how we can measure the regeneration time constant in simulations.

Solution

The waveforms in Fig. 6.5(b) suggest that the time shift of $0.69\tau_{reg}$ between V_{XY1} and V_{XY2} can be readily measured to yield τ_{reg} . Alternatively, we can take the logarithm (in base 10) of the two sides and write

$$\log V_{XY1}(t) = \log V_{XY0} + \frac{t}{\tau_{reg}} \log e, \quad (6.35)$$

concluding that τ_{reg} can be obtained as the inverse slope of the response on a log scale.

Example 6.6

We wish the circuit of Fig. 6.5(a) to provide a differential output equal to $0.9I_{SS}R_D$ in less than T_m seconds. How much is the minimum V_{XY0} that the circuit can sense?

Solution

We have

$$0.9I_{SS}R_D = V_{XY0} \exp \frac{T_m}{\tau_{reg}} \quad (6.36)$$

and hence

$$V_{XY0} = 0.9I_{SS}R_D \exp \frac{-T_m}{\tau_{reg}}. \quad (6.37)$$

For example, if $I_{SS}R_D = 300$ mV, $T_m = 50$ ps, and $\tau_{reg} = 5$ ps, then $V_{XY0} = 0.012$ mV. The exponential dependence on T_m and τ_{reg} underscores their critical role in the design.

We must now make an important observation. Since the input difference in Fig. 6.4 and hence V_{XY0} in Fig. 6.5(b) can be arbitrarily small, the regenerated output occasionally fails to reach a valid logical state. Suppose $|V_{in} - V_{REF}|$ in Fig. 6.4 is 0.1 mV and the LSB of the ADC employing the comparator is 1 mV. In this situation, the ADC does not “care” whether the comparator generates a ONE or a ZERO, i.e., the input difference is small enough that either decision is acceptable. However, it is the “indecisiveness” of the comparator that leads to errors: the comparator output is “indeterminate,” potentially causing the subsequent logic to fail. The nature of this failure is described in Chapter 13.

How do we quantify metastability? We proceed in three steps. First, we must define exactly what a metastable state is. This is somewhat arbitrary; we define such a state as when the comparator output reaches less than 90% of its final value in the time allocated to it (Example 6.6). Second, we compute the input difference sensed by the comparator that yields such a state and call it the comparator “sensitivity,” V_{sen} . For the current-steering topology of Fig. 6.5(a), we must divide V_{XY0} in (6.37) by the input pair’s voltage gain, $g_m R_D$, obtaining $V_{sen} = (0.9I_{SS}/g_m) \exp(-T_m/\tau_{reg})$. Thus, if $|V_{in} - V_{REF}|$ in Fig. 6.4 is below this value, the output is metastable.

Third, we must determine how *often* metastable states occur, and hence compute the “error rate.” Let us return to Fig. 6.4 and assume that V_{in} randomly swings between $V_{REF} - \Delta V$ and $V_{REF} + \Delta V$ [Fig. 6.6(a)]. For example, the signal sensed by an ADC may vary across this range. As a simple case, suppose V_{in} has a uniform distribution (probability density function), $f(V_{in})$, around V_{REF} [Fig. 6.6(b)]. Thus, the probability that $|V_{in} - V_{REF}| \leq V_{sen} = (0.9I_{SS}/g_m) \exp(-T_m/\tau_{reg})$ is equal to the shaded area and given

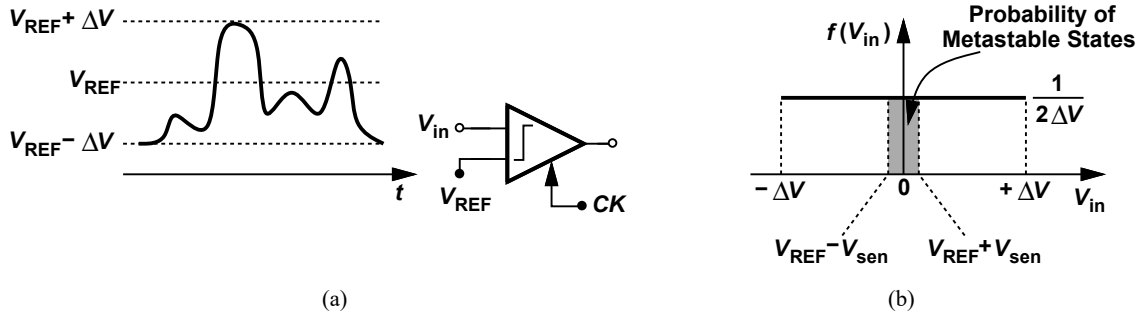


Figure 6.6 (a) Random signal swinging between two values, and (b) distribution of the signal.

by $(2V_{sen})/(2\Delta V)$:

$$P_{\text{error}} = \frac{0.9I_{SS} \exp \frac{-T_m}{\tau_{reg}}}{g_m \Delta V}. \quad (6.38)$$

The error rate is an exponential function of T_m/τ_{reg} and is proportional to $I_{SS}/(g_m \Delta V)$.

In practice, random signals have a Gaussian distribution. In the foregoing example, therefore, the signal has a mean value equal to V_{REF} and a standard deviation, σ (Fig. 6.7):

$$f(V_{in}) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(V_{in} - V_{REF})^2}{2\sigma^2}. \quad (6.39)$$

To find the probability of metastable states for the current-steering comparator, we must integrate the Gaus-

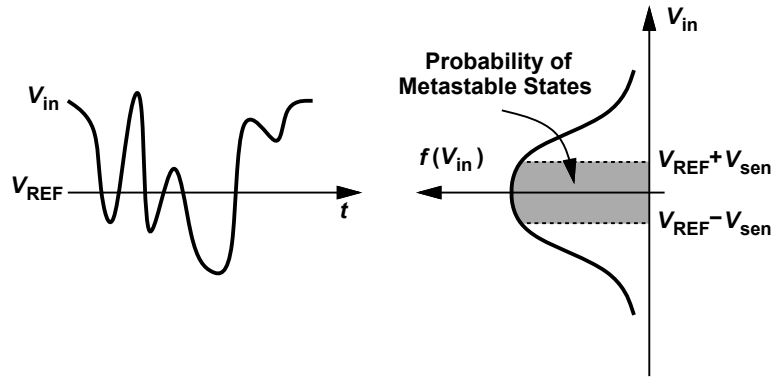


Figure 6.7 Random signal with Gaussian distribution.

sian from $V_{REF} - V_{sen}$ to $V_{REF} + V_{sen}$. But, since this input voltage difference is very small, we can approximate the area by a rectangle having a height of $1/(\sqrt{2\pi}\sigma)$ and a width of $2V_{sen}$:

$$P_{\text{error}} = \frac{1}{\sqrt{2\pi}\sigma} \frac{1.8I_{SS}}{g_m} \exp \frac{-T_m}{\tau_{reg}}. \quad (6.40)$$

How do (6.38) and (6.40) compare? We note that they differ by a factor of $(1/\Delta V)/[2/(\sqrt{2\pi}\sigma)]$. For a Gaussian signal, it is not meaningful to specify a *maximum* swing (why?) but we can say that the signal amplitude rarely exceeds $\pm 3\sigma$. Approximating the maximum swing in the Gaussian distribution by

$\Delta V = 3\sigma$, we obtain a ratio of 0.41. That is, the uniform approximation underestimates the error rate by about 60% because it assumes small and large voltage swings occur with equal probabilities.

Example 6.7

Suppose the input transistors of a current-steering comparator are designed with $V_{GS} - V_{TH} = 100$ mV. Using the values in Example 6.6 and assuming a Gaussian signal with $\sigma = 200$ mV, calculate the metastability error rate.

Solution

Since $g_m = I_{SS}/(V_{GS} - V_{TH})$, we have from Eq. (6.40):

$$P_{error} = 0.72 \frac{V_{GS} - V_{TH}}{\sigma} \exp \frac{-T_m}{\tau_{reg}}. \quad (6.41)$$

It follows that

$$P_{error} = 1.63 \times 10^{-5}. \quad (6.42)$$

As explained in the next chapter, metastable states can produce gross errors in an ADC output. Thus, typical ADCs must achieve metastability error rates in the range of 10^{-14} to 10^{-10} . For example, in data communication systems, even one error in 10^{10} samples can have disastrous consequences. We recognize that a design targeting such rare metastable states incurs negligible SNR loss. That is, the metastability rate and the SNR are fairly independent because metastable errors appear very infrequently.

6.2.2 Metastability in StrongArm Comparator

Recall from Section 5.8.3 that the StrongArm comparator in Fig. 6.8(a) provides regenerative action when the cross-coupled PMOS devices turn on. We also know that the circuit amplifies the input difference before this occurs. Let us examine the comparator's behavior as $V_{in1} - V_{in2}$ becomes very small. As shown in Fig. 6.8(b), the CM level at P and Q falls at a rate given by the capacitance at these nodes and the tail current; thus, the time at which V_P and V_Q drop to $V_{DD} - V_{THN}$ is relatively independent of $V_{in1} - V_{in2}$ and given by $C_{PQ}V_{THN}/I_T$. Similarly, V_X and V_Y fall, approximately linearly, to $V_{DD} - |V_{THP}|$ at a rate that does not depend much on $V_{in1} - V_{in2}$. We then conclude that, for a small input difference, M_3 and M_4 turn on together and at a fairly fixed time, t_2 . Beyond this point, regeneration occurs.

From the simplified circuit of Fig. 6.8(c) and with the aid of Eq. (6.32), we write

$$V_{XY} = V_{XY0} \exp \frac{t}{\tau_{reg}}, \quad (6.43)$$

where V_{XY0} is the difference between V_X and V_Y at the moment M_3 and M_4 turn on and, from (5.19),

$$\tau_{reg} = \frac{r_{OP}}{g_{mP}r_{OP} - 1} C_L. \quad (6.44)$$

Here, r_{OP} denotes the output resistance of M_3 and M_4 . As illustrated in Fig. 6.5(b), if V_{XY0} is halved, the regenerative waveforms at X and Y are delayed by $0.69\tau_{reg}$.

Example 6.8

Compute the probability of metastable states for a StrongArm comparator sensing a random signal whose amplitude is distributed uniformly between $-\Delta V$ and $+\Delta V$.

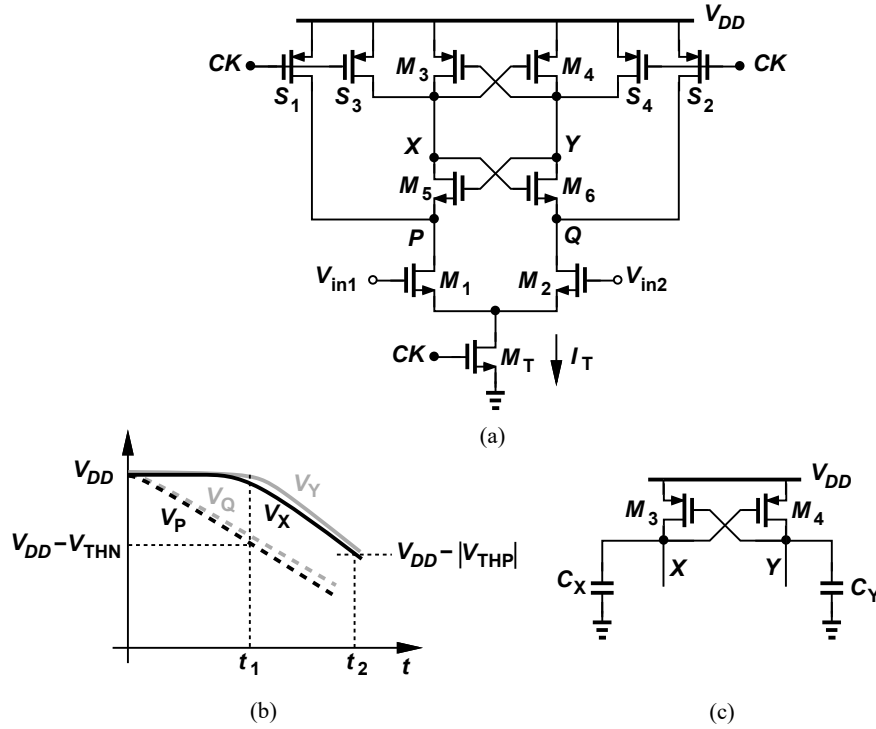


Figure 6.8 (a) StrongArm comparator, (b) its internal waveforms, and (c) PMOS pair in the regeneration mode.

Solution

In a manner similar to the current-steering topology analysis in Section 6.2.1, let us define metastability as when $V_{XY} < 0.9V_{DD}$ at the end of the comparison time. Taking into account the initial voltage gain, A_0 , provided by M_1 - M_2 and M_5 - M_6 in Fig. 6.8(a) (Section 5.8.3), we express the input difference causing metastability as $0.9V_{DD} \exp(-T_m/\tau_{reg})/A_0$. This input must be divided by ΔV to yield the probability:

$$P_{\text{error}} = \frac{0.9V_{DD}}{A_0\Delta V} \exp \frac{-T_m}{\tau_{reg}}. \quad (6.45)$$

Recall from Section 5.8.5 that the StrongArm comparator must be followed by an RS latch. Such a cascade exhibits interesting and more complex metastability properties that are beyond the scope of this book.

6.3 Noise in StrongArm Comparator

Unlike the current-steering comparator, the StrongArm topology undergoes different operation modes and does not lend itself to the noise analysis techniques that we learn in basic analog design. In this section, we introduce additional concepts that allow us to approximate the input-referred thermal noise of the StrongArm circuit.

6.3.1 Noise in Time Domain

For our study in this section, we rely on two principles from the theory of signals and systems. First, the average power of a random signal or noise, $x(t)$, is obtained as

$$P_{avg} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{+T/2} x^2(t) dt. \quad (6.46)$$

We can view this quantity as the *time average* of $x^2(t)$. Alternatively, the average of $x^2(t)$ can be obtained from the probability distribution function of $x(t)$:

$$P_{avg} = E[x^2(t)], \quad (6.47)$$

where $E(\cdot)$ denotes the expectation. This quantity is called the “ensemble” average of $x^2(t)$ or the “variance” of $x(t)$. The square root of the variance is the rms noise. For the random signals of interest to us, the time and ensemble averages are equal.

Second, if a signal with a (power) spectrum $S_x(f)$ is applied to a transfer function $H(s)$, it yields an output spectrum given by $S_x(f)|H(f)|^2$.

Now, consider a white noise process whose spectrum has a height (spectral density) of 1 [Fig. 6.9(a)]. For example, if $a(t)$ is a current quantity, then $S_a(f) = 1 \text{ A}^2/\sqrt{\text{Hz}}$. Let us call $a(t)$ “unit white noise.”

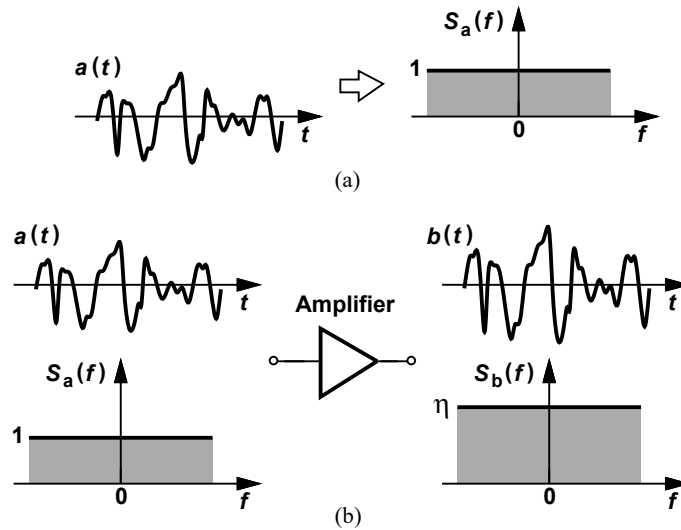


Figure 6.9 (a) White noise in time and frequency domains, and (b) white noise traveling through an amplifier.

Next, suppose we apply this noise to a noiseless amplifier, obtaining $b(t)$ and $S_b(f)$ at the output [Fig. 6.9(b)]. If $S_b(f) = \eta S_a(f)$, how do we express $b(t)$ in terms of $a(t)$? We recognize that the amplifier must have a gain of $\sqrt{\eta}$ to yield $(\sqrt{\eta})^2 S_a(f)$ at the output. Thus $b(t) = \sqrt{\eta} a(t)$. More generally, if the spectrum is multiplied by k , the time-domain noise is multiplied by $\sqrt{k}$.

For example, the thermal noise current of a MOSFET in saturation has a two-sided spectral density¹ of $2kT\gamma g_m$. Thus, the time-domain behavior can be expressed as

$$I_n(t) = \sqrt{2kT\gamma g_m} a(t), \quad (6.48)$$

where $a(t)$ denotes unit white noise.

¹A two-sided spectrum goes from $f = -\infty$ to $f = +\infty$.

Of particular interest to us is the case of *integrated* white noise. Suppose we apply our unit white noise, $a(t)$, to an integrator, as shown in Fig. 6.10. The output spectrum is readily obtained as $S_n(f)|H(f)|^2 =$

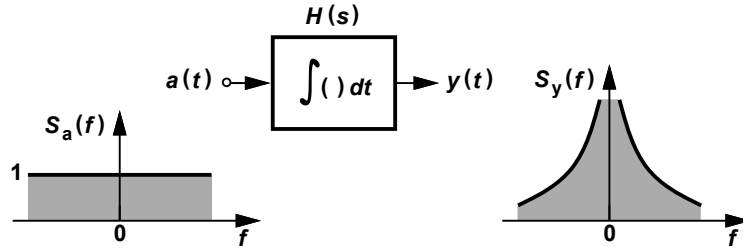


Figure 6.10 White noise applied to an integrator.

$1/(2\pi f)^2$ because $H(s) = 1/s$. But how about the time-domain behavior of the output? The average power of $y(t)$ can be computed as

$$E[y^2(t)] = E\left[\left(\int a(t)dt\right)^2\right]. \quad (6.49)$$

That is, we must find the expectation of the square of integrated white noise. Fortunately, this expression can be simplified with the aid of “Ito’s integrals” [2] for white noise:

$$E\left[\left(\int a(t)dt\right)^2\right] = E\left[\int dt\right] = t. \quad (6.50)$$

Interestingly, the short-term average *power* of integrated white noise grows linearly with time. Why is that? Illustrated in Fig. 6.11, $y(t)$ still varies randomly, but occasionally tends to accumulate in the same direction, exhibiting a greater *variance* as time progresses. We say the integral of white noise is unbounded.

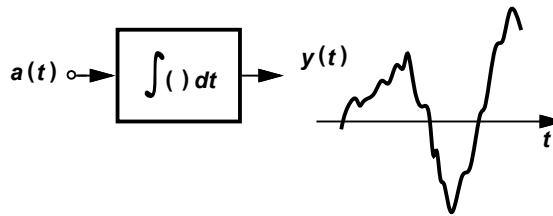


Figure 6.11 Growth of white noise at the output of an integrator.

Example 6.9

Consider the circuit shown in Fig. 6.12, where C_1 begins with a zero initial condition. Examine V_{out} in the time domain. Neglect channel-length modulation.

Solution

We have

$$V_{out} = \frac{1}{C_1} \int I_{D1}(t)dt. \quad (6.51)$$

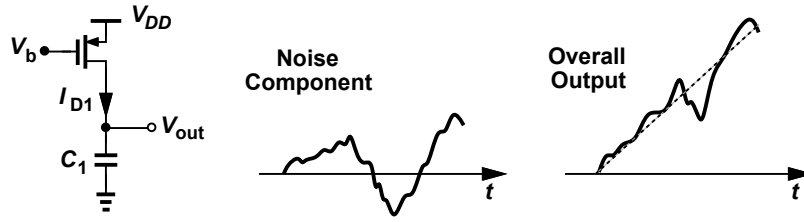


Figure 6.12 Example of white noise growth.

The drain current consists of a constant component, Eq. (6.48), which produces a ramp in V_{out} , and a random component, which is integrated by C_1 . Thus, if we disregard the ramp, the variance of the output noise voltage is given by

$$E \left[\left(\frac{1}{C_1} \int \sqrt{2kT\gamma g_m} a(t) dt \right)^2 \right] = \frac{2kT\gamma g_m}{C_1^2} t, \quad (6.52)$$

where $a(t)$ is our unit white noise. This quantity signifies the variance of V_{out} with respect to the ideal ramp at time τ . That is, statistically, the square of the difference between V_{out} and the ramp grows according to this expression.

6.3.2 Noise in Amplification Mode

We have seen in Section 5.8.1 that, in the amplification mode, the StrongArm comparator reduces to a differential pair with capacitive loads (Fig. 6.13). For simplicity, let us assume $V_{in1} = V_{in2}$ so that V_P and V_Q

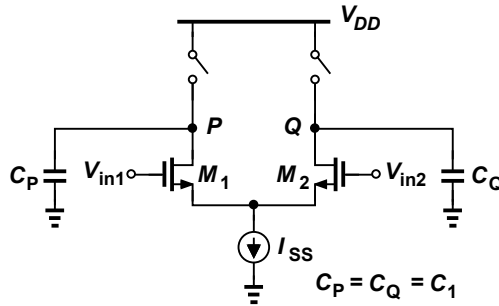


Figure 6.13 StrongArm comparator in its first amplification mode.

fall at the same rate. We wish to compute the noise in the *difference* between these two voltages, determine its rms value, and refer it to the input. Transistor M_1 generates a two-sided noise current spectral density given by $2kT\gamma g_m$, which is integrated by C_P .² Given by (6.52), the variance of the noise in V_P is equal to $(2kT\gamma g_m/C_1^2)t$ and so is that of the noise in V_Q . The noise in $V_P - V_Q$ has twice this variance.

For how long does this variance grow? The cross-coupled NMOS transistors in Fig. 6.8(a) turn on when V_P and V_Q fall to $V_{DD} - V_{THN}$. A reasonable approximation here is to consider the noise integration only

²Note that the output noise spectrum of a differential circuit can be obtained by finding that of the corresponding half circuit and doubling the result.

until this turn-on occurs [3]. The time at which $V_P \approx V_Q \approx V_{DD} - V_{THN}$ is given by $(2C_1/I_{SS})V_{THN}$, yielding

$$E[(V_P - V_Q)^2] = 2 \frac{2kT\gamma g_m}{C_1^2} \frac{2C_1}{I_{SS}} V_{THN}, \quad (6.53)$$

where the first factor of 2 on the right-hand side accounts for the noise in V_P and the noise in V_Q . The rms noise voltage measured between P and Q at this time is equal to

$$V_{PQ,rms} = \sqrt{\frac{8kT\gamma g_m}{C_1 I_{SS}} V_{THN}}. \quad (6.54)$$

In the next step, we divide the noise by the gain, which was computed as $(2g_m/I_{SS})V_{THN}$ in Section 5.8.3:

$$V_{n,in,rms} = \sqrt{\frac{2kT\gamma I_{SS}}{g_m C_1 V_{THN}}}. \quad (6.55)$$

Example 6.10

A StrongArm comparator is designed with the following parameter values: $C_1 = 5$ fF, $I_{SS} = 200$ μ A, $(V_{GS} - V_{TH})_{1,2} = 200$ mV, and $V_{THN} = 300$ mV. Determine the differential noise between P and Q and the input-referred noise if $\gamma = 1$.

Solution

We have $g_m \approx I_{SS}/(V_{GS} - V_{TH})_{1,2} = 1/(1 \text{ k}\Omega)$ and hence

$$V_{PQ,rms} = 3.15 \text{ mV}_{rms}. \quad (6.56)$$

The voltage gain is equal to 3, yielding

$$V_{n,in,rms} = 1.05 \text{ mV}_{rms}. \quad (6.57)$$

In addition to the noise of the input transistors, we must also take into account the kT/C noise associated with the switches that precharge P and Q in the reset mode. The total differential noise between these nodes is thus equal to

$$V_{PQ,rms}^2 = \frac{8kT\gamma g_m}{C_1 I_{SS}} V_{THN} + \frac{2kT}{C_1}. \quad (6.58)$$

The square root of this quantity must be divided by $(2g_m/I_{SS})V_{THN}$ to obtain the input-referred noise:

$$V_{n,in,rms} = \sqrt{\frac{2kT\gamma I_{SS}}{g_m C_1 V_{THN}} + \frac{kT}{C_1} \cdot \frac{I_{SS}^2}{2g_m^2 V_{THN}^2}}. \quad (6.59)$$

Example 6.11

Compare the relative values of the input transistors' noise and kT/C noise.

Solution

Writing $g_m = I_{SS}/(V_{GS} - V_{TH})_{1,2}$, we have

$$V_{n,in,rms}^2 = \frac{2kT}{C_1} \frac{(V_{GS} - V_{TH})_{1,2}}{V_{THN}} \left[\gamma + \frac{(V_{GS} - V_{TH})_{1,2}}{4V_{THN}} \right]. \quad (6.60)$$

In practice, $4\gamma V_{THN}$ tends to be several times $(V_{GS} - V_{TH})_{1,2}$. In Example 6.10, they have a ratio of 6. That is, the kT/C noise component is small.

Equation (6.59) provides a reasonable approximation of the StrongArm comparator input noise [3] even though P and Q in Fig. 6.8(a) are *not* the final output nodes of the circuits. Other transistors' noise contributions can also be formulated [3].

6.4 Simulation of Comparator Noise

The noise derivations in the previous sections allow us to estimate the noise of some comparator topologies, but we also wish to verify these results by simulations. Unlike small-signal analog circuits, a comparator generates only a digital output, which carries no direct information about the noise. In other words, we do not have an “output noise” or an input-output “gain” to compute the input-referred noise. We introduce here a general methodology for comparator noise simulation. We assume Gaussian noise sources, noting that the standard deviation, σ , of a Gaussian noise process is equal to the rms noise. We also assume that the circuit simulator can include noise in the transient analysis (e.g., as is possible in Cadence).

Our first simulation proceeds as follows. Suppose a comparator having a zero offset and a zero differential input is clocked many times. The output is initially metastable but, owing to the circuit's noise, it eventually recovers and assumes a valid logical level if the clock period is chosen long enough. Since the input-referred noise voltage, $V_{n,in}$, has positive and negative values with equal probabilities, the comparator output also reaches a high level for 50% of the samples and a low level for the other 50%. Figure 6.14(a) conceptually depicts this situation.³ Also plotted here is $f_X(x)$, which denotes the Gaussian input noise distribution (i.e., $x = V_{n,in}$). This distribution yields equal probabilities for ONEs (when $V_{n,in} > 0$) and ZEROs (when $V_{n,in} < 0$) because its mean is zero. [The total area under $f_X(x)$ is equal to 1.] We begin with this simulation so as to ensure numerical errors in the simulator do not create any “bias” (nonzero mean) in the output. Note that V_{CM} sets the input CM level and is constant.

In the next step, we apply a small, constant differential input, say, $V_S = +2$ mV [Fig. 6.14(b)], and repeat the simulations. In the absence of noise, the output would reach a ONE in every clock cycle. With noise present, on the other hand, the sum of $+2$ mV and the input-referred noise occasionally becomes *negative*, leading to a ZERO output. Thus, the relative numbers of low and high samples signify the “strength” of the noise. Writing

$$V_S + V_{n,in} < 0 \quad (6.61)$$

as the condition for an incorrect output, we observe that the probability of ZEROs is given by how often $V_{n,in}$ is less than $-V_S$, as shown in the Gaussian distribution of Fig. 6.14(b). If we denote the number of ONEs by n_1 and the number of ZEROs by n_0 , we have

$$\frac{n_0}{n_1} = \frac{\int_{-\infty}^{-V_S} f_X(x) dx}{1 - \int_{-\infty}^{-V_S} f_X(x) dx}, \quad (6.62)$$

because the total area under $f_X(x)$ is 1. Note that n_0 and n_1 are obtained from transient simulations.

³For illustration purposes, all of the output waveforms begin at $t = 0$ here.

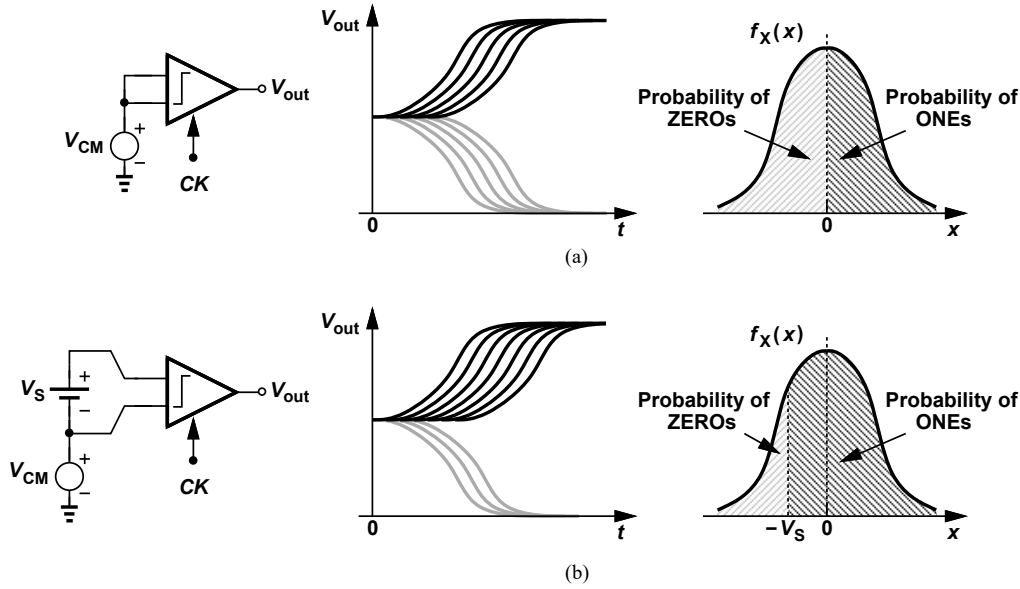


Figure 6.14 A comparator sensing a (a) zero input difference and its output behavior, and (b) a finite input difference and its output behavior.

In the last step of our analysis, we must compute $f_X(x)$ from Eq. (6.62), recognizing that a zero-mean Gaussian distribution is characterized by only one parameter, namely, its standard deviation, σ . We have

$$\frac{\int_{-\infty}^{-V_S} \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-x^2}{2\sigma^2} dx}{1 - \int_{-\infty}^{-V_S} \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-x^2}{2\sigma^2} dx} = \frac{n_0}{n_1}. \quad (6.63)$$

The calculation of σ (the input-referred rms noise voltage) from this equation requires numerical iterations. But the following example simplifies the task.

Example 6.12

How should V_S in Fig. 6.14(b) be chosen?

Solution

If $|V_S|$ is small compared to σ , then the two areas under the Gaussian in Fig. 6.14(b) have only a small difference, yielding $n_0/n_1 \approx 0.5$ and requiring a large number of clock cycles so as to accurately distinguish n_0/n_1 from 0.5. On the other hand, if $|V_S|$ is large compared to σ , then the area on the left is very small, demanding many data points as well. We should therefore select $|V_S| \approx \sigma$. Let us make such a choice (Fig. 6.15) and note that the area under a Gaussian between $-\sigma$ and $+\sigma$ is equal to 0.63. Since the area from $x = -\infty$ to $x = -\sigma$ is equal to $(1 - 0.63)/2 = 0.185$, we have from (6.62)

$$\frac{n_0}{n_1} = \frac{0.185}{1 - 0.185} \quad (6.64)$$

$$= 0.227. \quad (6.65)$$

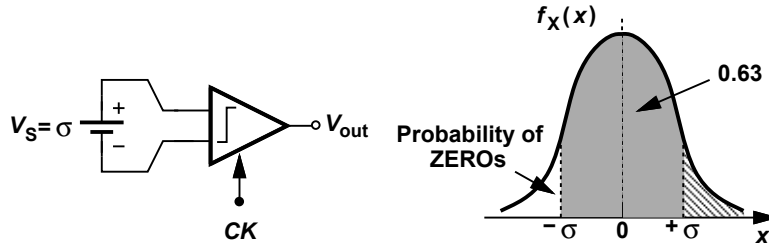


Figure 6.15 Behavior of a comparator with an input imbalance equal to its rms noise.

More importantly,

$$\frac{n_0}{n_1 + n_0} = \frac{0.227}{1 + 0.227} \quad (6.66)$$

$$= 0.185, \quad (6.67)$$

suggesting that the comparator output is low for about 18.5% of the clock cycles if $|V_S| = \sigma$.

Of course, we do not *know* the value of σ before we begin the simulations. Nonetheless, if we proceed with a guess for σ and observe that $n_0/(n_1 + n_0)$ is far from 0.185, we should adjust V_S .

This example also offers a method for automating the comparator noise simulation: we select a value for V_S , perform a long simulation, find $n_0/(n_1 + n_0)$, and if it is not equal to 0.185, adjust V_S accordingly, and repeat the simulation. After this “loop” settles, the value of V_S is equal to the comparator input-referred noise voltage.

6.5 Offset Cancellation

In our study of ADCs, we will see that some architectures are sensitive to comparator offset, requiring it to be well below 1 LSB. This can be accomplished by (a) employing large transistors (Section 6.1), but at the cost of greater input capacitance and higher power consumption, or (b) offset cancellation. As a rule of thumb, we follow the former approach if the offset can be larger than 10 mV and the latter for smaller offsets.

Offset cancellation is based on “offset storage” in the analog domain or in the digital domain. We study both methods here.

6.5.1 Analog Offset Cancellation

Consider the simple differential amplifier shown in Fig. 6.16(a), where V_{OS1} denotes its input-referred offset. Here, if $V_{in} = 0$, then $V_{out} \neq 0$. We then surmise that the circuit must be modified such that $V_{out} = 0$ when $V_{in} = 0$. Let us introduce two capacitors in series with the outputs [Fig. 6.16(b)]. How do we guarantee that $V_{out} = 0$ with $V_{in} = 0$? We must establish proper voltages on C_1 and C_2 . As depicted in Fig. 6.16(c), we short the input nodes and the output nodes, allowing V_{OS1} to be amplified and stored on C_1 and C_2 . After C_1 and C_2 are charged, we remove the input and output shorts [Fig. 6.16(d)]. Now, $V_{out} = 0$ if $V_{in} = 0$ because C_1 and C_2 retain their voltages as the circuit goes from Fig. 6.16(c) to Fig. 6.16(d). We say the amplifier offset is stored at the output.⁴

The output storage technique, in principle, reduces the offset to zero. But how can it be applied to the current-steering or StrongArm comparators? Since the input differential pairs in these topologies do not readily lend themselves to output offset storage, we can *precede* the comparator with the amplifier of Fig. 6.16(d),

⁴We have assumed that A_0 does not saturate due to its own offset.

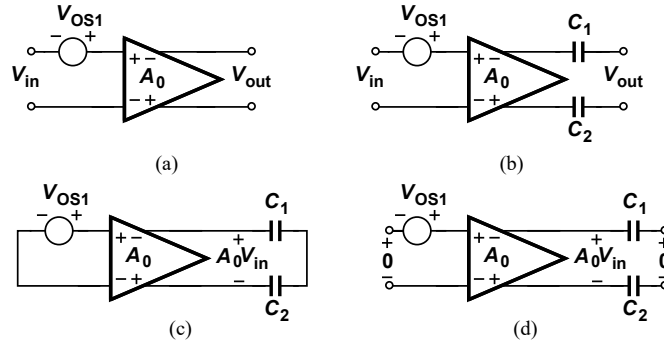


Figure 6.16 (a) An amplifier with an input-referred offset, (b) addition of capacitors at the output, (c) storage of output offset with a zero input, and (d) circuit after offset cancellation.

as illustrated in Fig. 6.17. This cascade contains two offset voltages: one from A_0 , which is completely canceled, and one from the comparator, which is divided by the preceding gain when referred to the main input. That is, the overall circuit now exhibits an input offset approximately equal to V_{OS2}/A_0 .

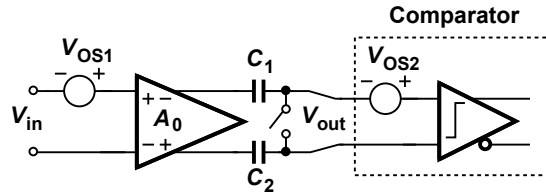


Figure 6.17 Offset-canceled amplifier followed by a comparator.

Example 6.13

Suppose V_{in} jumps from zero to 5 mV in Fig. 6.17. Explain how V_{out} changes.

Solution

The amplifier output changes from $A_0 V_{OS}$ to $A_0 V_{OS} + A_0 \times (5 \text{ mV})$. Taking into account the comparator input capacitance, we draw the equivalent circuit shown in Fig. 6.18. The voltage change provided by the amplifier is attenuated by the capacitive voltage divider. Assuming $C_1 = C_2$, the reader can prove that

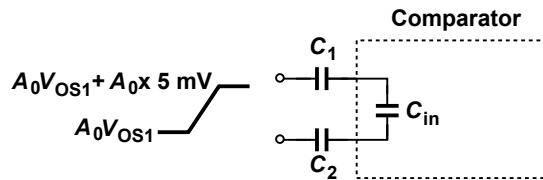


Figure 6.18 Effect of comparator input capacitance.

$$V_{out} = \frac{C_1}{C_1 + 2C_{in}} A_0 \times (5 \text{ mV}). \quad (6.68)$$

The key point here is that the gain preceding the comparator is less than A_0 . The comparator offset is then divided by $A_0 C_1 / (C_1 + 2C_{in})$ rather than by A_0 . For this reason, we choose $C_1 \gg C_{in}$.

The output offset storage scheme in Fig. 6.17 requires three modes of operation: (1) V_{in} and V_{out} are set to zero and the offset is stored on C_1 and C_2 ; (2) the desired input is applied and amplified; and (3) the comparator is clocked to deliver a logical output. These modes can be created through the use of switches and proper clock phases. Figure 6.19 depicts the switching network, where we disregard the C_p capacitors

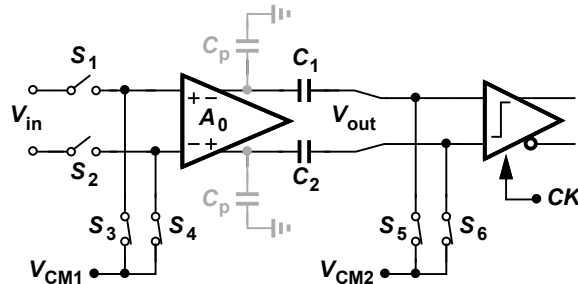


Figure 6.19 Detailed switching for offset cancellation.

for now. In the offset cancellation mode, S_3 - S_6 are on, forcing the amplifier input and the comparator input to zero. Common-mode levels V_{CM1} and V_{CM2} are so chosen to allow proper operation for A_0 and the comparator. Note that C_1 and C_2 present a CM level equal to V_{CM2} even after S_5 and S_6 turn off. Next, S_3 - S_6 turn off and S_1 and S_2 turn on. After the amplifier output settles to $A_0 V_{in}$, the comparator is clocked. The reader is encouraged to draw the clock phases that control the circuit.

Example 6.14

A student surmises that offset cancellation can be performed when the circuit is powered up and need not be repeated in every clock cycle. Is that possible?

Solution

No, it is not. From the simplified circuit in Fig. 6.20, we observe that, if S_5 and S_6 remain off for a long time, then the leakage current drawn by their drain-bulk diodes reduces the common-mode level at the comparator input to zero. As a result, the comparator fails. Thus, the three phases must be repeated periodically, and, in practice, in every clock cycle.

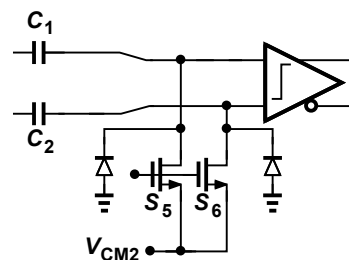


Figure 6.20 Effect of leakage on offset storage.

The arrangement of Fig. 6.19 exemplifies the complexity that accompanies offset cancellation. In fact, the area occupied by the circuitry preceding the comparator can exceed that of the comparator itself. Moreover, the circuit's speed degrades due to (a) the three necessary modes, and (b) the parasitic capacitances of C_1 and C_2 , denoted by C_p in Fig. 6.19. Typical on-chip capacitors exhibit a parasitic of about 5%, loading the output nodes of A_0 and slowing down the response in the amplification mode.

The reader may wonder whether A_0 can be increased by employing multiple gain stages. Such a structure consumes greater power and provides less bandwidth.

Example 6.15

Compute the residual offset of the circuit in Fig. 6.19.

Solution

The offset of A_0 is canceled, and that of the comparator, V_{OS2} , is divided by the gain preceding it. From Example 6.13, we have

$$V_{OS,in} = V_{OS2} \div \left(\frac{C_1}{C_1 + 2C_{in}} A_0 \right) \quad (6.69)$$

$$= \frac{V_{OS2}}{A_0} \left(1 + \frac{2C_{in}}{C_1} \right). \quad (6.70)$$

Neglected here are the capacitances associated with S_5 and S_6 , which add to C_{in} .

Another analog cancellation technique inserts the offset-storage capacitors at the *input* of the preamplifier. Consider the arrangement shown in Fig. 6.21(a). What happens if A_0 is placed in a unity-gain negative-

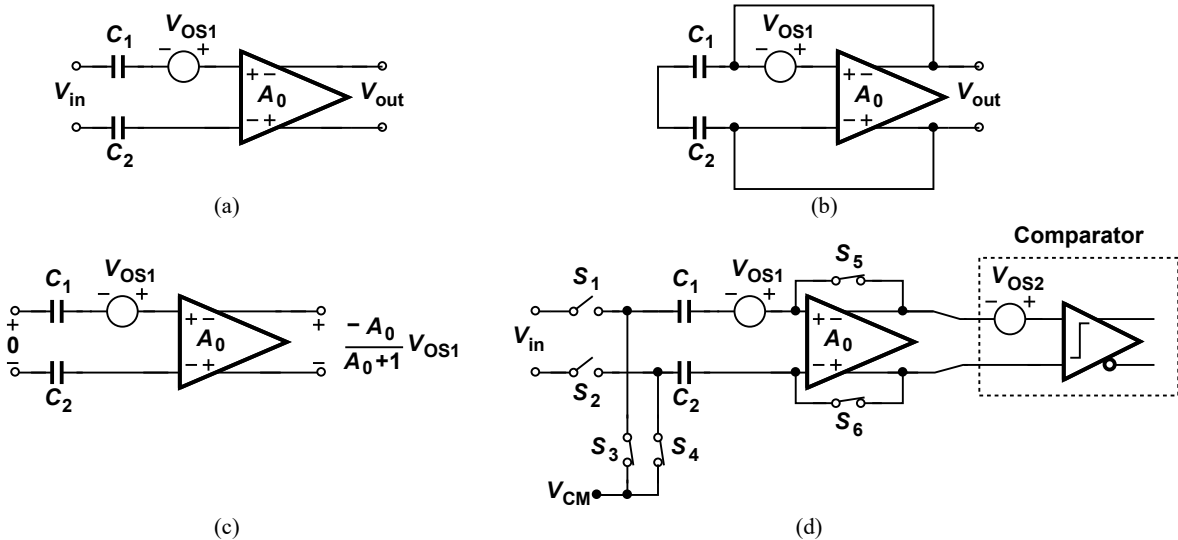


Figure 6.21 (a) Amplifier with input-offset storage capacitors, (b) offset cancellation mode, (c) residual offset at the output, and (d) amplifier followed by a comparator.

feedback loop while $V_{in} = 0$ [Fig. 6.21(b)]? Since the net voltage sensed by A_0 is equal to $V_{out} + V_{OS1}$, we have $(V_{out} + V_{OS1})(-A_0) = V_{out}$ and hence

$$V_{out} = \frac{-A_0}{1 + A_0} V_{OS1}. \quad (6.71)$$

It is seen that negative feedback reduces the offset by a factor of 1 plus the loop gain. This result is stored on C_1 and C_2 , maintaining the same V_{out} even if the feedback is removed. Thus, for a zero input, the output is equal to $[-A_0/(1 + A_0)]V_{OS1}$ [Fig. 6.21(c)], and the input-referred offset is approximately equal to

$-V_{OS1}/(1 + A_0)$. The attenuation due to the preamplifier input capacitance, C_{in} , raises the input-referred offset in the same manner as in Eq. (6.70).

Figure 6.21(d) depicts the input offset storage method along with the necessary switches. The operation phases are similar to those in Fig. 6.19. We recognize, however, that the residual offset is *greater* here because the preamplifier offset is not completely canceled:

$$V_{OS,in} = \frac{V_{OS1}}{1 + A_0} \left(1 + \frac{2C_{in}}{C_1} \right) + \frac{V_{OS2}}{A_0} \left(1 + \frac{2C_{in}}{C_1} \right). \quad (6.72)$$

In summary, analog offset cancellation entails two drawbacks: (1) it requires multiple phases, slowing down the operation, and (2) it reduces the offset by only a moderate amount because A_0 hardly exceeds 3 or 4 for a one-stage preamplifier. For these reasons, we generally prefer digital offset cancellation.

6.5.2 Digital Offset Cancellation

Consider the differential pair shown in Fig. 6.22(a). Owing to the offset, $V_X \neq V_Y$ when $V_{in1} = V_{in2}$. Can

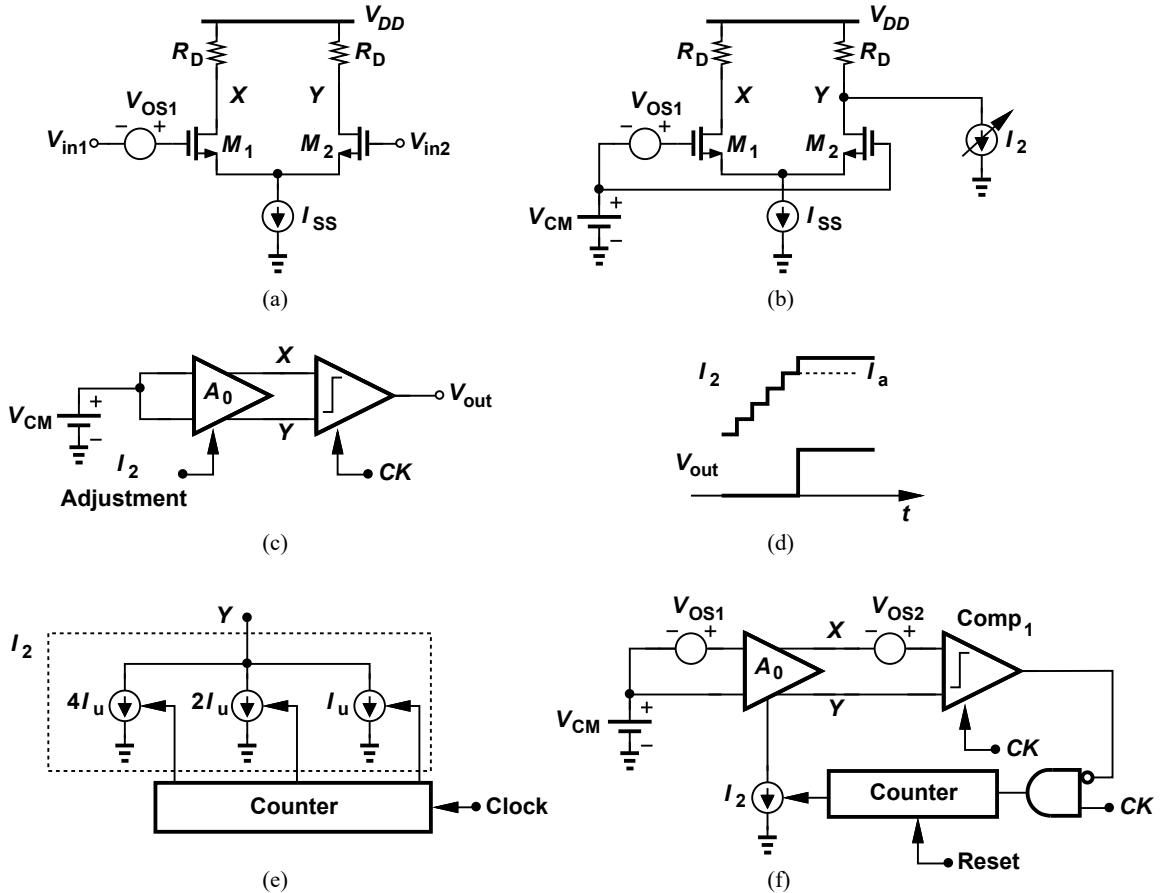


Figure 6.22 (a) Differential pair with an input-referred offset, (b) cancellation of offset by a current source, (c) measurement of $V_X - V_Y$ by a comparator, (d) adjustment of I_2 , (e) implementation of I_2 , and (f) overall digital control loop.

we add a mechanism to ensure $V_X = V_Y$ in this case? Suppose $V_{OS1} > 0$ and hence $V_X < V_Y$. We can lower V_Y by drawing a current from Y [Fig. 6.22(b)]. If chosen properly, I_2 yields $V_Y = V_X$. The offset is

thus canceled without the use of capacitors in the signal path. Also, if I_2 is set to and “frozen” at its proper value at power-up, offset cancellation need not be repeated in every clock cycle.

We must now answer two questions: (1) how do we set I_2 to its proper value?, and (2) how do we freeze I_2 ? For the first task, we must adjust I_2 while measuring the output until $V_X \approx V_Y$. This is accomplished by attaching a comparator to X and Y [Fig. 6.22(c)] and clocking it after each adjustment of I_2 .⁵ For example, we set I_2 to a low value, clock the comparator and examine its logical output, which would be low in this case because $V_X < V_Y$. Next, we repeat this sequence by stepping I_2 up [Fig. 6.22(d)] until the comparator output changes, signifying that V_X is slightly greater than V_Y . At this point, I_2 is set near its proper value, and the comparator is nearly metastable.

For the second task, we must control I_2 digitally. We decompose I_2 into a number of current sources so that they can be controlled by a digital binary input. As shown in Fig. 6.22(e), the current sources consist of an LSB unit, I_u , another twice its value, etc. This “binary-weighted array” is controlled by a counter. The counter begins from 000 and counts up, allowing I_2 to go from zero to I_u to $2I_u$, etc. As explained in Chapter 8, the array is a type of digital-to-analog converter.

Figure 6.22(f) conceptually illustrates the overall offset cancellation loop. The main differential input is set to zero, the counter is reset, and the comparator is clocked. If its output is low, the counter receives a clock edge and increments its output to 001, causing I_2 to jump to I_u . This sequence is repeated until the comparator output goes high, at which point the AND gate disables the clock, freezing the value stored in the counter.

The digital offset cancellation in Fig. 6.22(f) may appear more complex than the analog schemes studied in the previous section, but, in modern CMOS technologies, it occupies less area due to the absence of capacitors. We must nonetheless address two points. First, what happens if Comp_1 itself has an offset, V_{OS2} ? The loop then drives the preamplifier output to slightly above $-V_{OS2}$ so that the comparator flips and stops the counter. Since $V_X - V_Y \neq 0$, the preamplifier offset is not canceled completely. Interestingly, however, the offset of the preamplifier/comparator cascade is canceled. This can be seen by noting that, with a zero input difference, I_2 is incremented to the point that the comparator output barely flips. We thus view and use A_0 and Comp_1 as one comparator.

Second, how can offset cancellation proceed if V_{OS} in Fig. 6.22(b) is *negative*? This requires that we include another programmable current source at node X [Fig. 6.23(a)]. In this case, I_1 and I_2 begin at half of their maximum values and, under the command of the counter, vary differentially [Fig. 6.23(b)] until $V_X - V_Y$ makes the comparator nearly metastable. At the beginning of the iteration, the logic must decide whether $I_1 - I_2$ should increase or decrease.

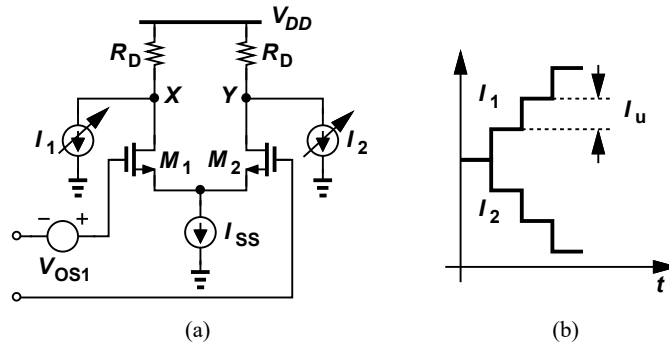


Figure 6.23 Differential adjustment of offset.

⁵We assume for now that the comparator itself has no offset.

Example 6.16

How are I_u and the maximum (“full-scale”) current in Fig. 6.23(b) chosen?

Solution

The LSB value, I_u , determines the residual offset (the value after cancellation is finished) while the total current defines the maximum offset that can be corrected. For example, suppose V_{OS1} yields a maximum of $|V_X - V_Y| = 30$ mV. Also, $R_D = 1$ k Ω . We target a residual offset of 5 mV. The full-scale value of I_1 and I_2 should be around 1 k Ω /30 mV = 30 μ A so that the offset can be removed when I_1 reaches its maximum and I_2 drops to zero. After offset cancellation is completed, the residual value is $|V_X - V_Y| \leq I_u R_D$, and hence I_u must be no more than 5 μ A. In other words, I_1 and I_2 must be adjustable by six or seven steps and with an LSB of 5 μ A. If the comparator in Fig. 6.22(f) has its own offset, the full-scale current must be accordingly larger.

6.5.3 Offset Cancellation in StrongArm Comparator

Recall from Chapter 5 that the amplification phase in the StrongArm comparator is based on drawing charge from load capacitances rather than drawing currents from load resistors. We also pointed out that mismatch between these capacitances contributes offset. We then surmise that the circuit’s offset can be nulled by deliberately introducing mismatch between the load capacitances. Referring to Fig. 6.24(a), let us first assume $V_{in1} - V_{in2}$ is small and constant. We observe that the currents flowing through C_P and C_Q are equal to $(I_{SS}/2) + g_m(V_{in1} - V_{in2})$ and $(I_{SS}/2) - g_m(V_{in1} - V_{in2})$, respectively. Thus, V_P and V_Q fall according to:

$$V_P = V_{DD} - \left[\frac{I_{SS}}{2} + g_m(V_{in1} - V_{in2}) \right] \frac{t}{C_P} \quad (6.73)$$

$$V_Q = V_{DD} - \left[\frac{I_{SS}}{2} - g_m(V_{in1} - V_{in2}) \right] \frac{t}{C_Q}. \quad (6.74)$$

For offset cancellation, we choose $V_{in1} = V_{in2}$ and write $V_P - V_Q = V_{PQ}$ as

$$V_{PQ}(t) = -\frac{I_{SS}}{2} \left(\frac{1}{C_P} - \frac{1}{C_Q} \right) t \quad (6.75)$$

$$= \frac{I_{SS}}{2} \frac{C_P - C_Q}{C_P C_Q} t. \quad (6.76)$$

This difference grows for the duration of the amplification phase, which, from Chapter 5, is roughly given by

$$t_1 = \frac{2C_P}{I_{SS}} V_{THN}. \quad (6.77)$$

It follows that

$$V_{PQ}(t_1) \approx \left(\frac{C_P}{C_Q} - 1 \right) V_{THN}. \quad (6.78)$$

Dividing this result by the gain, $(2g_{m1,2}/I_{SS})V_{THN}$, we obtain the input-referred offset:

$$V_{OS,in} = \frac{I_{SS}}{2g_{m1,2}} \left(\frac{C_P}{C_Q} - 1 \right). \quad (6.79)$$

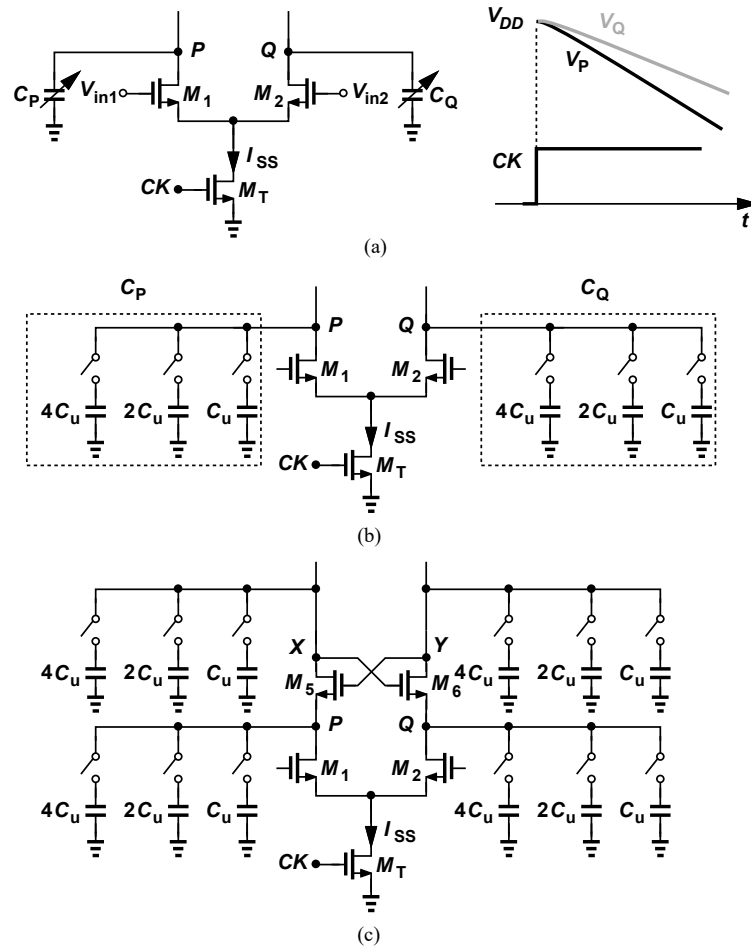


Figure 6.24 (a) StrongArm input stage with unequal load capacitances, (b) programmable capacitors at P and Q for offset cancellation, and (c) additional capacitors at X and Y for offset cancellation.

Thus, adjusting the capacitance at P or Q compensates for the circuit's offset. For example, C_P and C_Q can be realized as binary-weighted programmable capacitors [Fig. 6.24(b)] and controlled by counters. The same idea can be applied to nodes X and Y as well [Fig. 6.24(c)].

This offset cancellation method faces two issues. First, the use of additional node capacitances degrades the speed and raises the power consumption ($= fCV_{DD}^2$). Second, the capacitor arrays attached to each node occupy a large area, a serious difficulty if a large number of comparators are necessary in a design.

References

- [1]. M. Pelgrom et al, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 1433-1439, Oct. 1989.
- [2]. B. Oksendal, *Stochastic Differential Equations*, Springer-Verlag, New York, 1992.
- [3]. P. Nuzzo et al, "Noise analysis of regenerative comparators for reconfigurable ADC architectures," *IEEE Transactions on Circuits and Systems I*, vol. 55, pp. 1441-1454, July 2008.

Problems

- 6.1.** In the amplifier shown in Fig. 6.25, the first stage exhibits an offset, V_{OS1} , with $\sigma_1 = 5$ mV. If the voltage gain of this stage, A_1 , is equal to 3, determine from Eq. (6.1) the maximum tolerable σ_2 of V_{OS2} if the overall input offset must not exceed 6 mV.

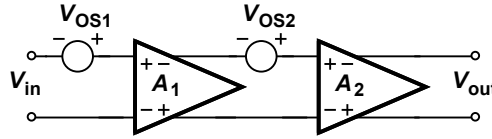


Figure 6.25 A cascade.

- 6.2.** In the circuit of Fig. 6.2, we double I_{SS} . Assume that M_1 and M_2 act as square-law devices, explain what happens to the input-referred offset of the circuit.
- 6.3.** Suppose the width of the transistor shown in Fig. 6.3(a) experiences a variability of δW : $W = W_0 + \delta W$. Assume δW has a Gaussian distribution with an SD of σ . Now, we double the length to $2L_1$ and visualize the device as two in series (Fig. 6.26). Assuming, for the sake of simplicity, that both devices operate in the deep-triode region, express their total resistance in terms of $W_1/L = (W_0 + \delta W)/L$ and $W_2/L = (W_0 + \delta W)/L$ and show that the equivalent device has an equivalent W/L of $(W/2 + \delta W/2)/L$ and hence an SD of $\sigma/\sqrt{2}$ for its width.

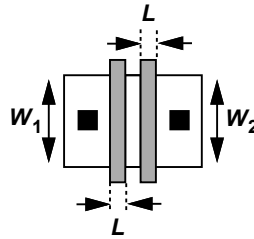


Figure 6.26 Series devices.

- 6.4.** Repeat Example 6.3 if the length of the transistor is doubled. For this purpose, refer to the equivalent structure in Fig. 6.26 and assume both devices act as resistors and have the same W/L ratio. Now, denote their thresholds by $V_{THa} = V_{TH0} + \delta V_a$ and $V_{THb} = V_{TH0} + \delta V_b$ and compute the overall resistance. If $\delta V_a \delta V_b$ is neglected, we have

$$R_{eq} = \frac{1}{\mu_n C_{ox} \frac{W}{2L}} \cdot \frac{V_{GS} - V_{TH0} - \frac{\delta V_a + \delta V_b}{2}}{(V_{GS} - V_{TH0})(V_{GS} - V_{TH0} - \delta V_a - \delta V_b)}. \quad (6.80)$$

Next, prove that $(A - \epsilon/2)/(A - \epsilon) \approx [1 + \epsilon/(2A)]^{-1}$ and simplify the result to

$$R_{eq} = \frac{1}{\mu_n C_{ox} \frac{W}{2L}} \cdot \frac{1}{(V_{GS} - V_{TH0}) \left[1 - \frac{\delta V_a + \delta V_b}{2(V_{GS} - V_{TH0})} \right]}. \quad (6.81)$$

We thus conclude that $R_{on} \propto [V_{GS} - V_{TH0} - (\delta V_a + \delta V_b)/2]$ and hence the threshold SD falls by a factor of $\sqrt{2}$.

- 6.5.** The initial difference between V_X and V_Y in Fig. 6.5(a) is reduced by a factor of 10. How much is the time shift of these waveforms?
- 6.6.** Suppose $V_X - V_Y$ in Fig. 6.5(a) reaches V_Δ in T_m seconds with an initial imbalance of V_{XY0} . We now double R_D . Derive an expression for the time it takes $V_X - V_Y$ to reach V_Δ in this case. (Note that τ_{reg} is a function of R_D .)

- 6.7.** The comparator in Fig. 6.6 receives a signal uniformly distributed between -100 mV and $+100$ mV with respect to V_{REF} . The comparator's output can be expressed as

$$V_{out} = A_0(V_{in} - V_{REF}) \exp \frac{t}{\tau_{reg}}. \quad (6.82)$$

Determine the probability of a metastable state, defined as $|V_{out}| \leq V_{max}$, where V_{max} is a desired level. Assume the comparator has T_m seconds for regeneration.

- 6.8.** An engineer decides to precede the StrongArm comparator of Fig. 6.8(a) with a simple differential pair having a voltage gain of A_1 . Explain how the probability of metastable states in Example 6.8 changes in this case.
- 6.9.** A unit white-noise source drives a cascade consisting of an amplifier having a voltage gain of 5 and an integrator having $H(s) = 1/s$. Determine the variance of the integrator output as a function of time.
- 6.10.** We double the width of the PMOS device in Fig. 6.12. Derive an expression for the variance of V_{out} as a function of time. (Note that the bias current of the transistor is doubled.)
- 6.11.** We construct an amplifier in the form of the circuit shown in Fig. 6.13. Suppose $V_{in1} = V_{in2} = V_{CM,in}$. We wish to determine the noise in $V_P - V_Q$ until M_1 and M_2 reach the edge of saturation. Derive the variance of $V_P - V_Q$ as a function of time, compute its value at the time when M_1 and M_2 reach the edge of saturation, and find the input-referred rms noise. Do we prefer to maximize or minimize V_{CM} for reducing this noise?
- 6.12.** In the circuit of Fig. 6.13, we double I_{SS} and the widths of M_1 and M_2 . Explain what happens to $V_{PQ,rms}$ and $V_{n,in,rms}$.
- 6.13.** We have concluded in Example 6.12 that $|V_S| = \sigma$ causes the comparator output to be low for 18.5% of the time. In a particular simulation, we observe that the output is low for 30% of the time. How much is $|V_S|$ in terms of σ ?
- 6.14.** Preamplifier A_0 in Fig. 6.19 exhibits an output resistance equal to R_{out} . Determine the bandwidth at the output of A_0 , assuming that the comparator presents an input capacitance of $2C_{in}$ between its input terminals.
- 6.15.** Repeat Example 6.15 if switches S_5 and S_6 in Fig. 6.19 introduce a capacitance equal to C_S at each input of the comparator.
- 6.16.** How do we select the widths of the transistors serving as S_5 and S_6 in Fig. 6.19? In other words, how do we select their on-resistance?
- 6.17.** Does the input capacitance of A_0 in Fig. 6.21(b) affect the residual offset equation $V_{out} = -[A_0/(1 + A_0)]V_{OS1}$?
- 6.18.** Suppose V_{OS1} in Fig. 6.22(b) can be as large as 20 mV. We wish to guarantee a residual offset of less than 1 mV. Determine the unit current source, I_u , and the number of DAC bits in terms of $g_{m1,2}R_D$.
- 6.19.** In Fig. 6.24(c), we can switch a unit capacitor, C_u , onto P or X . Which one provides finer control of the offset? (Recall from Chapter 5 that the voltage gain from the main input to P and Q is *less* than that to X and Y .)

Design Study of a Comparator

With the knowledge gained in the previous chapters about comparators, we can now embark upon an actual design. In this chapter, we present the step-by-step design of a comparator and discover its various trade-offs. We select the StrongArm topology.

7.1 General Considerations

In an ADC environment, we are interested in the following comparator design parameters: input offset, speed, power consumption, metastability, kickback noise, and input-referred electronic noise. We aim for an input offset of less than 5 mV, a clock rate, f_{CK} , of 5 GHz, and a power consumption of 1 mW. After the design meets these requirements, we examine the remaining parameters and decide whether they are adequate.

Shown in Fig. 7.1, the StrongArm topology offers several desirable attributes: it requires a single clock

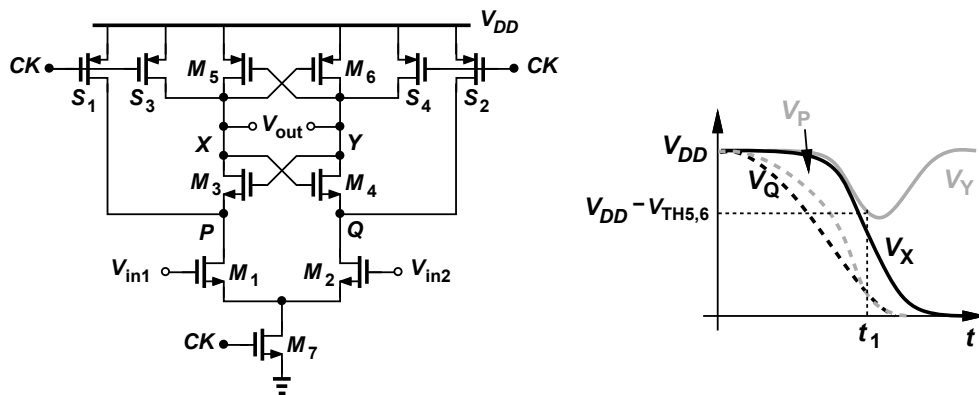


Figure 7.1 StrongArm comparator and its waveforms.

phase, draws no static power, exhibits an input offset that arises primarily from the input pair, M_1 and M_2 , and delivers rail-to-rail output swings.

The precharge action in the StrongArm latch offers two benefits. First, it enables V_P and V_Q in Fig. 7.1 to begin from V_{DD} , thus keeping M_1 and M_2 in saturation for some time. This allows the input transistors to provide gain. Second, after each comparison, the four internal nodes recover from the states developed on them and are “equalized.” This ensures that the states in one clock cycle are not inherited by the next, suppressing “dynamic” offsets. As depicted in Fig. 7.2, if, at the end of the precharge mode, V_P and V_Q do not become exactly equal and bear a difference of ΔV , the subsequent amplification mode begins with such a difference stored on C_P and C_Q , suffering from offset.

Most of our design effort is expended on selecting the transistor dimensions in Fig. 7.1. We generally begin with near-minimum dimensions unless there is a compelling reason not to do so. Also, our simulations are

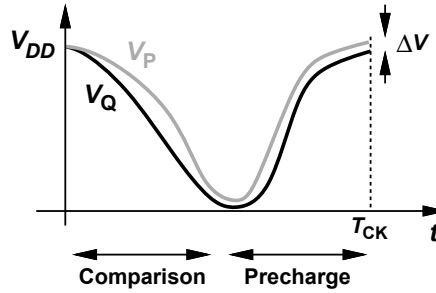


Figure 7.2 Illustration of dynamic offset.

performed under worst-case process, supply voltage, and temperature (PVT) conditions. This is because the circuit must eventually operate satisfactorily in such a corner. In this spirit, we select the slow-slow corner, $V_{DD} = 1\text{ V} - 5\% = 0.95\text{ V}$, and $T = 75^\circ\text{C}$. We also assume for the clock a 50% duty cycle and 10-ps rise and fall times. The comparator is designed in 28-nm CMOS technology.

7.2 Choice of Device Dimensions

Comparator design begins with selecting the transistor dimensions so as to meet the offset requirement. In our case, the pairs M_1 - M_2 , M_3 - M_4 , and M_5 - M_6 in Fig. 7.1 appear in the signal path and must be crafted first. Let us consider M_1 and M_2 and write their threshold voltage mismatch as

$$\Delta V_{TH1,2} = \frac{A_{VTH}}{\sqrt{(WL)_{1,2}}}, \quad (7.1)$$

where A_{VTH} is a constant [1] and about $2.2\text{ mV} \cdot \mu\text{m}$ in 28-nm technology. If we choose $W_{1,2} = 10\text{ }\mu\text{m}$ and an effective length of 25 nm, then $\Delta V_{TH} = 4.4\text{ mV}$. This appears a reasonable starting point provided that the other pairs' contributions do not raise the offset beyond the 5-mV target.

We should remark that Eq. (7.1) gives the standard deviation, σ , of the mismatch; i.e., about 68% of differential pairs in a Gaussian distribution exhibit offsets lower than this amount. In practice, we seek higher yields and must either enlarge the transistors or incorporate offset cancellation.

The tail transistor, M_7 , in Fig. 7.1 must draw sufficient current with $V_{GS7} = V_{DD}$ and $V_{DS7} = V_{in,CM} - V_{GS1,2}$, where $V_{in,CM}$ denotes the input common-mode (CM) level. With $V_{in,CM} = 0.5\text{ V}$ and $V_{GS1,2} \approx 0.35\text{ V}$, we have $V_{DS7} \approx 0.15\text{ V}$. The device thus operates in the deep triode region. Let us select $W_7 = 2\text{ }\mu\text{m}$ for a current of about 0.5 mA.

Given that the circuit provides gain before M_3 and M_4 turn on, we expect that the offset of this pair is reduced when referred to the main input. The reduction factor is, in fact, greater than the value of $2g_{m1,2}V_{TH3,4}/I_{SS}$ mentioned above. To understand why, suppose M_3 and M_4 are on (Fig. 7.3) and neglect the capacitances at nodes P and Q . Thus, I_{D1} and I_{D2} entirely flow through M_3 and M_4 , respectively, as if these transistors were absent. The offset contributed by this pair is therefore negligible unless the circuit's capacitances are taken into account. As seen below, the threshold mismatch between M_3 and M_4 is divided by a factor of 3 to 5 in typical designs. We select $W_{3,4} = 10\text{ }\mu\text{m}$ for now, expecting that this choice only slightly raises the input offset.

The PMOS cross-coupled pair in Fig. 7.1 turns on after V_X and V_Y fall by one PMOS threshold. Before this time, the circuit provides a high voltage gain, thereby reducing this pair's offset contribution considerably. In this respect, we surmise that a width of a few microns suffices for M_5 and M_6 , but we must bear in mind that these devices also amplify regeneratively and play a role in the comparator's speed. We return to this point when we optimize the design.

The reset switches, S_1 - S_4 , in Fig. 7.1 must pull their drain nodes to V_{DD} in less than 100 ps. We predict that a width of $0.5\text{ }\mu\text{m}$ to $1\text{ }\mu\text{m}$ can meet this constraint.

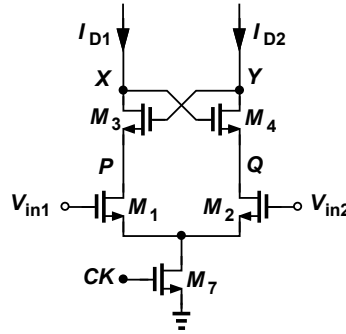


Figure 7.3 Effect of mismatch between M_3 and M_4 in the absence of capacitances at P and Q .

7.3 Basic Waveforms

Based on our foregoing thoughts, we construct the comparator shown in Fig. 7.4 and simulate it in the time

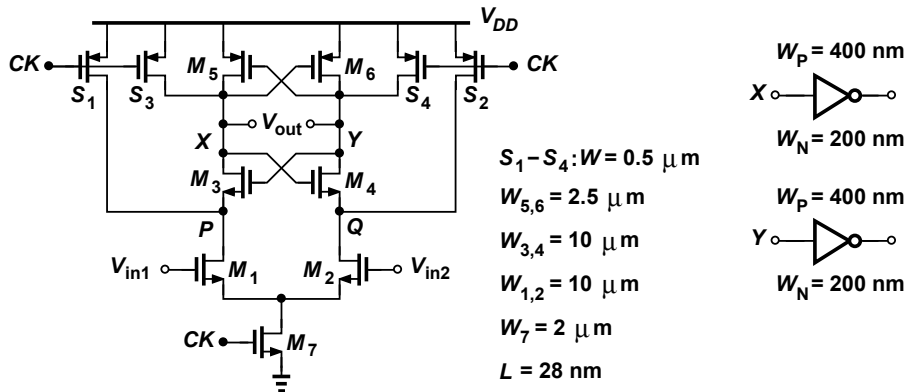


Figure 7.4 Initial design of comparator core.

domain. The output inverters act as buffers and employ relatively small transistors for now. Before optimizing the design, we familiarize ourselves with the circuit's waveforms.

The speed of the comparator depends on the input voltage difference, ultimately requiring a metastability analysis (explained later). However, it is common in ADC design to select this difference to be half of the least-significant bit (LSB), which, in view of our tolerable offset, would be 10 to 20 mV for this design. However, we apply a difference of 1 mV so as to place the circuit in “slow motion” and examine its operation details. Plotted in Fig. 7.5(a) are the voltages at nodes P , Q , X , and Y . The clock rises from zero to V_{DD} between $t = 300 \text{ ps}$ and $t = 310 \text{ ps}$. Note that V_X and V_Y experience a common-mode drop of about 400 mV before they begin to depart as a result of the regeneration provided by M_5 and M_6 . We also observe from Fig. 7.5(b) that the tail current reaches a peak of about $800 \mu\text{A}$ before V_P and V_Q drop so much as to drive the input transistors into the triode region and cause the tail node voltage to collapse.

Figure 7.6 plots $V_P - V_Q$ for $V_{in1} - V_{in2} = 1 \text{ mV}$. Two observations prove important here. First, as shown in the inset, $V_P - V_Q$ reaches -3.15 mV at $t = 315 \text{ ps}$, the greatest difference before M_3 and M_4 turn on. That is, the initial voltage gain is equal to 3.15. Second, $V_P - V_Q$ is less than $100 \mu\text{V}$ at $t = 500 \text{ ps}$, i.e., before the next clock cycle. Thus, the precharge devices are strong enough and the dynamic offset is negligible.

For design optimization, we need a metric for the circuit's speed. For example, we can find the time it takes for $|V_X - V_Y|$ to reach a certain amount, say, 200 mV. This time is measured with respect to when the clock's rising edge crosses $V_{DD}/2$, and is equal to 36 ps in Fig. 7.5(a).

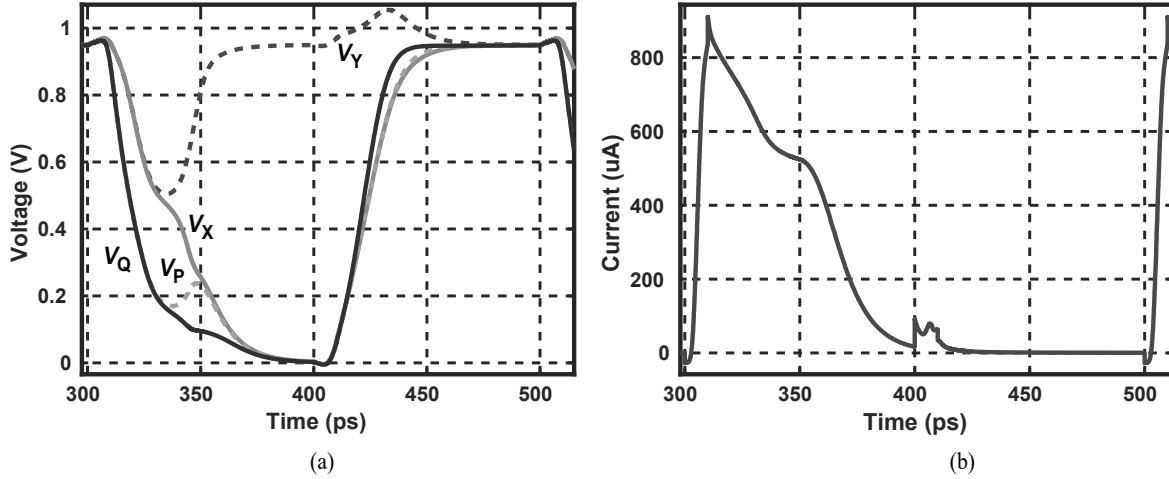


Figure 7.5 The comparator's (a) voltage waveforms, and (b) tail current waveform.

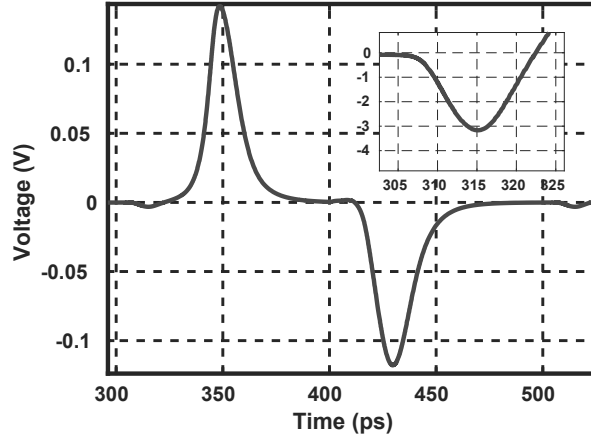


Figure 7.6 Difference between V_P and V_Q as a function of time.

7.4 Offset and Speed Optimization

For the design in Fig. 7.4, we must quantify the input offset contributed by the M_3 - M_4 and M_5 - M_6 pairs. To this end, we place a voltage source equal to $\Delta V_{TH3,4} = 4.4$ mV in series with the gate of M_3 while the other pairs remain matched [Fig. 7.7(a)]. We then adjust the input voltage difference so that the circuit is nearly balanced and $V_X - V_Y$ tends to stay near zero for a relatively long time. With some iteration, we find that $V_{in1} - V_{in2} \approx 1.15$ mV leads to such a behavior [Fig. 7.7(b)]. This suggests that the offset of M_3 and M_4 is divided by a factor of $4.4/1.15 = 3.8$ when referred to the input. The offset standard deviation arising from the M_1 - M_2 and M_3 - M_4 pairs is thus given by $\sqrt{(4.4 \text{ mV})^2 + (1.15 \text{ mV})^2} = 4.5$ mV.

Given the small offset contribution of M_3 and M_4 , we ask whether their widths can be reduced so as to increase the speed. Indeed, if $W_{3,4} = 5 \mu\text{m}$, then $|V_X - V_Y|$ reaches 200 mV in 28 ps. A test similar to that in Fig. 7.7(a) with $\Delta V_{TH3,4} = A_{VTH}/\sqrt{WL} = 6.2$ mV indicates an input contribution of 1.5 mV. That is, the offset rises from 4.5 mV to $\sqrt{(4.4 \text{ mV})^2 + (1.5 \text{ mV})^2} = 4.6$ mV. The minute increase in the offset makes $W_{3,4} = 5 \mu\text{m}$ a more favorable choice.

In the next step of optimization, we turn to M_5 and M_6 in Fig. 7.4 and quantify their offset contribution. With $W_{5,6} = 2.5 \mu\text{m}$, we have $\Delta V_{TH5,6} = 8.8$ mV. Inserting this voltage in series with the gate of M_5 or M_6 and repeating the procedure of Fig. 7.7(a), we arrive at an input contribution equal to 0.9 mV. The total input

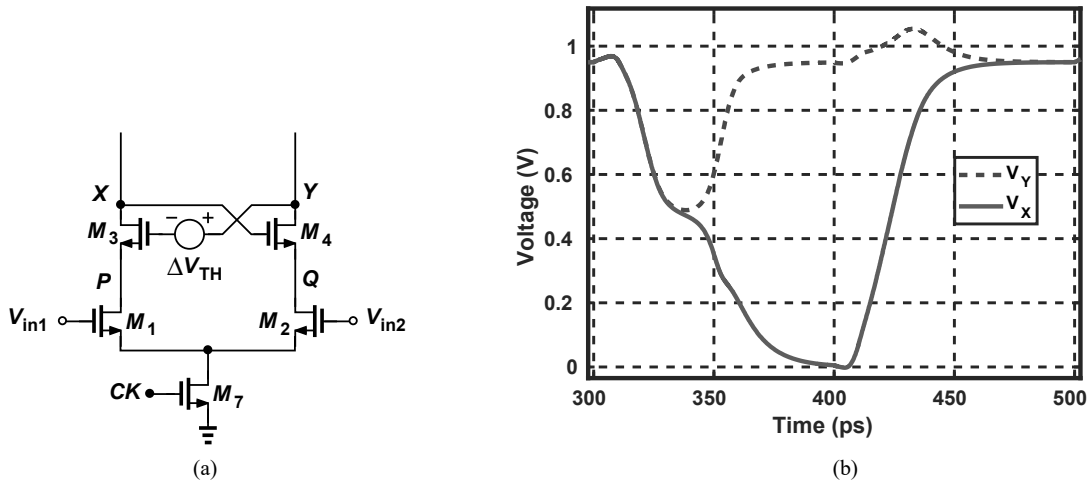


Figure 7.7 (a) Inclusion of mismatch between M_3 and M_4 , and (b) output waveforms when the circuit is nearly balanced.

offset is $\sqrt{(4.4 \text{ mV})^2 + (1.5 \text{ mV})^2 + (0.9 \text{ mV})^2} = 4.7 \text{ mV}$. If we double the widths of M_5 and M_6 , their input contribution is still around 0.9 mV because their larger capacitances lower the voltage gain developed by the circuit before this pair turns on. We therefore retain $W_{5,6} = 2.5 \mu\text{m}$. The offset calculations can be verified through the use of Monte Carlo simulations that incorporate the foundry's mismatch models.

It is possible to increase the comparator's speed by raising the tail current, i.e., by widening M_7 in Fig. 7.4. Plotted in Fig. 7.8 are V_X and V_Y for $W_7 = 2 \mu\text{m}$ and $4 \mu\text{m}$. The time necessary for $|V_X - V_Y|$

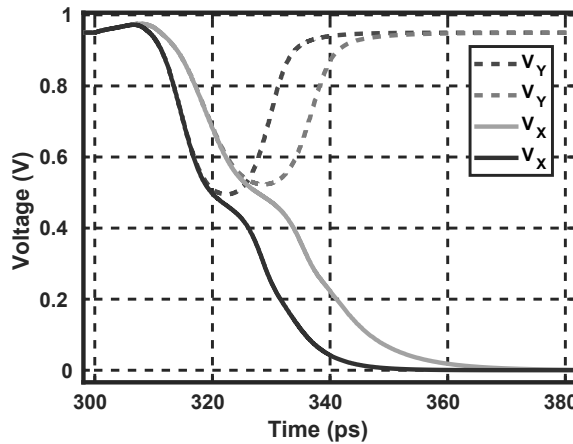


Figure 7.8 Comparator output waveforms for $W_7 = 2 \mu\text{m}$ and $4 \mu\text{m}$.

to reach 200 mV drops from 28 ps to 22 ps. Interestingly, the power consumed in the signal path, $2f_{CK}C_PV_{DD}^2 + f_{CK}C_XV_{DD}^2$, remains fairly constant, but the clock path draws greater power. As explained later, a wider M_7 translates to higher kickback noise.

7.5 Addition of RS Latch

In the precharge mode, the StrongArm comparator's decision is erased and the outputs do not represent a valid logical level, potentially confusing the following stages. To resolve this issue, we insert a reset-set (RS)

latch in the output path. Illustrated in Fig. 7.9, the RS latch can change its state only if M_{11} or M_{12} turns

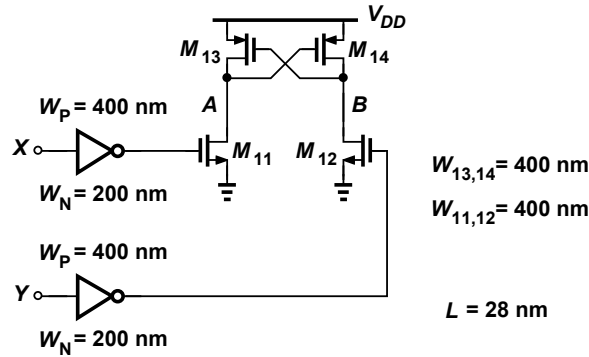


Figure 7.9 An RS latch following the comparator.

on, i.e., when V_X or V_Y falls to zero. This latch then retains the state as the StrongArm circuit enters the precharge mode.

The RS latch's delay proves critical in some ADC architectures and must be minimized. Since the latch operates by pulling one output low, we surmise that M_{13} and M_{14} must be relatively wide. Moreover, if we increase the width of the inverters' PMOS devices, their outputs rise with less delay. We then change $W_{11,12}$ to 800 nm and the inverters' PMOS widths also to 800 nm. As shown in Fig. 7.10, the total delay now drops

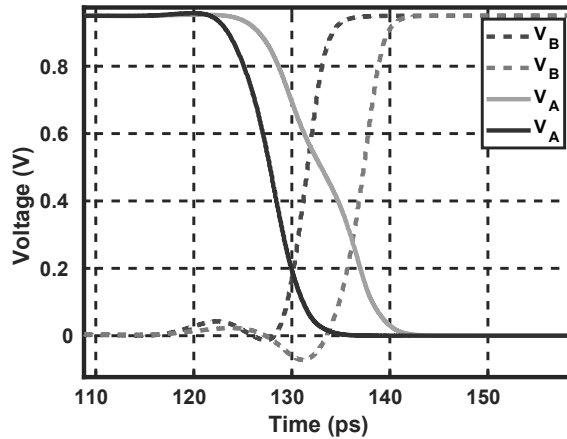


Figure 7.10 RS latch outputs for original and modified designs.

by 6 ps. The overall comparator circuit draws about 0.2 mW at 5 GHz.

7.6 Metastability

In an ADC environment, a comparator senses a random signal at the moment it is clocked. Thus, its input difference can be arbitrarily small. For example, if the signal has a peak-to-peak swing of $2A_0$ with a uniform distribution, then the probability that the difference (positive or negative) presented to the comparator is less than ΔV is equal to $\Delta V/A_0$.

Upon sensing a small difference, a comparator takes some time to generate a well-defined logical output. If it cannot do so in half of the clock period, we say the circuit is “metastable” [Fig. 7.11(a)]. During metastability, the indefinite nature of the comparator outputs can propagate to the subsequent logic, introducing large errors. This issue becomes particularly serious in digital communication systems where the error rate must be extremely low, e.g., less than 10^{-14} .

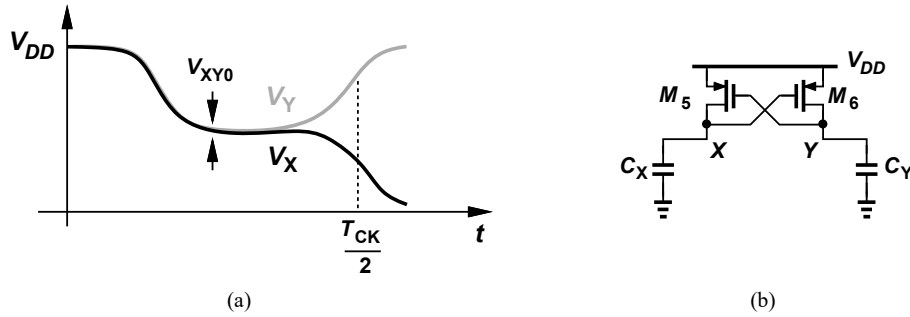


Figure 7.11 (a) Comparator metastability, and (b) behavior of PMOS transistors during this period.

In the StrongArm latch of Fig. 7.4, M_5 and M_6 serve as the primary amplifying circuit during a metastable state. From the model shown in Fig. 7.11(b), we can prove that the positive feedback around the loop is characterized by

$$V_{XY} = V_{XY0} \exp \frac{t}{\tau_{reg}}, \quad (7.2)$$

where V_{XY0} denotes the initial value, and the regeneration time constant, τ_{reg} , is given by $g_{m5,6}/C_X$. Note that this expression is valid only after the initial fall of V_X and V_Y . If V_{XY} is not large enough after $T_{CK}/2$ seconds to write a well-defined state onto the RS latch, an error can occur. Denoting the minimum acceptable value of V_{XY} by V_1 , we require that

$$V_{XY0} \geq V_1 \exp \frac{-T_{CK}/2}{\tau_{reg}}. \quad (7.3)$$

Input differences that yield a V_{XY0} less than this value cause a metastability error. The error rate is therefore proportional to $\exp[-T_{CK}/(2\tau_{reg})]$, underscoring the high impact of τ_{reg} .

We can compute τ_{reg} by measuring $g_{m5,6}$ and C_X in Fig. 7.4. Alternatively, we can obtain τ_{reg} directly from simulations. Consider the two regeneration scenarios depicted in Fig. 7.12, where V_{XY0} is chosen equal

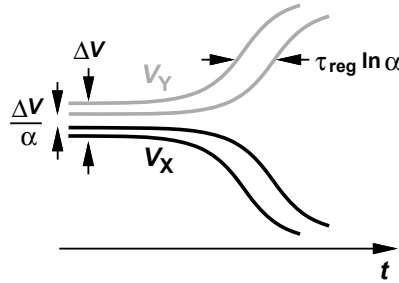


Figure 7.12 Shift of regeneration response for different initial conditions.

to some amount, ΔV , or a smaller amount, $\Delta V/\alpha$. We denote $V_X - V_Y$ for the two cases by V_{XY1} and V_{XY2} , respectively, and write from (7.2):

$$V_{XY2} = \frac{\Delta V}{\alpha} \exp \frac{t}{\tau_{reg}} \quad (7.4)$$

$$= \Delta V \exp(-\ln \alpha) \exp \frac{t}{\tau_{reg}}. \quad (7.5)$$

It follows that

$$V_{XY2} = \Delta V \exp \frac{t - \tau_{reg} \ln \alpha}{\tau_{reg}}. \quad (7.6)$$

Interestingly, V_{XY2} is simply equal to V_{XY1} but shifted by $\tau_{reg} \ln \alpha$. For the design in Fig. 7.4, we select an input difference of 1 mV, 100 μ V, and 10 μ V, arriving at the waveforms shown in Fig. 7.13. The time shift is

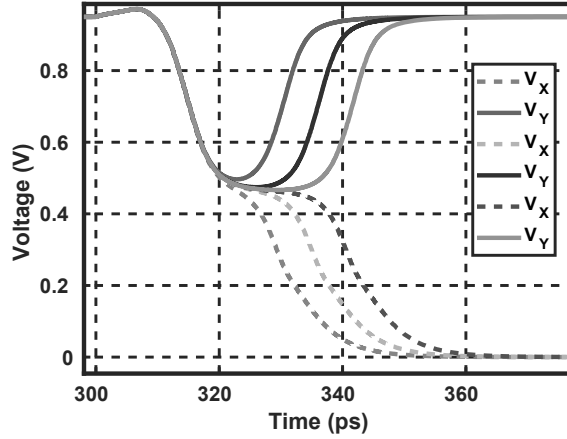


Figure 7.13 Output waveforms of comparator for input differences equal to 1 mV, 100 μ V, and 10 μ V.

in each case is 5.7 ps and implies that $\tau_{reg} = 5.7 \text{ ps} / \ln 10 = 2.5 \text{ ps}$.

Can we reduce τ_{reg} by adjusting the widths of M_5 and M_6 in Fig. 7.4? If $W_{5,6}$ is doubled, these two devices' capacitances double but their transconductance rises by about a factor of $\sqrt{2}$. That is, τ_{reg} decreases only if M_5 and M_6 do not dominate the capacitance at X and Y . In our design, changing $W_{5,6}$ from 2.5 μm to 5 μm increases τ_{reg} slightly.

The foregoing observations prescribe a simple method for estimating the input difference, V_{XY0} in (7.3), that leads to an error. We first simulate the comparator with a moderate value for $V_{in1} - V_{in2}$, e.g., 1 mV, and find the delay, e.g., 22 ps. We also recognize that (a) reducing $V_{in1} - V_{in2}$ by a factor of 10^n shifts the response by $n\tau_{reg} \ln 10$, and (b) if the shift exceeds $T_{CK}/2 - 22 \text{ ps}$, an error is likely to occur. In our design,

$$n\tau_{reg} \ln 10 \approx \frac{T_{CK}}{2} - 22 \text{ ps} \quad (7.7)$$

$$\approx 78 \text{ ps} \quad (7.8)$$

and hence $n \approx 13.5$. It follows that input differences less than $1 \text{ mV}/10^{13.5}$ may generate errors.

We should remark that simulating a comparator with very small input differences, e.g., 1 fV, requires minimizing all sources of asymmetry in the circuit and in the simulation tool. Specifically, the presence of the RS latch in Fig. 7.9 does lead to a slight asymmetry in the StrongArm circuit. Suppose the stored state is $V_A = 0$ and $V_B = V_{DD}$. As a result, the gate input capacitances of M_{11} and M_{12} are slightly different. This means that the capacitances seen at the inverters' inputs are also slightly unequal (due to the Miller effect of their gate-drain parasitics). This deterministic imbalance causes the StrongArm latch to favor one logical output for very small input differences. We therefore disconnect the RS latch for such simulations. Alternatively, we can short A and B to V_{DD} so as to maintain the loading presented to the inverters.

Another metastability simulation issue relates to the simulator's accuracies. In Cadence, we set three parameters as follows: $\text{reltol} = 10^{-6}$, $\text{vabstol} = 10^{-6}$, and $\text{iabstol} = 10^{-12}$.

7.7 Input-Referred Noise

The standard method of computing the output noise and dividing it by the gain does not apply to comparators because they produce a digital output. As explained in [2], we perform a transient noise simulation so that the comparator's time-domain decision is randomly affected by the noise of its constituent devices. We first set $V_{in1} - V_{in2}$ to zero [Fig. 7.14(a)] and ensure that the logical output assumes a value of ZERO or ONE with

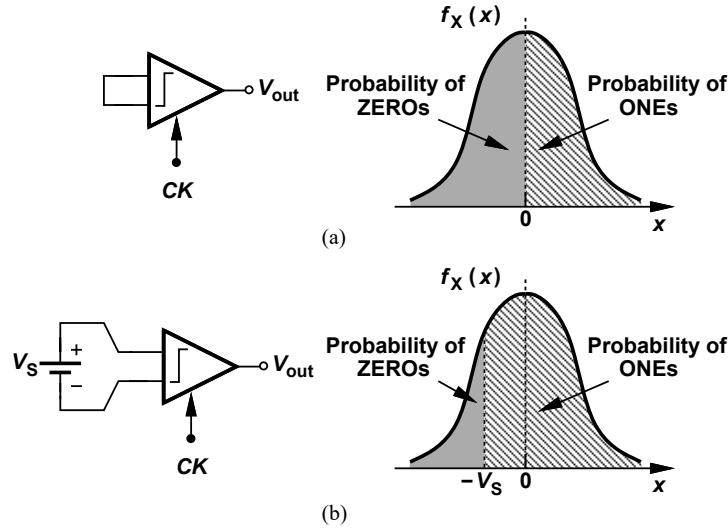


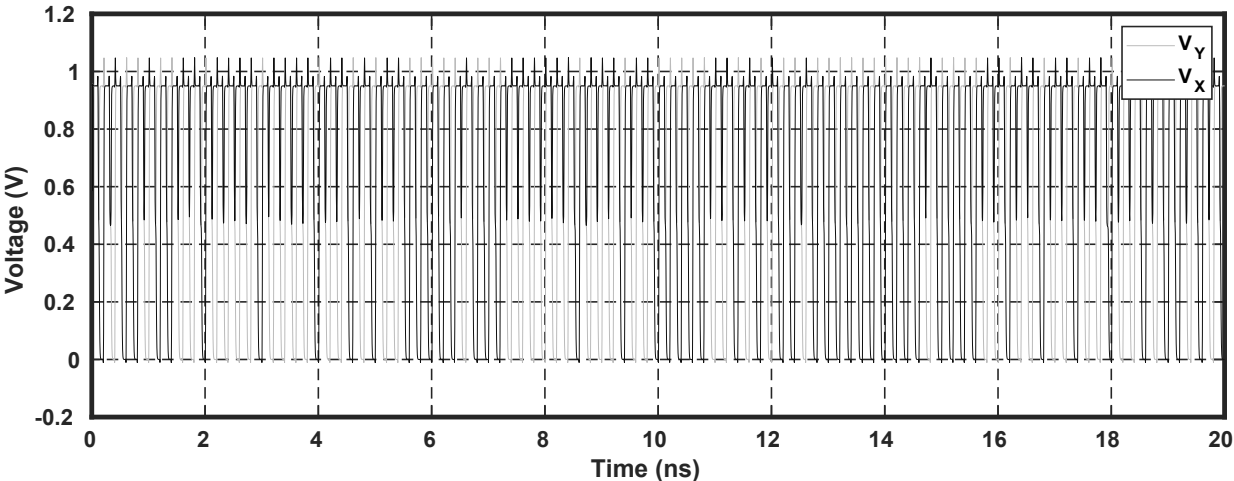
Figure 7.14 (a) Perfectly balanced comparator generates ONES and ZEROs with equal probabilities, and (b) a finite input imbalance skews the decisions.

equal probabilities. Plotted in Fig. 7.15(a) are V_X and V_Y in this case for 100 clock cycles. We observe that V_X goes to zero about 50 times.

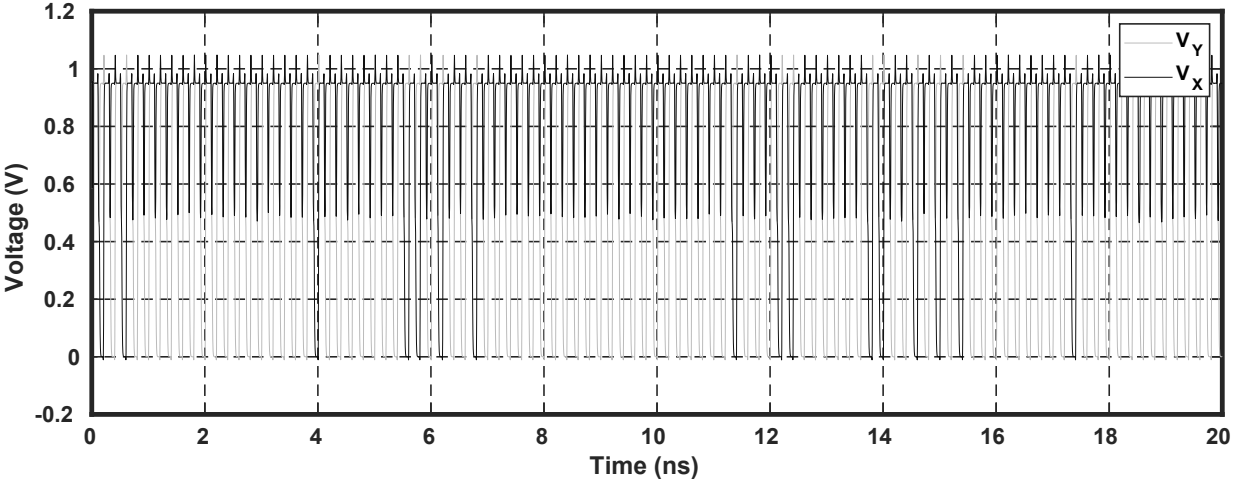
Next, we select a small, constant value for $|V_{in1} - V_{in2}|$ so as to skew the decisions [Fig. 7.14(b)]. We recall that the area under a Gaussian distribution from $-\sigma$ to $+\sigma$ is equal to 68% and hence that from $-\infty$ to $-\sigma$ is $100\% - (34\% + 50\%) = 16\%$. Thus, if V_S is so chosen as to reduce the probability of ZEROs to 16%, then $V_S = \sigma$, which is also the total rms noise referred to the input. After a few iterations, we observe the waveforms in Fig. 7.15(b), where V_X goes to zero about 16 times for $V_S = 0.31$ mV. The comparator's input rms noise is approximately equal to this value. For greater precision, we can run the two simulations for a larger number of clock cycles.

7.8 Kickback Noise

The StrongArm latch draws large transient currents from its inputs during switching. Called the “kickback noise,” this phenomenon proves undesirable if it affects the comparator's own decision or corrupts the input voltage while it is sensed by other circuits. For example, in a flash ADC, all of the comparators generate kickback noise while one must make a critical decision. Figure 7.16 plots the kickback noise currents of our design when $V_{in1} - V_{in2} = 1$ mV. The clock begins to rise at $t = 300$ ps. We recognize that the two exhibit both common-mode and differential components. The former are objectionable if they flow through unequal source impedances, and the latter prove problematic generally. Kickback noise trades with the dimensions of the input transistors and hence with the offset voltage. But the timing of this noise determines whether it has an adverse effect on the performance. For example, we expect the noise around $t = 300$ ps in Fig. 7.16 to be more serious as it coincides with the comparator's decision time.



(a)



(b)

Figure 7.15 Comparator outputs for (a) perfect balance, and (b) with an input difference equal to 0.31 mV.

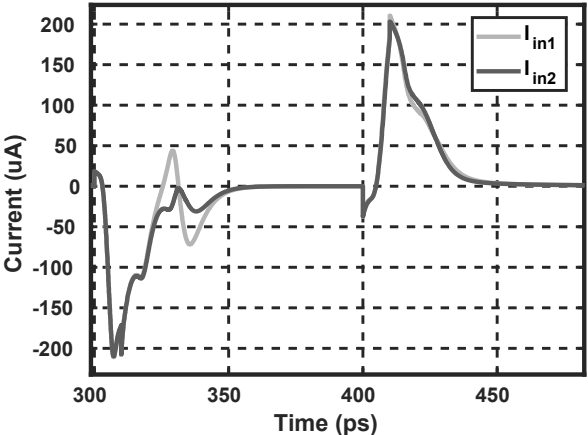


Figure 7.16 Input kickback noise currents.

References

- [1]. M. J. M. Pelgrom, A. C. J. Duinmaijer, and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 1433-1440, Oct. 1989.
- [2]. B. Razavi, "The StrongARM latch," *IEEE Solid-State Circuits Magazine*, issue 2, pp. 12-17, Spring 2015.

Problems

- 7.1. Recall from Chapter 6 that the offset of the StrongArm comparator shown in Fig. 7.1 can be canceled by attaching programmable capacitors to nodes P and Q . Explain how this modification affects the circuit's dynamic offset.
- 7.2. Using Eq. (7.1), explain what happens to the input-referred offset of the comparator in Fig. 7.1 if the widths of all of the transistors are scaled up by a factor of α . How does the circuit's average power consumption change?
- 7.3. As mentioned in Problem 7.1, the offset can be canceled by programmable capacitors. Explain how attaching capacitors to P and Q in Fig. 2.14 affects the offset contributed by M_3 and M_4 .
- 7.4. Sketch the waveforms in Fig. 7.1 if M_5 and M_6 are weak devices.
- 7.5. How much is the voltage at the source of M_1 and M_2 in Fig. 7.4 after the comparator has completed its decision?
- 7.6. In Fig. 7.4, $V_{in1} = 600$ mV and $V_{in2} = 590$ mV. We clock the comparator and examine the voltage at the source of M_1 and M_2 after a few hundred picoseconds. Viewing M_1 and M_2 as a differential pair, a student reasons that this voltage should be approximately equal to the input CM level minus the threshold voltage of M_1 and M_2 since these transistors carry a negligible current. Explain the flaw in this argument.
- 7.7. The tail current waveform in Fig. 7.5(b) has a certain area representing the total charge that M_7 draws from the circuit. If we examine the comparator around $t = 390$ ps, from which capacitances in the circuit has this charge been drawn?
- 7.8. From the current waveform in Fig. 7.5(b), can you predict the voltage waveform at the drain of M_7 ? (Hint: This transistor resides in the deep-triode region.)
- 7.9. We have observed a residual dynamic offset of $100 \mu\text{V}$ between V_P and V_Q in Fig. 7.6. How much is the input-referred value? How does this value change if we halve the widths of S_1 - S_4 ?
- 7.10. The waveforms in Fig. 7.6 indicate an initial voltage gain of 3.15 for the input differential pair. What is the smallest capacitance that the circuit can tolerate at P and Q in Fig. 7.4 if the kT/C noise from these two nodes should remain below $0.5 \text{ mV}_{\text{rms}}$ when referred to the input?
- 7.11. Explain why M_3 and M_4 in Fig. 7.7(a) contribute (almost) no offset if the capacitance at P and Q is zero and M_1 and M_2 reside in saturation.
- 7.12. We examine the circuit of Fig. 7.4 immediately after CK rises, noting that M_3 - M_6 are off. Should we include the entire C_{GS3} and C_{GS4} in the capacitance at P and Q ? (Hint: Draw the equivalent circuit for M_3 - M_6 while they are off, including only their gate-source capacitances for simplicity.)
- 7.13. Figure 7.8 suggests that making M_7 wider in Fig. 7.4 improves the speed. If we continue to increase W_7 , at what point do we reach diminishing returns? (Hint: How much voltage at the source of M_1 and M_2 yields the highest current immediately after CK rises?)
- 7.14. To reduce the delay, a student decides to omit the inverters in Fig. 7.9 and connect the latch directly to a StrongArm comparator. Explain why this arrangement fails in the precharge mode.
- 7.15. In the previous problem, the student still omits the inverters but also changes M_{11} and M_{12} to PMOS devices and M_{13} and M_{14} to NMOS devices. What criteria govern the device widths in this case?
- 7.16. Explain why the latch in Fig. 7.9 loses its state if X and Y remain high for a long time (i.e., if the comparator is clocked at low rates).
- 7.17. Explain why V_A and V_B in Fig. 7.10 cross around 200 mV rather than around $V_{DD}/2$. (Hint: The switching sequence proceeds with, for example, M_{11} turning on and then M_{14} .)
- 7.18. Capacitances C_X and C_Y in Fig. 7.11 arise from M_5 , M_6 , and other transistors connected to these nodes (including those in the next stage). Student A says increasing $W_{3,4}$ yields a higher transconductance and faster regeneration. Student B says increasing $W_{3,4}$ raises C_X and C_Y , making regeneration slower. Who is correct?
- 7.19. It is possible to plot the waveforms of Fig. 7.12 on a logarithmic scale for the vertical axis. Sketch such waveforms and explain how the two cases of $V_{XY0} = \Delta V$ and $V_{XY0} = \Delta V/\alpha$ appear in such a plot.
- 7.20. Suppose we run the comparator of Fig. 7.4 at $f_{CK} = 10$ GHz. From the metastability analysis, determine the minimum input difference that guarantees error-free operation.
- 7.21. The analysis leading to Fig. 7.15 implies an input-referred noise of 0.31 mV. Returning to the StrongArm comparator noise analysis in Chapter 6, do you expect this value to change if the circuit is clocked at 2.5 GHz rather than at 5 GHz?

- 7.22.** Consider the circuit of Fig. 7.4, assuming $V_{in1} = 600$ mV and $V_{in2} = 590$ mV or $V_{in1} = 700$ mV and $V_{in2} = 690$ mV. Which case yields a faster response? Why?

General DAC Concepts

Digital-to-analog converters (DACs) find application either as stand-alone circuits or as a building block of ADCs. A DAC delivering an image or a video to a display exemplifies the former. Some of the design principles apply to both types of DACs but some do not.

In this chapter, we develop a basic understanding of DACs and their performance parameters. The foundation thus formed will prove essential to our studies of specific DAC architectures in Chapters 9 to 11.

8.1 Basic Concepts

A DAC converts a digital input, D_{in} , to an analog output, A_{out} (Fig. 8.1):

$$A_{out} = \alpha D_{in}. \quad (8.1)$$

We denote the output (analog) LSB by ΔA_{LSB} and the “full scale” (the maximum value) by A_{FS} . The factor

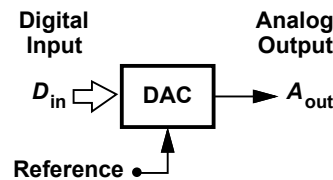


Figure 8.1 Conceptual DAC.

α is necessary to convert the dimensionless D_{in} to a proper dimension for A_{out} . For example, if A_{out} is a current quantity, we have

$$A_{out} = I_{REF} D_{in}, \quad (8.2)$$

where I_{REF} is a dc reference current that defines the DAC’s output LSB size. Thus, DACs must receive a “reference” that defines their output dimension and full scale. For a DAC with a resolution of N bits, the full scale occurs with $D_{in} = 11 \cdots 1$ and is equal to $(2^N - 1) \times \text{LSB}$. We denote 2^N by m , i.e., the DAC output has m levels. In the 3-bit example depicted in Fig. 8.2, as the binary input, $D_{in} = D_3 D_2 D_1$, varies from 000 to 111, the DAC generates an output current ranging from zero to $7I_{REF}$. Note that the characteristic jumps seven times.

We should remark that the input-output characteristic in Fig. 8.2 consists of only discrete *points* as there are no analog values between the input codes. Nonetheless, we typically draw the dashed characteristic instead

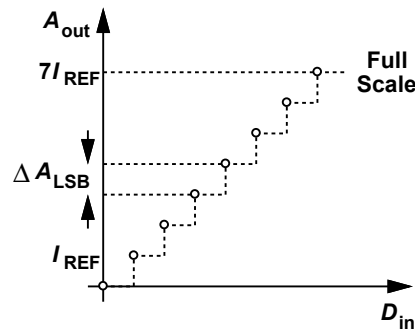


Figure 8.2 DAC input-output characteristic.

for better visualization of nonidealities. In an ideal DAC, these points follow a straight line that passes through the origin. The analog output can be a voltage or a current.¹

As an example, let us construct a simple 3-bit DAC employing current sources. Shown in Fig. 8.3, the circuit incorporates three current sources controlled by the binary input code. For $D_3D_2D_1 = 001$, $I_{out} = I_{REF}$; for $D_3D_2D_1 = 010$, $I_{out} = 2I_{REF}$; etc. That is, this “current-mode” DAC provides eight output levels ranging from zero to $7I_{REF}$.

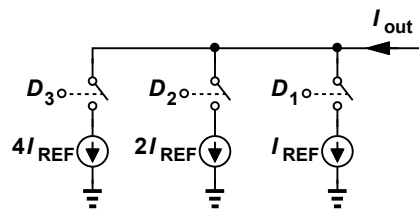


Figure 8.3 3-bit DAC example.

Example 8.1

Suppose the LSB current source in Fig. 8.3 incurs an error of ΔI while the others remain equal to their nominal values. How is the input-output characteristic affected?

Solution

For $D_3D_2D_1 = 001$, we have $I_{out} = I_{REF} + \Delta I$, but for $D_3D_2D_1 = 010$, we obtain $I_{out} = 2I_{REF}$. As shown in Fig. 8.4, I_{out} departs by ΔI for the input codes that activate the LSB current source. The input-output characteristic is therefore “distorted.” The reader can repeat this example if the next LSB current source, i.e., $2I_{REF}$, incurs an error.

8.2 DAC Performance Parameters

A DAC serving in a larger system must meet a number of performance requirements. We define in this section the parameters that prove critical in DAC design. This section is relatively long, but the reader is encouraged to be patient as many fundamental concepts are introduced here.

¹One can argue that capacitor DACs (Chapter 11) operate with charge quantities, but their final output is still a voltage.

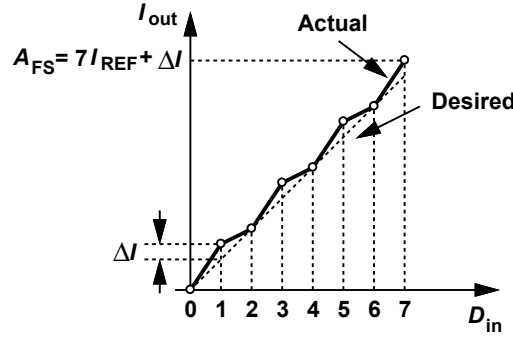


Figure 8.4 Effect of mismatch on DAC characteristic.

8.2.1 Static Parameters

The input-output characteristic of Fig. 8.2 in reality can suffer from five imperfections: offset, gain error, differential nonlinearity (DNL), integral nonlinearity (INL), and non-monotonicity. We wish to define each and determine whether and how it arises in the simple DAC of Fig. 8.3. In the analysis of DACs, we often draw a straight line between the end points of the characteristic. In Fig. 8.4, for example, such a line would start at the origin and end at $(D_{in} = 7, A_{FS} = 7I_{REF} + \Delta I)$.

Offset Offset is defined as the vertical shift in the characteristic with respect to the ideal case [Fig. 8.5(a)]. As explained in Chapter 17, this offset proves undesirable if the DAC serves in an ADC. For stand-alone

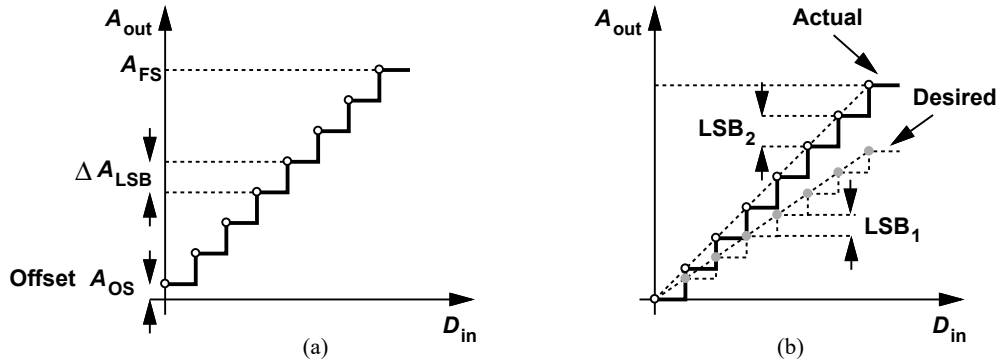


Figure 8.5 (a) Offset in a DAC, and (b) gain error in a DAC.

DACs, on the other hand, offset is less critical. For example, in the presence of offset, a DAC delivering a video signal to a display would simply make all of the pixels slightly brighter or darker. The DAC in Fig. 8.3 does not exhibit an offset because $I_{out} = 0$ for $D_3D_2D_1 = 0$.

Example 8.2

Given a DAC characteristic such as that in Fig. 8.5(a), how do we define the analog LSB?

Solution

We write $\Delta A_{LSB} = (A_{FS} - A_{OS})/(m - 1)$ because there are $m - 1$ jumps between these two levels. In a well-designed DAC, $A_{OS} \ll A_{FS}$, $m \gg 1$, and hence $\Delta A_{LSB} \approx A_{FS}/m$.

Gain Error Gain error is the deviation of the slope of the line passing through the end points from its nominal value [Fig. 8.5(b)]. In this simple case, all of the output steps are greater or less than the ideal value by the same factor, e.g., $\text{LSB}_2 = 1.1 \text{ LSB}_1$. Like offset, this effect is critical only if the DAC operates in an ADC environment. In Fig. 8.2, gain error occurs if all of the current sources are proportionally greater or less than their ideal values. Let us consider the implementation shown in Fig. 8.6, where the current sources are copied from I_{REF} . We observe that an error in I_{REF} translates to gain error for the DAC. In the absence of

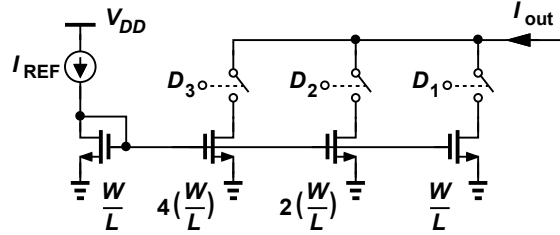


Figure 8.6 DAC implementation example.

channel-length modulation and mismatches,

$$I_{out} = 4I_{REF}D_3 + 2I_{REF}D_2 + I_{REF}D_1 \quad (8.3)$$

$$= (4D_3 + 2D_2 + D_1)I_{REF}. \quad (8.4)$$

Thus, I_{REF} directly scales I_{out} .

Differential Nonlinearity Offset and gain error are tolerable in some applications because they do not introduce nonlinearity.² In other words, the output values in Figs. 8.5(a) and (b) still lie on a straight line. On the other hand, we have observed in Example 8.1 and Fig. 8.4 that the characteristic can become nonlinear as a result of circuit imperfections. We therefore need metrics that represent DAC nonlinearity. The first metric is “differential nonlinearity,” defined as the deviation in each output step size from the nominal LSB value, ΔA_{LSB} . As illustrated in Fig. 8.7, we measure the DNL by incrementing D_{in} from D_j to D_{j+1} , finding the

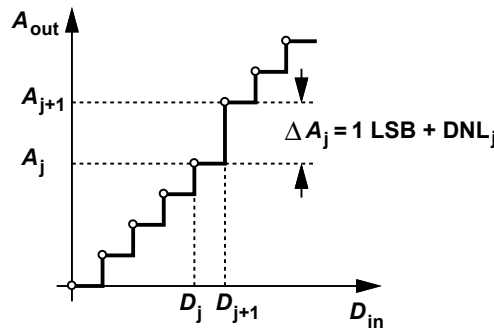


Figure 8.7 DNL in a DAC.

change in the output, ΔA_j , and subtracting the nominal LSB from it:

$$\text{DNL}_j = \Delta A_j - \Delta A_{\text{LSB}}. \quad (8.5)$$

²Strictly speaking, a system does not satisfy linear superposition in the presence of an offset (why?), but we still consider the DAC in Fig. 8.5(a) linear as far as *changes* in D_{in} are concerned.

(As mentioned above, ΔA_{LSB} is approximately equal to the full scale divided by m .) The DNL has a unit of volts or amperes. We prefer to normalize it to 1 LSB.

Example 8.3

Compute and plot the DNL for the DAC characteristic shown in Fig. 8.4. Assume $\Delta A_{\text{LSB}} = I_{\text{REF}}$ even though $A_{\text{FS}}/7$ is slightly greater.

Solution

As D_{in} goes from 000 to 001, I_{out} changes by $I_{\text{REF}} + \Delta I$, yielding $\text{DNL}_0 = \Delta I$. In the next code transition, I_{out} changes by $I_{\text{REF}} - \Delta I$ (why?), exhibiting $\text{DNL}_1 = -\Delta I$. This pattern repeats and leads to the “DNL profile” depicted in Fig. 8.8. Note that there is no DNL_7 because the DAC does not provide a transition at $D_{\text{in}} = 7$. This profile must be divided by I_{REF} so as to express the DNL in LSB. The reader is encouraged to perform this exercise if the second LSB current source, $2I_{\text{REF}}$, in Fig. 8.3 has an error.

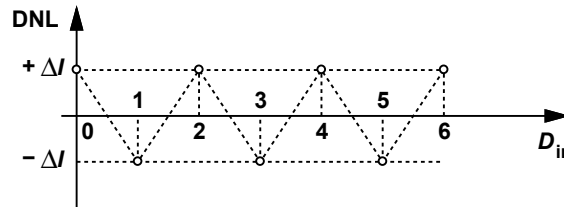


Figure 8.8 DNL plot for a DAC.

We should make a remark about gain error and DNL. The gain error illustrated in Fig. 8.5(b) arises from “linear scaling” of all of the output steps. But that need not be the case. For example, the characteristic in Fig. 8.4 does suffer from gain error because, as illustrated in Fig. 8.9, a straight line passing through its end points (line OA) exhibits a slope different from the nominal (line OB). That is, DNL introduces gain error in this case.

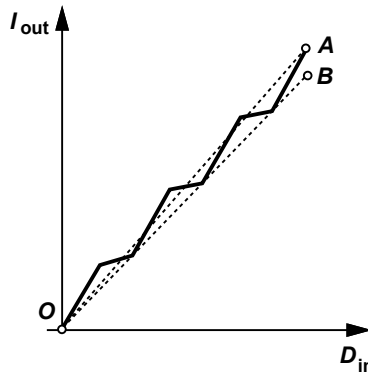


Figure 8.9 Gain error due to DNL.

Example 8.4

The 3-bit DAC of Fig. 8.3 exhibits the characteristic shown in Fig. 8.10(a). Explain what type of circuit imperfection causes this behavior and plot the DNL profile. Does the DAC suffer from gain error?

Solution

We observe that only the “major” code transition, namely, that from 3 to 4, displays an error. Thus, the

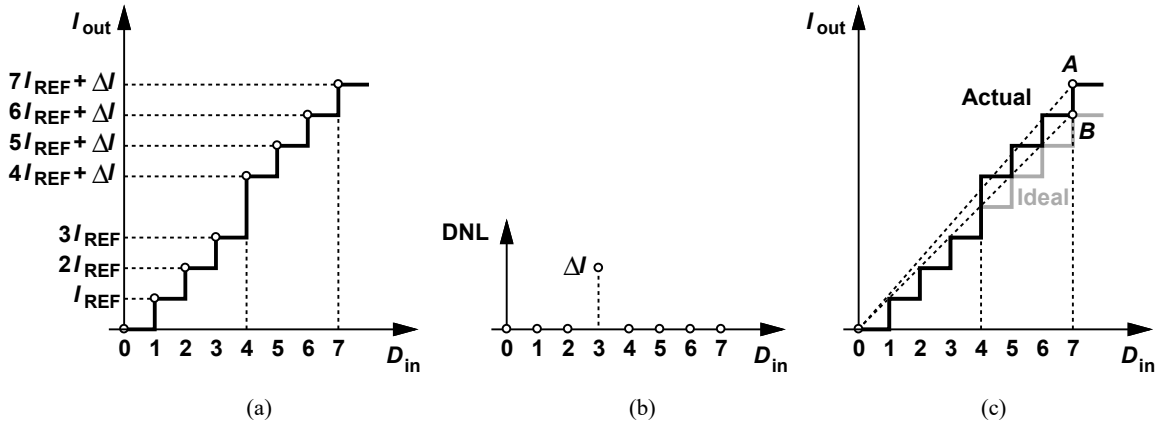


Figure 8.10 (a) Output error for a major code transition, (b) corresponding DNL profile, and (c) resulting gain error.

MSB current source in Fig. 8.3 is equal to $4I_{REF} + \Delta I$ while the other two remain at their nominal values. Figure 8.10(b) plots the DNL profile. Passing a straight line between the end points, we conclude that the ΔI mismatch also introduces gain error [Fig. 8.10(c)].

Integral Nonlinearity Another source of distortion in DACs is “integral nonlinearity,” defined as the deviation of the characteristic from a straight line passing through the end points. Illustrated in Fig. 8.11(a), the INL is zero at $D_{in} = 0$ and at $D_{in} = 11 \dots 11$, reaching a maximum for some intermediate value of D_{in} . Subtracting the straight line from the characteristic yields the INL profile [Fig. 8.11(b)]. The INL can be expressed in three forms: (1) in volts or amperes; (2) as a percentage (normalized to the full scale); or (3) in LSB (normalized to the analog LSB value).

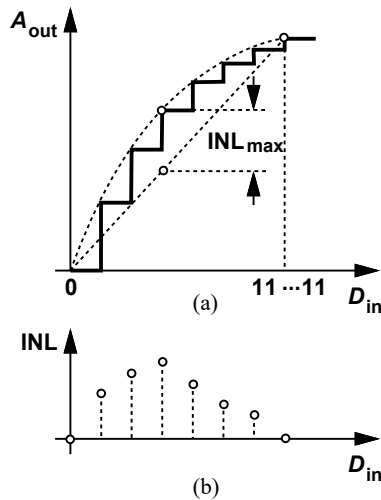


Figure 8.11 (a) INL in a DAC, and (b) corresponding INL profile.

It is important not to confuse DNL with INL. The former signifies how close to the ideal each output *step* is, representing a “local” effect, whereas the latter corresponds to a “global” curvature in the characteristic.

Example 8.5

Plot the INL profile for the characteristic in Fig. 8.4.

Solution

Passing a straight line through the end points [Fig. 8.12(a)], we note that output levels A_1 , A_3 , A_5 , and A_7 incur less and less error, whereas A_2 , A_4 , and A_6 incur more and more. Thus, the INL profile emerges as shown in Fig. 8.12(b). The reader can obtain the values of the INL errors in this plot.

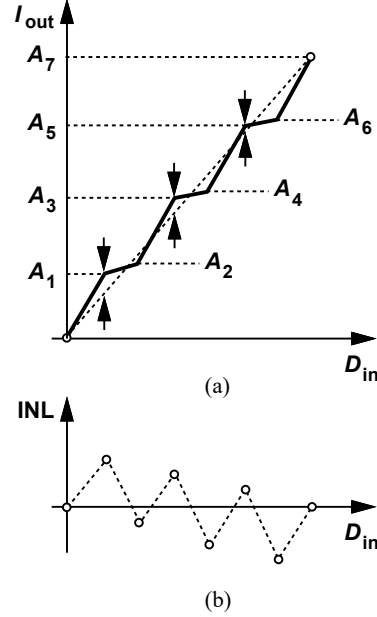


Figure 8.12 (a) DAC characteristic along with a straight line passing through its end points, and (b) corresponding INL profile.

Relation Between INL and DNL The reader may wonder if integral nonlinearity is indeed equal to the integral (summation) of differential nonlinearity. Let us write from Fig. 8.7 and Eq. (8.5)

$$DNL_j = \Delta A_j - \Delta A_{LSB} \quad (8.6)$$

$$= A_{j+1} - A_j - \Delta A_{LSB}. \quad (8.7)$$

We can express any output level A_n as

$$A_n = \sum_{j=1}^n (A_j - A_{j-1}) \quad (8.8)$$

$$= \sum_{j=1}^n (DNL_{j-1} + \Delta A_{LSB}) \quad (8.9)$$

$$= \sum_{j=1}^n DNL_{j-1} + n\Delta A_{LSB}, \quad (8.10)$$

where we have assumed $A_0 = 0$, i.e., a zero offset. Note that $n\Delta A_{\text{LSB}}$ represents the ideal output. Can we conclude that this quantity

$$A_n - n\Delta A_{\text{LSB}} = \sum_{j=1}^n \text{DNL}_{j-1} \quad (8.11)$$

is equal to the INL? To answer this question, we turn to the ideal and actual characteristics shown in Fig. 8.13(a), where the gain error is zero (as the two meet at $D_{\text{in}} = 00 \cdots 0$ and $D_{\text{in}} = 11 \cdots 1$). In this case,

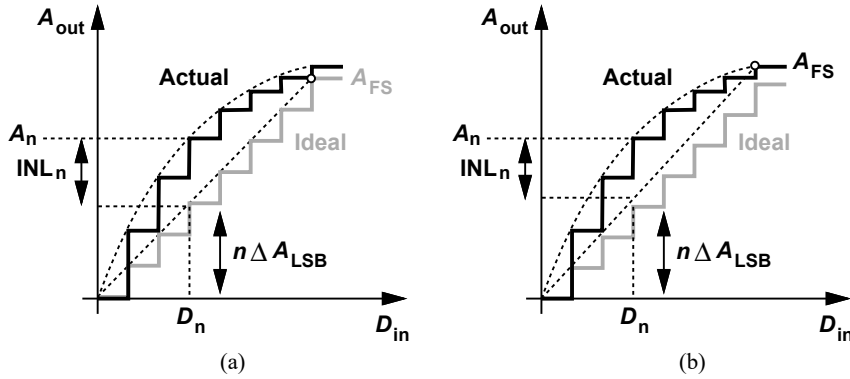


Figure 8.13 DAC with (a) a zero gain error, and (b) a finite gain error.

$\text{INL}_n = A_n - n\Delta A_{\text{LSB}}$ and hence

$$\text{INL}_n = \sum_{j=1}^n \text{DNL}_{j-1}. \quad (8.12)$$

Now, consider the situation depicted in Fig. 8.13(b), where the gain error is not zero. We observe that (8.11) is not valid here unless ΔA_{LSB} is adjusted according to the gain error.

Equation (8.12) can be applied to the characteristic in Fig. 8.13(b) by a simple modification: we define the output LSB as the actual full scale, A_{FS} , divided by 2^N . The proof is left as an exercise for the reader.

An interesting observation proves helpful in practice. Suppose the INL profile has the general shape shown in Fig. 8.14(a), starting and ending at zero. Since this profile is the integral of the DNL, the latter must behave as depicted in Fig. 8.14(b), i.e., it must be positive for one part of the range and negative for another part.

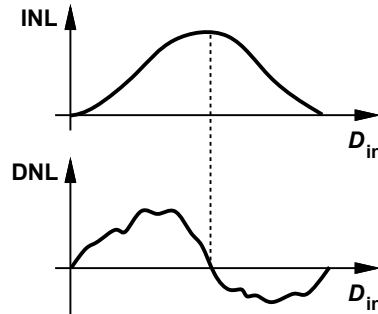


Figure 8.14 (a) An INL profile, and (b) its corresponding DNL profile.

Nonmonotonicity In the DAC characteristic examples studied thus far, the output always increases when the input increases. We say these DACs are “monotonic.” However, under certain conditions, the characteristics become nonmonotonic. For example, suppose the DAC of Fig. 8.3 suffers from a gross error in the MSB

current source and is equal to $4I_{REF} - \Delta I$, where $\Delta I = 1.5I_{REF}$. Then, as shown in Fig. 8.15(a), I_{out} falls from $3I_{REF}$ to $2.5I_{REF}$ as D_{in} rises from 011 to 100. We also observe a DNL of $-1.5I_{REF}$ [Fig. 8.15(b)].

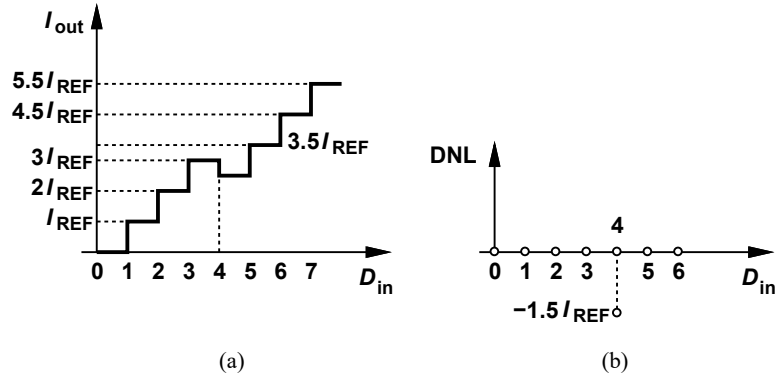


Figure 8.15 (a) Nonmonotonicity in a DAC, and (b) the corresponding DNL profile.

Nonmonotonicity causes the *polarity* of the DAC gain (i.e., the slope) to change; in Fig. 8.15(a), the gain changes from positive to negative around $D_{in} = 4$. As such, it is undesirable in some applications.³

8.2.2 Dynamic Parameters

In addition to the five imperfections studied in the previous section, DACs are also characterized by a number of “dynamic” effects, namely, settling time, “glitch impulse area,” “signal-to-(noise + distortion) ratio” (SNDR), and “spurious-free dynamic range” (SFDR).

Settling Time The conversion speed of DACs is primarily determined by their output settling time, t_s . As depicted in Fig. 8.16, we allow the digital input to jump from $00 \dots 00$ to $11 \dots 11$ and observe the analog output. When the output enters an acceptable error band around the final value, we say the DAC has settled. The desired error band is typically equal to ± 0.5 LSB around the final value.

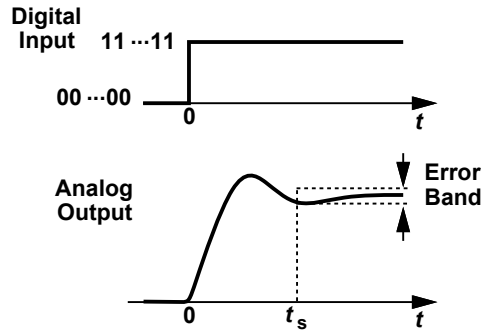


Figure 8.16 DAC output settling.

Example 8.6

The DAC of Fig. 8.3 drives a load resistor and incurs a load capacitance [Fig. 8.17(a)]. Determine the settling time for a resolution of 3 bits.

³Nonmonotonicity also makes calibration difficult as it precludes a one-to-one correspondence between the input and the output.

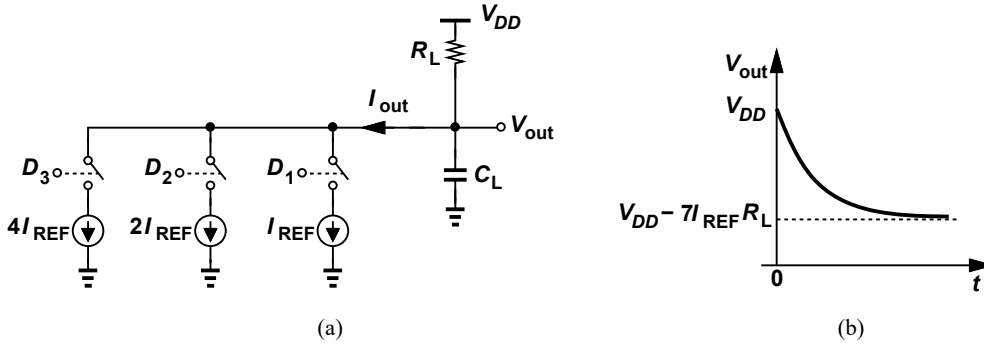


Figure 8.17 (a) Simple DAC, and (b) its settling behavior.

Solution

When the digital input jumps from 000 to 111, V_{out} begins from V_{DD} and travels toward $V_{DD} - 7I_{REF}R_L$ [Fig. 8.17(b)]. For V_{out} to reach within 0.5 LSB of its final value, we have

$$V_{DD} - 7I_{REF}R_L + 7I_{REF}R_L \exp \frac{-t_s}{R_L C_L} = V_{DD} - 7I_{REF}R_L + \frac{I_{REF}R_L}{2}, \quad (8.13)$$

where the right-hand side denotes the final value plus 0.5 LSB. Thus,

$$t_s = R_L C_L \ln 14 \quad (8.14)$$

$$\approx 2.63 R_L C_L. \quad (8.15)$$

DACs with higher resolutions require a longer time to enter the 0.5-LSB error band.

Glitch Impulse Area During switching from one value to another, the output of a DAC may exhibit a glitch, a critical issue in stand-alone DACs. For example, such a glitch is readily visible if a DAC drives a display. Let us return to the DAC of Fig. 8.3 and investigate how a glitch can arise in I_{out} . Suppose the digital input bits travel on wires and experience different delays. For example, the MSB current source arrives slightly later than the other two. Now, if the input changes from 100 to 011 (Fig. 8.18), then $2I_{REF}$ and I_{REF} turn

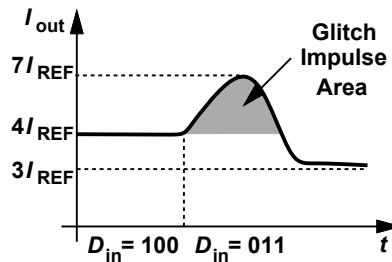


Figure 8.18 Glitch at a DAC output.

on and $4I_{REF}$ turns off, but with a delay. Thus, I_{out} goes from $4I_{REF}$ to $7I_{REF}$ and then to $3I_{REF}$. The resulting glitch is quantified by the area under it.⁴ In DACs used within ADCs, the glitch is less critical as it only increases the settling time.

⁴This is often called the “glitch energy” even though it has a dimension of voltage $\times$ time.

SNDR The measurement of the DAC settling time proves difficult in practice: the analog output must be sensed by a high-speed oscilloscope that itself has a faster settling speed. For this reason, we characterize the speed of DACs by another approach. We apply a digital sinusoid to the input and measure the harmonic content of the analog output (Fig. 8.19). This is similar to the characterization of sampling circuits (Chapter 3). The total harmonic distortion (THD) is obtained by dividing the sum of the powers of the higher harmonics

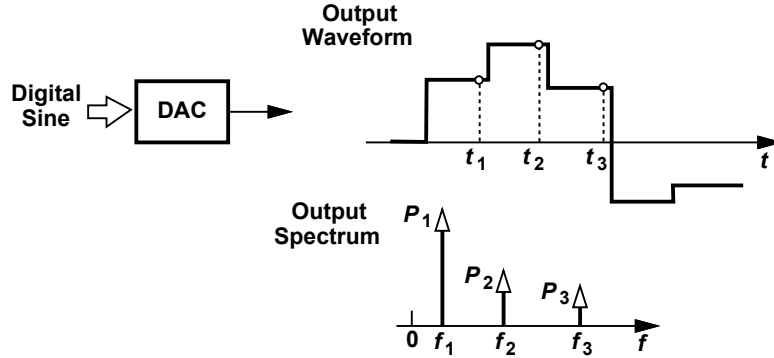


Figure 8.19 DAC SNDR measurement.

by the power of the fundamental:

$$\text{THD} = \frac{P_2 + P_3 + \cdots}{P_1}. \quad (8.16)$$

The sinusoid frequency can be swept from low to high values so as to quantify the DAC's dynamic behavior.

If the output noise is also critical, we compute the SNDR as the power of the fundamental divided by the sum of the distortion power and the noise power, P_{noise} :

$$\text{SNDR} = \frac{P_1}{P_2 + P_3 + \cdots + P_{\text{noise}}}. \quad (8.17)$$

We should point out that the entire output waveform in Fig. 8.19 matters if the DAC operates as a stand-alone circuit, e.g., for driving a display. On the other hand, if the DAC serves within an ADC, only the *settled* values of the waveform, e.g., those at t_1 , t_2 , and t_3 prove critical.

SFDR The distortion produced by a DAC can alternatively be characterized by simply dividing the power of the fundamental by the power of the *largest* unwanted component in the output spectrum, P_{max} . The result is called the SFDR:

$$\text{SFDR} = \frac{P_1}{P_{\text{max}}}. \quad (8.18)$$

We also write $\text{SFDR} = 10 \log(P_1/P_{\text{max}}) = 10 \log P_1 - 10 \log P_{\text{max}}$.

The SFDR merits two remarks. First, the largest unwanted component is typically a harmonic of the input frequency, but not always. As shown in Fig. 8.20, it is possible that a “spur,” i.e., a component having a frequency $f_a \neq (\text{integer}) \times f_1$, appears at the output and happens to have the highest power. We study the source of spurs in Section 8.4. Second, in typical DAC designs, the SFDR improves if the output swing is reduced because the harmonic amplitudes drop by a larger factor than does the fundamental amplitude. For example, if P_1 is reduced by 3 dB, P_3 can fall by 9 dB. It is therefore important to specify the output amplitude along with the SFDR. Of course, a low output amplitude degrades the SNDR.

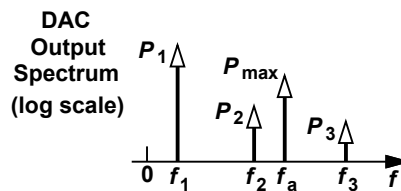


Figure 8.20 Spurs at a DAC output.

8.3 Digital Codes in DACs

The digital input to a DAC can assume different formats in different applications. Shown in Table 8.1 are three types often used in DACs (and in ADCs): binary, “thermometer,” and “1-of- n ” codes. The latter two are shown in *column* format to make their visualization easier.

Decimal Equivalent	0	1	2	3
Binary	00	01	10	11
Thermometer	0 0 0 0	0 0 0 1	0 0 1 1	0 1 1 1
1-of- n	0 0 0 1	0 0 1 0	0 1 0 0	1 0 0 0

Table 8.1 Digital codes used in DACs.

In the binary format, an m -bit number $D_m D_{m-1} \cdots D_1$ represents a decimal value of $D_m 2^{m-1} + D_{m-1} 2^{m-2} + \cdots + D_1 2^0$. In the thermometer code, a number j is represented by a column of j consecutive ONEs on the bottom and k consecutive ZEROS on the top such that $j + k$ is constant. For example, as shown in Table 8.1, the decimal number 3 translates to three ONES and one ZERO. This code can be viewed as a thermometer in which the height of mercury changes in proportion to the temperature.

In the 1-of- n code, each value is represented as a single ONE in a column of ZEROS, with the *position* of that ONE signifying the actual value. In Table 8.1, for example, the decimal number 3 is depicted as a ONE in the fourth position from the bottom.

Example 8.7

Design the logic necessary to convert a 3-bit binary input to a 1-of- n code.

Solution

As illustrated in Fig. 8.21, we employ an AND gate for each 1-of- n code entry, detecting when $D_3 D_2 D_1 = 000, 001, 010$, etc.

Example 8.8

Design the logic for converting a 3-bit binary input to a thermometer code.

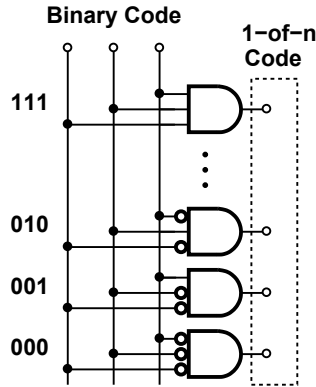


Figure 8.21 Binary to 1-of- n conversion.

Solution

We draw the truth table as shown in Fig. 8.22, where $T_8 T_7 \cdots T_1$ denotes the thermometer code and T_1 is the

$D_3 D_2 D_1$	T_8	T_7	T_6	T_5	T_4	T_3	T_2	T_1
000	0	0	0	0	0	0	0	0
001	0	0	0	0	0	0	0	1
010	0	0	0	0	0	0	1	1
011	0	0	0	0	0	1	1	1
100	0	0	0	0	1	1	1	1
101	0	0	0	1	1	1	1	1
110	0	0	1	1	1	1	1	1
111	0	1	1	1	1	1	1	1

Figure 8.22 Truth table for binary-to-thermometer conversion.

“bottom” entry. That is, the code fills up from the right. Note that $T_8 = 0$. From this inspection, we have

$$T_1 = D_3 + D_2 + D_1, \quad (8.19)$$

$$T_2 = D_3 + D_2, \quad (8.20)$$

$$T_3 = D_3 + D_2 D_1, \quad (8.21)$$

$$T_4 = D_3, \quad (8.22)$$

$$T_5 = D_3(D_2 + D_1), \quad (8.23)$$

$$T_6 = D_3 D_2, \quad (8.24)$$

$$T_7 = D_3 D_2 D_1. \quad (8.25)$$

The logic can be readily constructed.

8.4 Clocking in DACs

Recall from Section 8.2.2 that timing misalignment among the bits applied to a DAC’s cell introduces a glitch at the output. To avoid this issue, we wish to align the bits, a task performed by a register (a set of flipflops)

that precedes the DAC (Fig. 8.23). Even though the input bits may not arrive simultaneously, the “retimed” signal, D_R , changes only when CK goes high (or low), delivering properly aligned bits to the DAC.

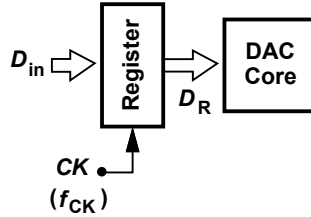


Figure 8.23 Use of a register to align the input data of a DAC.

Since the register *samples* the data at a rate of f_{CK} , it can *alias* the input spectrum unless it satisfies the Nyquist criterion. That is, f_{CK} must be at least twice the input signal bandwidth. As evident from Fig. 8.19, the output waveform resembles that of a sample-and-hold circuit, exhibiting a sinc droop (Chapter 3).

Sampling in DACs naturally leads to a more complex output spectrum, especially in the presence of distortion. Consider the scenario depicted in Fig. 8.24(a), where a sinusoid near the Nyquist-rate is applied to a

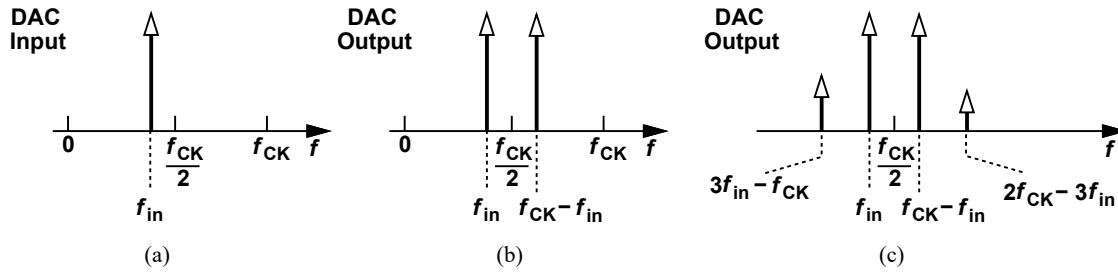


Figure 8.24 (a) A DAC receiving a sinusoid at f_{in} , (b) output spectrum, and (c) effect of nonlinearity.

DAC. Due to sampling, a component at $f_{CK} - f_{in}$ appears [Fig. 8.24(b)] (Chapter 3). Now, suppose these two components are denoted by $x(t) = A \cos \omega_1 t + A \cos \omega_2 t$, where $\omega_1 = 2\pi f_{in}$ and $\omega_2 = 2\pi(f_{CK} - f_{in})$. If the DAC introduces third-order distortion of the form $y = \alpha_1 x + \alpha_3 x^3$, then the output exhibits components given by $\alpha_3 A^3 \cos^3 \omega_1 t + 3\alpha_3 A^3 \cos^2 \omega_1 t \cos \omega_2 t + 3\alpha_3 A^3 \cos \omega_1 t \cos^2 \omega_2 t + \alpha_3 A^3 \cos^3 \omega_2 t$. The second and third terms translate to frequency components equal to $2\omega_1 - \omega_2 = 2\pi(3f_{in} - f_{CK})$ and $2\omega_2 - \omega_1 = 2\pi(2f_{CK} - 3f_{in})$, respectively (why?). Shown in Fig. 8.24(c), the resulting spectrum contains an in-band *spur* at $3f_{in} - f_{CK}$, i.e., a frequency that is not harmonically related to f_{in} or f_{CK} . The reader is encouraged to repeat these calculations if $y = \alpha_1 x + \alpha_2 x^2$.

Example 8.9

While testing a DAC in a laboratory, we observe a number of spurs at the output. How can we write the spur frequencies as $\pm m f_{in} \pm n f_{CK}$?

Solution

We increment f_{in} by Δf_{in} and observe the change, Δf_{out} , in a given spur frequency. The ratio, $\Delta f_{out} / \Delta f_{in} = m$ signifies that the spur arises from the m th harmonic of the input. A similar test can be performed for the clock.

Problems

- 8.1. Suppose the $2I_{REF}$ current source in Fig. 8.3 incurs an error equal to ΔV while the other two are accurate. Construct the input-output characteristic.
- 8.2. Does ΔI_D in the previous problem introduce offset and/or gain error?
- 8.3. Write an expression for the straight line passing through the end points of the characteristic obtained in Problem 8.1 and compute the DNL and INL profiles.
- 8.4. In the 3-bit DAC of Fig. 8.6, the width of the second LSB current source is mistakenly chosen equal to W rather than $2W$. Draw the input-output characteristic and determine the offset, gain error, DNL, and INL.
- 8.5. Repeat the previous problem if the MSB current source has a width of $3W$ rather than $4W$ while the other two are correct.
- 8.6. The switch controlled by D_1 in Fig. 8.6 is mistakenly replaced with a short circuit. Construct the input-output characteristic and determine the offset, gain error, DNL, and INL.
- 8.7. Repeat the previous problem if only the switch controlled by D_2 is mistakenly replaced with a short circuit.
- 8.8. Repeat the previous problem if the switch controlled by D_2 is mistakenly replaced with an open circuit.
- 8.9. Consider the circuit shown in Fig. 8.25. Suppose the initial condition on C_u is zero and the switch turns on at $t = 0$. Explain what happens to V_P and express V_{out} as a function of time.

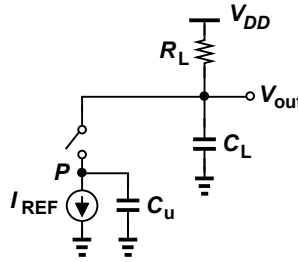


Figure 8.25 Switching network.

- 8.10. As in the previous problem, suppose a current source of I_{REF} in Fig. 8.17 exhibits a capacitance equal to C_u to ground. Thus, $2I_{REF}$ incurs $2C_u$, etc. Assuming $C_L > 0$, compute the worst-case settling time of the DAC.
- 8.11. Suppose we place the circuit of Fig. 8.17 in a black box and consider V_{out} as the DAC output. Plot the input-output characteristic of the circuit (while ignoring C_L). If R_L departs from its nominal value by ΔR , does the DAC exhibit offset, gain error, or nonlinearity?
- 8.12. Repeat the previous problem if V_{DD} departs from its nominal value by ΔV .
- 8.13. In the circuit of Fig. 8.17, we have $D_3 D_2 D_1 = 011$. We wish to change the input to 100 at $t = 0$, but D_3 jumps to 1 with a delay of Δt while D_2 and D_1 jump to 0 immediately. Sketch the output current and voltage waveforms. Do you observe a glitch at the output?
- 8.14. The input-output characteristic of a DAC is expressed as $y(t) = \alpha_1 x(t) + \alpha_3 x^3(t)$. If $x(t) = A \cos \omega_{in} t$, determine the amplitude of the third harmonic and plot the SFDR as a function of A . Assume $\alpha_1 > 0$ and $\alpha_3 < 0$, and ignore clocking effects.
- 8.15. Repeat the previous problem if $y(t) = \alpha_1 x(t) + \alpha_2 x^2(t)$, where $\alpha_1 > 0$ and $\alpha_2 < 0$.
- 8.16. A DAC receives a (digital) sinusoid of the form $A \cos(2\pi f_{in} t)$ and samples it at a rate of f_{CK} . The result then experiences nonlinearity characterized by $y(t) = \alpha_1 x(t) + \alpha_2 x^2(t)$. Assuming that $f_{in} = f_{CK}/2 - \Delta f$, sketch the output spectrum of the DAC's register (which acts as a sampler) and at the final output of the DAC.
- 8.17. Repeat the previous problem if $f_{in} = f_{CK}/2 + \Delta f$.
- 8.18. Two DACs exhibit nonlinear characteristics given by $y_1(t) = \alpha_1 x(t) + \alpha_2 x^2(t)$ and $y_2(t) = \alpha_1 x(t) + \alpha_3 x^3(t)$. If $x(t) = A \cos \omega_{in} t$, under what condition do they provide the same SFDR? Ignore clocking effects.
- 8.19. Suppose R_b in Fig. 8.17 is slightly nonlinear and its current and voltage follow the expression $I = \alpha_1 V + \alpha_2 V^2$. (Note that α_1 is the inverse resistance if $\alpha_2 = 0$.)
(a) Express V in terms of I .

(b) Express V_{out} in terms of $D_{in} = D_3 D_2 D_1$.

- 8.20.** Repeat the previous problem if α_2 is small. Note that the approximation $\sqrt{1+\epsilon} \approx 1 + \epsilon/2$ for $\epsilon \ll 1$ may not suffice here, requiring a more accurate one in the form of $1 + \epsilon/2 + \epsilon^2/k$.

Resistor-Ladder DACs

In this chapter, we analyze resistor-ladder DACs (RDACs). We begin with basic structures and study their speed and precision limitations. We then deal with differential realizations. Next, we examine the effect of gradients and random mismatches as well as the problem of “reference bounce” and methods of alleviating it.

9.1 Basic Structures

Recall that the output of a DAC is expressed as $A_{out} = \alpha D_{in}$, where α defines the dimension and full scale of A_{out} . For an output voltage, we can select a suitable full-scale value, V_{REF} , and subdivide it to obtain α , a task readily performed by “resistor ladders.” Shown in Fig. 9.1(a), such a structure consists of m nominally-equal resistors, generating $V_1, V_2, \dots, V_{m-1}, V_m$, where $V_j - V_{j-1} = V_{REF}/m$ is the output LSB. For example, an 8-bit DAC employs 256 resistors.

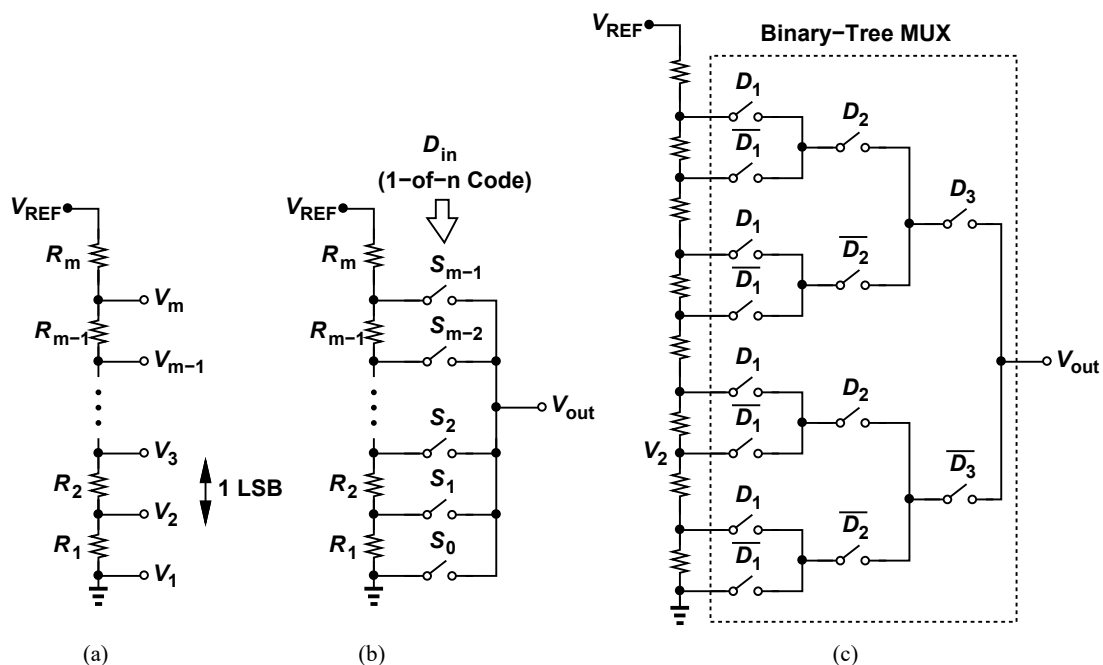


Figure 9.1 (a) Resistor ladder providing equally-spaced voltages, (b) RDAC using a 1-of- n code, and (c) 3-bit RDAC using a binary code.

For a given digital input, one of the voltages generated by the ladder must be selected and routed to the DAC output. This can be accomplished as shown in Fig. 9.1(b): m switches are attached to the ladder, serving as an “analog multiplexer” (MUX). For $D_{in} = j$, switch S_j turns on, and $V_{out} = j(V_{REF}/m)$. We recognize that the switches must be driven by a 1-of- n code (Chapter 8). That is, if the main input is in binary form, it must be applied to the logic shown in Fig. 8.21 before reaching the switches.

It is possible to avoid the 1-of- n logic if the analog multiplexer is so arranged as to directly operate with a binary code. Depicted in Fig. 9.1(c) is a 3-bit example driven by $D_{in} = D_3D_2D_1$. The “binary-tree multiplexer” selects adjacent ladder voltages by D_1 and $\overline{D_1}$, and successively halves the number of outputs in each stage. For example, $D_3D_2D_1 = 010$ selects V_2 .

Two properties of RDACs make them attractive for some applications. First, they provide a wide output voltage range, e.g., from zero to V_{DD} if $V_{REF} = V_{DD}$ in Fig. 9.1. As explained in Chapter 10, that is not the case for current-mode DACs. Second, RDACs exhibit an unconditionally monotonic characteristic because V_{out} in Figs. 9.1(b) and (c) can only increase as D_{in} increases regardless of resistor mismatches. This point stands in contrast to the behavior depicted in Fig. 8.15(b) for current-mode DACs.

One drawback of RDACs is that they cannot drive resistive loads, as illustrated by the following example.

Example 9.1

Shown in Fig. 9.2(a) is an RDAC driving a load resistor. Estimate the INL profile for this arrangement while neglecting the on-resistance of the switches. The ladder consists of m equal segments.

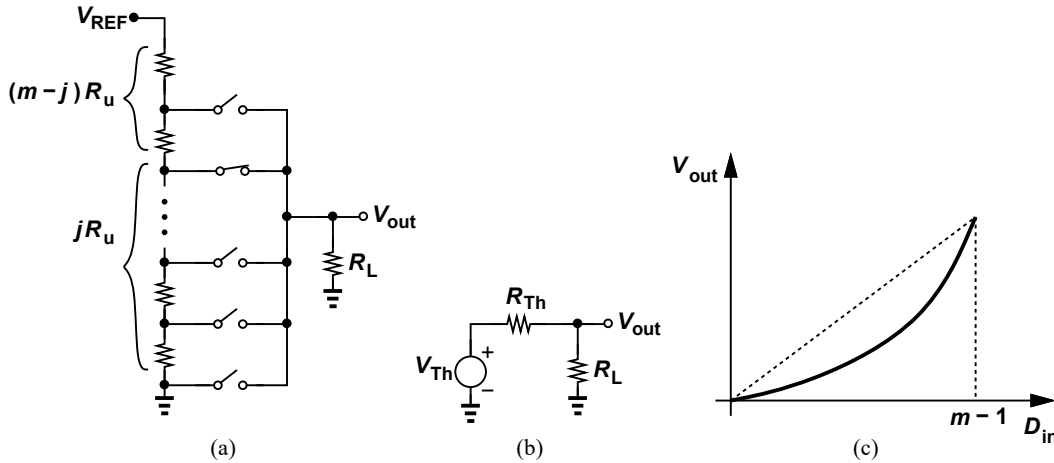


Figure 9.2 (a) Ladder driving a resistive load, (b) its equivalent circuit, and (c) resulting nonlinear characteristic.

Solution

We observe that the Thevenin resistance of the ladder changes with the digital input, creating nonlinear voltage division [Fig. 9.2(b)]. For $D_{in} = j$, $V_{Th} = jV_{REF}/m$ and R_{Th} is given by

$$R_{Th} = (jR_u) \parallel [(m-j)R_u], \quad (9.1)$$

where R_u denotes the unit resistor value. This expression simplifies to

$$R_{Th} = \frac{j(m-j)}{m} R_u, \quad (9.2)$$

yielding

$$V_{out} = \frac{R_L}{R_L + \frac{j(m-j)}{m} R_u} \cdot \frac{jV_{REF}}{m}. \quad (9.3)$$

Note that V_{out} is equal to zero for $j = 0$ and V_{REF} for $j = m$. For simplicity, we assume these two define the end points of the characteristic even though the last point is in fact at $j = m - 1$.

The INL profile is given by the difference between Eq. (9.3) and a straight line passing through the end points [Fig. 9.2(c)]. This line is expressed as $V_{out} = (j/m)V_{REF}$. It follows that

$$INL = \frac{-j^2(m-j)R_u}{R_L + \frac{j(m-j)}{m}R_u} \frac{V_{REF}}{m^2}. \quad (9.4)$$

For a small INL, we surmise that the load resistance must be much greater than ladder's Thevenin resistance, concluding that the denominator of the first fraction is approximately equal to R_L . That is,

$$INL \approx \frac{-j^2(m-j)}{m^2} \frac{R_u}{R_L} V_{REF}. \quad (9.5)$$

The absolute value of INL reaches a maximum of $(4/27)(mR_u/R_L)V_{REF}$ at $j = 2m/3$. For example, an 8-bit DAC targeting $INL < 0.5 \text{ LSB} = 0.5(V_{REF}/256)$ must satisfy the condition $mR_u/R_L < 27/(8 \times 256) = 0.013$; i.e., the total ladder resistance, mR_u , must be no more than $0.013R_L$.

The INL calculation procedure illustrated in the above example proves generally useful. We first express the actual characteristic as well as the straight line passing through its end points. We then subtract the two and find the peak INL, and normalize it to the full scale or to 1 LSB.

9.2 Speed Limitations

RDACs suffer from certain speed limitations, some intrinsic and some that trade with the ladder's power consumption. Consider the RDAC of Fig. 9.1(b), assuming switch S_j is turned on at $t = 0$ and $V_{out} = 0$. With no explicit load capacitance, the network can be reduced to that in Fig. 9.3, where R_{sw} denotes the on-resistance of S_j and C_{sw} the parasitic capacitance of each switch at the output node. The latter consists of the

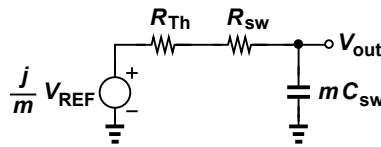


Figure 9.3 RDAC equivalent circuit for settling time calculation.

gate-drain overlap capacitance and the drain-bulk junction capacitance. It also includes the gate capacitance of S_j , which we neglect here. For the time constant, we have $\tau = (R_{Th} + R_{sw})(mC_{sw})$.¹ To minimize τ , we can reduce R_{Th} , at the cost of power consumption, but even for $R_{Th} \ll R_{sw}$, the ladder exhibits an intrinsic time constant equal to $mR_{sw}C_{sw}$. One switch drives the parasitic capacitances of m switches, i.e., the “fanout” is equal to m .

The time constant $R_{sw}C_{sw}$ is a fundamental property of the technology and does not depend on the width, W , of the switches because $R_{sw} \propto 1/W$ and $C_{sw} \propto W$. Thus, the fastest settling occurs if $R_{Th} \ll R_{sw}$.

Example 9.2

A 7-bit RDAC must be redesigned for a resolution of 8 bits. Neglecting the ladder resistance, explain what happens to the DAC settling time, t_s .

¹We assume C_{sw} is constant even though the drain-bulk capacitance varies with V_{out} .

Solution

The intrinsic time constant is doubled, but the settling precision must also be commensurate with the new resolution. The worst-case settling in Fig. 9.1(b) occurs when D_{in} goes from $00 \cdots 0$ to $11 \cdots 1$, forcing V_{out} to travel from 0 to $V_{REF} - 1 \text{ LSB} = V_{REF} - V_{REF}/m$ (why?). We then have

$$\left(V_{REF} - \frac{V_{REF}}{m}\right) \left(1 - \exp \frac{-t_s}{\tau}\right) = V_{REF} - \frac{V_{REF}}{m} - \frac{V_{REF}}{2m}, \quad (9.6)$$

where the right-hand side represents an error of 0.5 LSB with respect to the final value (Section 8.2.2). It follows that

$$t_s = \tau \ln(2m - 2) \quad (9.7)$$

$$\approx \tau \ln(2m) \quad (9.8)$$

$$\approx mR_{sw}C_{sw} \ln(2m). \quad (9.9)$$

To redesign a 7-bit DAC for a resolution of 8 bits, we have $\tau \rightarrow 2\tau$ and $m \rightarrow 2m$, obtaining a new settling time given by $2\tau \ln(4m)$.

Equation (9.9) represents the fastest settling of RDACs, but, in practice, the ladder resistance is not negligible. Let us consider the other extreme and choose a switch width that yields $R_{sw} \ll R_{Th}$, where R_{Th} is the highest Thevenin resistance of the ladder and equal to $mR_u/4$ (why?). Under this condition, the worst-case settling time is derived from Fig. 9.3 and Eq. (9.9):

$$t_s \approx \frac{mR_u}{4} mC_{sw} \ln(2m) \quad (9.10)$$

$$\approx \frac{m^2}{4} R_u C_{sw} \ln(2m). \quad (9.11)$$

Note the strong dependence on m . For example, an 8-bit DAC exhibits $t_s \approx 1.02 \times 10^5 R_u C_{sw}$.

It is important not to confuse the two cases studied above: if $R_{sw} \gg mR_u/4$, then $t_s \approx mR_{sw}C_{sw} \ln(2m)$, and if $R_{sw} \ll mR_u/4$, then $t_s \approx (m^2/4)R_u C_{sw} \ln(2m)$. We observe that neither case is optimum. Specifically, the latter case suffers from a long settling time because the wide switches lead to a large value for C_{sw} .

The design of RDACs proceeds as follows: (1) select a power budget, P , and the total ladder resistance, mR_u , so as to satisfy $P = V_{REF}^2/(mR_u)$, (2) choose, as an initial guess, a switch width that yields $R_{sw} \approx R_{Th} = mR_u/4$, and (3) including the load capacitance, C_L , use simulations to adjust the switch width for optimum settling. If t_s is excessively long, P must be raised and R_u lowered.

It is instructive to compare the speeds of the two RDAC topologies shown in Figs. 9.1(b) and (c). For an N -bit DAC, the former exhibits a time constant equal to

$$\tau_1 = 2^N (R_{Th} + R_{sw}) C_{sw}, \quad (9.12)$$

where we have replaced m with 2^N . The latter, on the other hand, contains N switches in series and $N - 1$ intermediate nodes loaded by $3C_{sw}$ (Fig. 9.4). The output node is loaded by $2C_{sw} + C_L$. Lumping the switch resistances and capacitances and neglecting C_L , we have

$$\tau_2 \approx (R_{Th} + NR_{sw})[3(N - 1)C_{sw} + 2C_{sw}] \quad (9.13)$$

$$\approx (R_{Th} + NR_{sw})[(3N - 1)C_{sw}]. \quad (9.14)$$

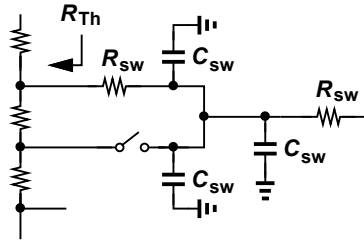


Figure 9.4 Capacitances associated with switches in an RDAC.

If $R_{Th} \ll NR_{sw}$, then $\tau_2 \approx (3N - 1)NR_{sw}C_{sw}$, a weaker function of the resolution than the value $\tau_1 \approx 2^N R_{sw}C_{sw}$ found for the former RDAC. For example, for a resolution of 8 bits, $\tau_2 \approx 184R_{sw}C_{sw}$ whereas $\tau_1 \approx 256R_{sw}C_{sw}$. A more accurate result can be obtained with the aid of the Elmore delay equation.

The choice between the RDACs of Figs. 9.1(b) and (c) depends on the environment in which the DAC is used, particularly, on the type of digital code that arrives at the input. In ADCs employing DACs, this code is more readily available in 1-of- n form than in binary form, making the topology in Fig. 9.1(b) more attractive.

9.3 Differential RDACs

In most systems, we prefer differential signaling for its robust operation in the presence of common-mode noise and for its lower distortion. Let us construct a differential 3-bit version of the RDAC in Fig. 9.1(b). Shown in Fig. 9.5, the structure provides two outputs that vary differentially between 0 and

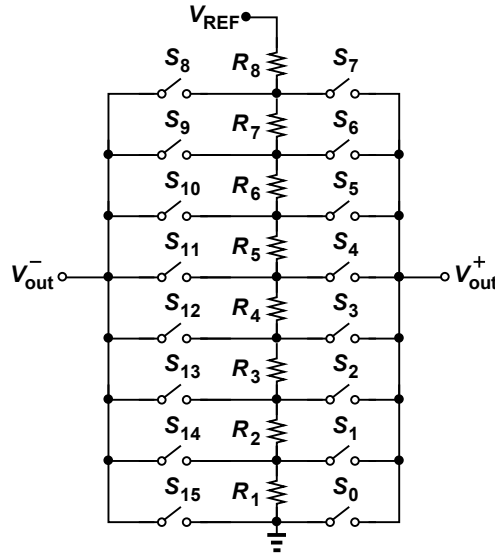


Figure 9.5 Differential RDAC.

$V_{REF} - 1$ LSB. For example, S_0 and S_8 turn on together, yielding $V_{out}^+ = 0$ and $V_{out}^- = V_{REF} - 1$ LSB, or S_1 and S_9 turn on together, delivering $V_{out}^+ = 1$ LSB, $V_{out}^- = V_{REF} - 2$ LSB, etc. Similarly, we have $V_{out}^+ = V_{REF} - 1$ LSB and $V_{out}^- = 0$ if S_7 and S_{15} are on. The differential output is zero if S_4 and S_{11} turn on.

The RDAC of Fig. 9.5 faces two issues. First, for a 1-LSB change in the digital input, $V_{out}^+ - V_{out}^-$ changes by $2V_{REF}/8$ rather than by $V_{REF}/8$. That is, the analog LSB size is twice that of the single-ended counterpart. This can be readily corrected by halving the value of V_{REF} .

Example 9.3

A single-ended RDAC uses $V_{REF} = 0.6$ V. Explain what type of MOS switches this DAC and its differential counterpart should incorporate. Assume a supply voltage of 1 V.

Solution

For switches tied to voltages near V_{REF} , the single-ended implementation requires PMOS or complementary transistors [Fig. 9.6(a)]. We note that M_1 has a gate-source voltage of about 0.4 V and hence a low overdrive; M_2 has $|V_{GS}| = 0.6$ V, exhibiting a lower resistance. The layout thus becomes more complex as the n -well of M_2 must be placed next to M_1 with proper spacing.

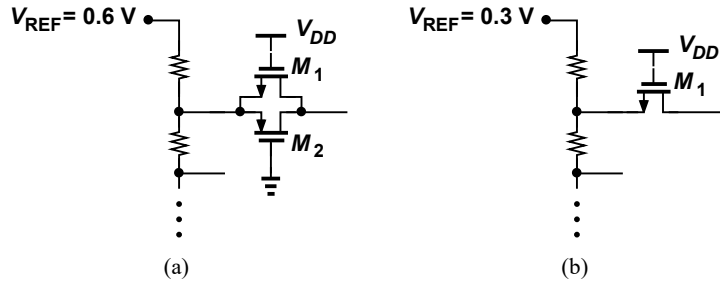


Figure 9.6 (a) Single-ended RDAC with complementary switches, and (b) differential RDAC using NMOS switches.

For the differential RDAC, on the other hand, $V_{REF} = 0.3$ V, allowing the use of only NMOS switches [Fig. 9.6(b)] as the worst-case V_{GS} is still around 0.7 V. This is another advantage of differential operation.

The second issue related to Fig. 9.5 is that the digital control lines driving the switches travel a long distance for higher resolutions. This is because S_0 and S_8 are connected to the same control signal (1-of- n code), and so are S_1 and S_9 , etc. This difficulty can be remedied through the use of a “folded” ladder. As illustrated in Fig. 9.7, a ladder resembling an inverted U provides the voltage taps to V_{out}^+ and V_{out}^- . Now, S_0 and S_8 are in close proximity, and so are S_1 and S_9 , etc.

9.4 Effect of Gradients

In the study of RDACs, we have thus far assumed that the unit resistors are exactly equal. In reality, they suffer from mismatches due to “gradients” and random effects.

During chip fabrication, it is possible that one parameter gradually varies from one side of the wafer to the other. For example, the thickness of the resistors may vary across the chip. For simplicity, we model such effects by a linear gradient in the parameter of interest; e.g., we assume the RDAC unit resistors have values equal to R_0 , $R_0 + \Delta R$, $R_0 + 2\Delta R$, etc. We expect that these mismatches translate to both DNL and INL.

Example 9.4

Consider the m -unit RDAC shown in Fig. 9.8(a), where the resistors suffer from a linear gradient. Assuming $\Delta R > 0$, sketch the input-output characteristic.

Solution

We observe that for $D_{in} = 00 \cdots 0$, $V_{out} = 0$, and for $D_{in} = 11 \cdots 1$, V_{out} is close to V_{REF} . That is, the characteristic is “pinned” at 0 and V_{REF} regardless of the value of ΔR . Examining the DAC for $D_{in} = 10 \cdots 0$, we recognize that $V_{out} < V_{REF}/2$ because the top half of the ladder has a *greater* resistance than the bottom half [Fig. 9.8(b)]. Thus, the actual characteristic is less than the ideal one for the entire range.

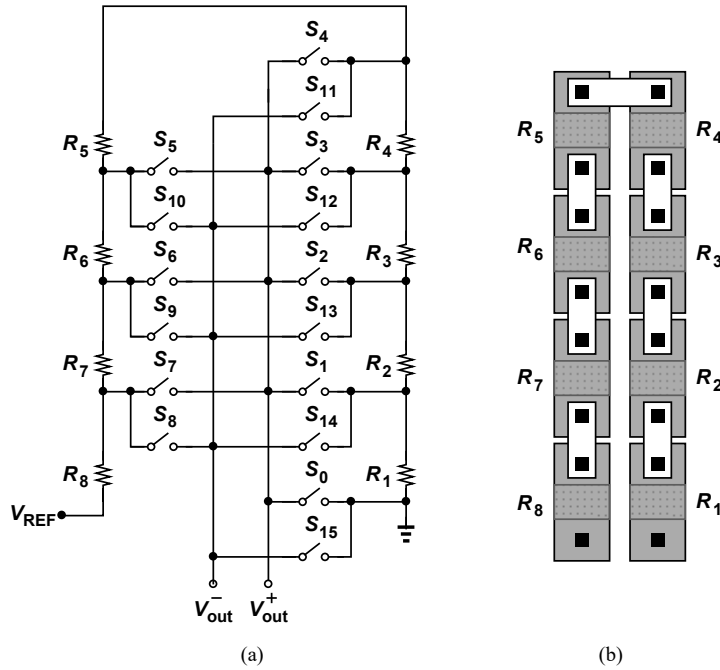


Figure 9.7 (a) Differential RDAC using a folded ladder, and (b) the ladder layout.

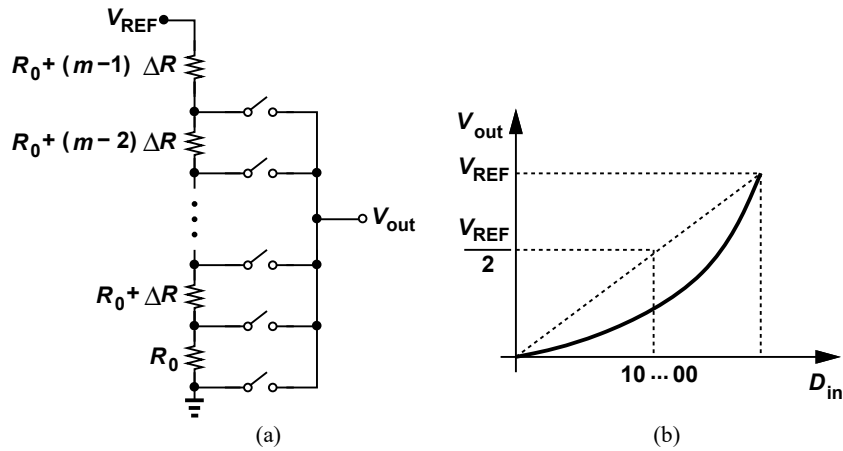


Figure 9.8 (a) RDAC experiencing a resistance gradient, and (b) resulting nonlinear characteristic.

Example 9.5

Explain why the DNL in Fig. 9.8(b) is negative for half of the input range and positive for the other half.

Solution

Writing $DNL_j = V_{j+1} - V_j - 1 \text{ LSB}$ and considering the INL as the integral of the DNL, we note that V_{out} can be less than the ideal value only if the step heights are less than 1 LSB and hence $DNL < 0$. This holds up to $D_{in} = 10 \dots 00$, beyond which the steps must be *taller* than 1 LSB ($DNL > 0$) so as to bring V_{out} closer to the straight line.

Let us compute the INL for the RDAC shown in Fig. 9.8(a). The voltage at the j -th tap of the ladder is equal to

$$V_j = \frac{\sum_{k=0}^{j-1} (R_0 + k\Delta R)}{\sum_{k=0}^{m-1} (R_0 + k\Delta R)} V_{REF} \quad (9.15)$$

$$= \frac{jR_0 + \frac{j(j-1)}{2}\Delta R}{mR_0 + \frac{m(m-1)}{2}\Delta R} V_{REF}. \quad (9.16)$$

The INL profile is given by the difference between this result and the straight line passing through the end points, $(j/m)V_{REF}$. It follows that

$$\text{INL}_j = \frac{j(j-m)}{R_0 + \frac{m-1}{2}\Delta R} \cdot \frac{\Delta R}{2m} V_{REF}. \quad (9.17)$$

Assuming $R_0 \gg (m-1)\Delta R/2$, we have

$$\text{INL}_j \approx \frac{j(j-m)}{2m} \cdot \frac{\Delta R}{R_0} V_{REF}, \quad (9.18)$$

whose magnitude reaches a maximum of $m[\Delta R/(8R_0)]V_{REF}$ at $j = m/2$. To ensure that $\text{INL} < 0.5 \text{ LSB} = V_{REF}/(2m)$,

$$\frac{\Delta R}{R_0} < \frac{4}{m^2}. \quad (9.19)$$

For example, if the INL in an 8-bit RDAC must remain less than $0.5 \text{ LSB} = V_{REF}/512$, we have $\Delta R/R_0 < 1/128^2$, a very small value.² Fortunately, in practice, the gradients are small enough to yield as much as 10 bits of linearity for RDACs.

The DNL of the DAC can be obtained from the definition:

$$\text{DNL}_j = V_{j+1} - V_j - 1 \text{ LSB} = V_{j+1} - V_j - V_{REF}/m. \quad (9.20)$$

Using Eq. (9.16), we have

$$\text{DNL}_j = \left(j - \frac{m-1}{2} \right) \frac{\Delta R}{R_0} \frac{V_{REF}}{m}. \quad (9.21)$$

Note that the DNL is negative up to $j = (m-1)/2$ and then becomes positive, as predicted in the above example. Our findings are conceptually summarized in Fig. 9.9. The reader is encouraged to explain intuitively why the DNL is greatest near the ends of the ladder. We show in Problem 9.8 that the DNL can also be derived by differentiating the INL expression in (9.18) with respect to j .

9.5 Effect of Random Mismatches

Before analyzing their effect on the INL and DNL, let us study resistor mismatches in general. Returning to Section 6.1, we surmise that, as with transistors, the mismatch between two resistors is inversely proportional

²Note that this condition justifies our earlier assumption that $R_0 \gg (m-1)\Delta R/2$.

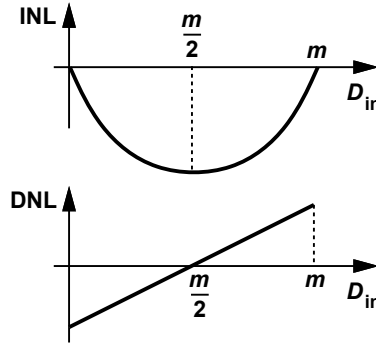


Figure 9.9 INL and DNL profiles due to resistance gradient.

to their areas:

$$\frac{\Delta R}{R_0} \propto \frac{1}{\sqrt{WL}}, \quad (9.22)$$

where W and L denote the resistors' nominal width and length, respectively. In practice, mismatches as low as 0.1% can be achieved by proper choice of resistor dimensions. The exact values of W and L necessary for a given mismatch are specific to the CMOS process in which the circuit is designed.

The resistors comprising an RDAC suffer from random mismatches. The analysis in [1] shows that, for a Gaussian resistor distribution, the tap voltages of an m -unit ladder also follow a nearly Gaussian behavior with a mean equal to jV_{REF}/m and a standard deviation equal to

$$\sigma_{V_j} = \sqrt{j \left(1 - \frac{j}{m}\right) \frac{\Delta R}{R_0} \frac{V_{REF}}{m}}, \quad (9.23)$$

where ΔR denotes the resistors' standard deviation and R_0 their mean. We therefore expect that, statistically, the midpoint of the ladder (at $j = m/2$) is likely to display the largest INL, equal to

$$\text{INL}_{max} = \frac{1}{\sqrt{4m}} \frac{\Delta R}{R_0} V_{REF}. \quad (9.24)$$

To ensure $\text{INL} < 0.5 \text{ LSB}$,

$$\frac{\Delta R}{R_0} < \frac{1}{\sqrt{m}}. \quad (9.25)$$

For example, an 8-bit RDAC requires $\Delta R/R < 6.3\%$. Interestingly, the unit resistor matching need not be as tight as 8 bits (i.e., $1/512$) for the ladder to achieve 8-bit linearity, a result of the mismatch averaging that occurs over the resistors. The following example illustrates this point.

Example 9.6

Recall from Section 6.1 that if k nominally-equal, independent Gaussian variables are added, the mean is multiplied by k and the standard deviation (SD) by $\sqrt{k}$. Using this principle, compute the SD of the voltage at the midpoint of a ladder. Assume a mean value of R_0 and an SD of ΔR for the resistors.

Solution

Drawing the ladder as shown in Fig. 9.10, we observe that the top $m/2$ resistors exhibit a mean value of $R_t = (m/2)R_0$ and a standard deviation of $\sigma_t = \sqrt{m/2}\Delta R$. Similarly, we have for the bottom $m/2$ units:

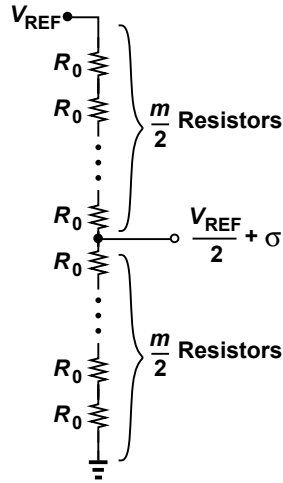


Figure 9.10 Random mismatches in an RDAC.

$R_b = (m/2)R_0$ and $\sigma_b = \sqrt{m/2}\Delta R$. In general, the top $m/2$ units add up to $(m/2)R_0 + \Delta R_t$, where ΔR_t is a random variable with a zero mean and an SD equal to σ_t . By the same token, the bottom $m/2$ units add up to $(m/2)R_0 + \Delta R_b$. Noting that $1/(2 + \epsilon) \approx 1 - \epsilon/2$ for $\epsilon \ll 1$, we write the midpoint voltage as

$$V_{mid} = \frac{(m/2)R_0 + \Delta R_b}{(m/2)R_0 + \Delta R_b + (m/2)R_0 + \Delta R_t} V_{REF} \quad (9.26)$$

$$= \frac{1 + \frac{\Delta R_b}{(m/2)R_0}}{2 + \frac{\Delta R_b + \Delta R_t}{(m/2)R_0}} V_{REF} \quad (9.27)$$

$$\approx \left[1 + \frac{\Delta R_b}{(m/2)R_0} \right] \left[1 - \frac{\Delta R_b + \Delta R_t}{2(m/2)R_0} \right] \frac{V_{REF}}{2} \quad (9.28)$$

$$\approx \frac{V_{REF}}{2} + \frac{\Delta R_b - \Delta R_t}{4(m/2)R_0} V_{REF}. \quad (9.29)$$

Since the mismatches between the top and bottom resistors are uncorrelated, the SD of $\Delta R_2 - \Delta R_1$ should be computed as $\sqrt{\sigma_t^2 + \sigma_b^2} = \sqrt{m}\Delta R$. It follows that the SD of the midpoint voltage is equal to

$$\sigma_{max} = \frac{\sqrt{m}\Delta R}{4(m/2)R_0} V_{REF} \quad (9.30)$$

$$= \frac{1}{\sqrt{4m}} \frac{\Delta R}{R_0} V_{REF}, \quad (9.31)$$

which agrees with (9.24). The other tap voltages can be analyzed in a similar manner.

Example 9.7

We double the number of resistors in an 8-bit RDAC without changing the resistor dimensions [Fig. 9.11(a)]. Determine the midpoint INL of the resulting 9-bit DAC in volts and in terms of the 9-bit LSB.

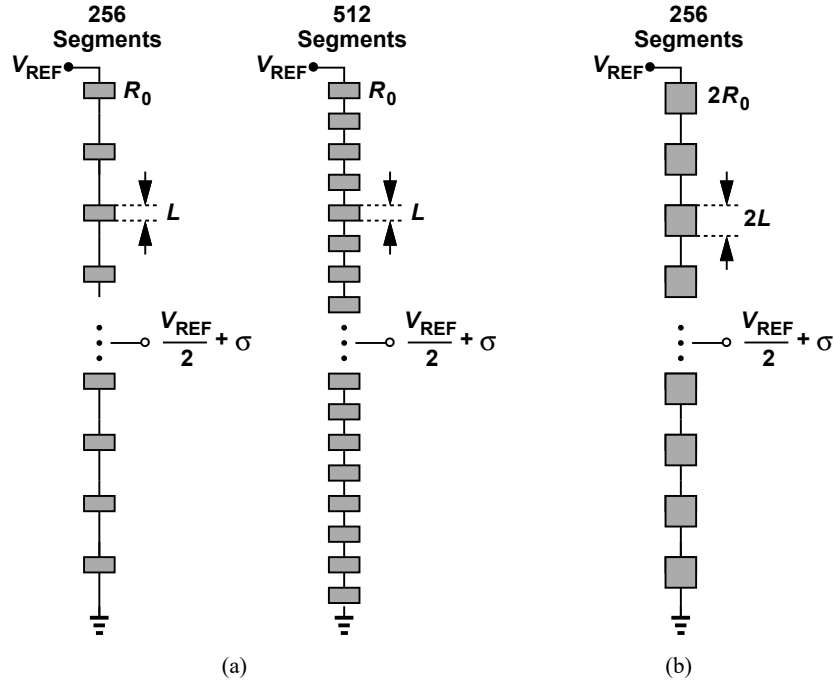


Figure 9.11 (a) Transformation of an 8-bit RDAC to a 9-bit RDAC, and (b) equivalent structure of 9-bit RDAC.

Solution

According to (9.24), INL_{max} falls by a factor of $\sqrt{2}$, i.e., the midpoint of the ladder statistically departs from $V_{REF}/2$ by a standard deviation equal to $(1/\sqrt{4 \times 512})(\Delta R/R_0)V_{REF}$ volts. Why does this happen? Considering only the midpoint, we observe that doubling the number of the resistors is equivalent to increasing their lengths by a factor of 2 [Fig. 9.11(b)]. For these longer resistors, (9.22) suggests a $\sqrt{2}$ reduction in mismatch. In other words, we can view the new RDAC from two different perspectives: (a) the number of resistors, m , in (9.24) is doubled, or (b) the ladder still has 256 units but $\Delta R/R$ is lowered by a factor of $\sqrt{2}$.

To express the INL in LSB, we must divide (9.24) by V_{REF}/m :

$$INL_{max} = \sqrt{\frac{m}{4}} \frac{\Delta R}{R_0} \text{ LSB.} \quad (9.32)$$

For example, if the 8-bit DAC is designed for $INL_{max} = 0.5 \text{ LSB}$, the 9-bit DAC exhibits an INL of $0.5 \times \sqrt{2} \text{ LSB}$. The reader is encouraged to ponder the difference between the two views here: the INL expressed in volts versus the INL expressed in LSB.

9.6 Problem of Reference Bounce

We have thus far assumed that the ladder in an RDAC is driven by an ideal voltage source, V_{REF} . In practice, this reference must be generated by an on-chip circuit, which inevitably suffers from a finite output impedance. The DAC output settling then takes longer because V_{REF} itself “bounces” up and down.³

Figure 9.12 depicts a simplified situation, modeling the reference generator by a voltage source and an output resistance. In this case, we add R_T to the ladder’s resistance to estimate the settling time constant.

³The term “bounce” is used even if the reference voltage is simply disturbed.

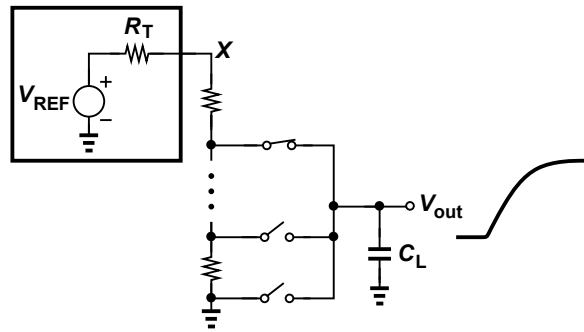


Figure 9.12 Reference bounce in an RDAC.

Example 9.8

A student decides to tie a large (“decoupling” or “bypass”) capacitor from X in Fig. 9.12 to ground so as to minimize the voltage bounce at this node. Is this a good idea?

Solution

It is if the capacitor is sufficiently large. Let us consider a simpler situation, depicted in Fig. 9.13(a). Here, we have neglected the ladder’s resistance⁴ and wish to study the settling time of V_{out} before and after C_1 is

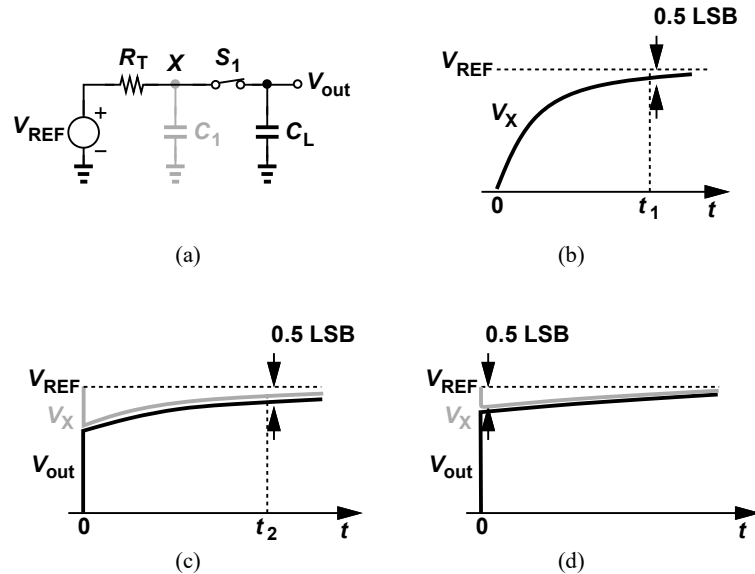


Figure 9.13 (a) Equivalent RDAC circuit, (b) settling behavior, (c) effect of adding C_1 , and (d) waveforms for a very large C_1 .

added. Suppose C_L has a zero initial voltage and V_{out} must settle to within 0.5 LSB of V_{REF} . For a resolution of N bits, we write $2^N = m$. We first plot V_X in the absence of C_1 [Fig. 9.13(b)]. The settling time for V_X and V_{out} is given by Eq. (9.8):

$$t_1 = R_T C_L \ln(2m). \quad (9.33)$$

⁴Which is a good approximation if the RDAC’s topmost switch is turned on.

With $C_1 > 0$, C_L receives some charge from C_1 when S_1 turns on. Consequently, V_X jumps down and V_{out} jumps up, both reaching a value of

$$V_0 = \frac{C_1}{C_1 + C_L} V_{REF} \quad (9.34)$$

at $t = 0^+$ (why?). We may think that $t_2 < t_1$ because V_{out} now begins closer to its final value. However, since the time constant has increased from $\tau_1 = R_T C_L$ to $\tau_2 = R_T (C_L + C_1)$, the situation is more complex. We have

$$V_{out} = V_{REF} + (V_0 - V_{REF}) \exp \frac{-t}{\tau_2}. \quad (9.35)$$

The time necessary for V_{out} to reach $V_{REF} - 0.5 \text{ LSB} = V_{REF} - V_{REF}/(2m)$ is given by

$$t_2 = R_T (C_L + C_1) \ln \left(2m \frac{C_L}{C_1 + C_L} \right). \quad (9.36)$$

How must we choose C_1 to obtain $t_2 < t_1$? We have from Eqs. (9.33) and (9.36):

$$(C_L + C_1) \ln \frac{2m C_L}{C_1 + C_L} < C_L \ln(2m) \quad (9.37)$$

and hence

$$\left(1 + \frac{C_L}{C_1} \right) \ln \frac{C_L}{C_1 + C_L} < \ln \frac{1}{2m}. \quad (9.38)$$

If $C_1 \gg C_L$,

$$1 + \frac{C_L}{C_1} > 2m. \quad (9.39)$$

If $2m \gg 1$, we have $C_1 \approx (2m)C_L$. For example, an 8-bit DAC requires $C_1 \approx 512C_L$, typically a very large value. This condition simply means that the initial charge sharing between C_1 and C_L is so small that V_{out} immediately jumps into the error band and remains there [Fig. 9.13(d)]. In summary, for C_1 to improve the settling, it must be about $2m$ times the load capacitance.

It is common practice to interpose a low-impedance buffer between the reference generator and the ladder, as exemplified by the structure in Fig. 9.14(a). However, the buffer's output impedance, Z_{out} , does rise with frequency, acting as an inductor and degrading the DAC settling speed. For example, if the op amp in Fig. 9.14(a) is modeled by a one-pole circuit, $A_0/(1 + s/\omega_0)$, and an open-loop output resistance equal to R_0 , we have

$$Z_{out} = \frac{R_0}{1 + \frac{A_0}{1 + \frac{s}{\omega_0}}} \quad (9.40)$$

$$= \frac{s + \omega_0}{s + (A_0 + 1)\omega_0} R_0. \quad (9.41)$$

Thus, Z_{out} exhibits a zero at ω_0 and a pole at $(A_0 + 1)\omega_0$. As illustrated in Fig. 9.14(b), $|Z_{out}|$ begins at a low value, $R_0/(A_0 + 1)$, at low frequencies, but climbs to R_0 beyond $(A_0 + 1)\omega_0$, which is close to the unity-gain bandwidth (UGB) of the op amp. For fast settling, the op amp must be designed for a wide bandwidth, thereby suffering from a high power consumption.

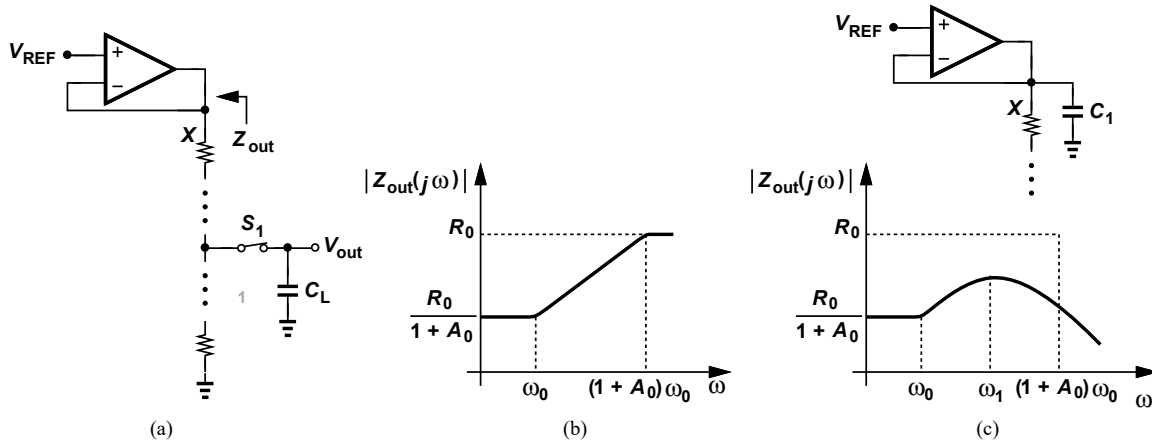


Figure 9.14 (a) Reference buffer driving an RDAC, (b) behavior of buffer's output impedance, and (c) behavior with C_1 added.

The rise of $|Z_{out}|$ in Fig. 9.14(b) with frequency proves problematic. As a remedy, we tie a capacitor from node X to ground so as to force $|Z_{out}|$ to fall at high frequencies [Fig. 9.14(c)]. Now, the peak occurs at a certain frequency, ω_1 . The choice of ω_1 is not straightforward and must be made by careful simulations of the output settling behavior (in the time domain).

References

- [1]. S. Kuboki et al, "Nonlinearity analysis of resistor string A/D converters," *IEEE Transactions on Circuits and Systems*, vol. 29, pp. 383-389, June 1982.

Problems

- 9.1. Suppose the load resistor, R_L , in Fig. 9.2 is tied to V_{REF} rather than to ground. Estimate the INL profile.
- 9.2. A student designing a 3-bit RDAC [similar to that in Fig. 9.1(a)] mistakenly uses seven resistor segments and seven switches in the ladder rather than eight. Plot the input-output characteristic and determine the gain error, DNL, and INL.
- 9.3. Determine the INL profile for the circuit of Fig. 9.2 if each switch exhibits an on-resistance of R_{sw} .
- 9.4. Repeat the previous problem for the DAC shown in Fig. 9.1(c), assuming that a large load resistance, R_L , is tied between the output node and ground. The ladder consists of m equal segments.
- 9.5. Consider a 3-bit RDAC similar to that in Fig. 9.1 and suppose the nodes generating V_3 and V_4 are accidentally shorted to each other. Plot the input-output characteristic and determine the gain error, DNL, and INL.
- 9.6. For the differential RDAC shown in Fig. 9.5, plot V_{out}^+ , V_{out}^- , and $V_{out}^+ - V_{out}^-$ versus the digital input.
- 9.7. Suppose the differential RDAC shown in Fig. 9.5 drives large load resistances and its two outputs in a manner similar to Fig. 9.2.
 - (a) Find V_{out}^+ and V_{out}^- when S_0 and S_8 are on.
 - (b) Find V_{out}^+ and V_{out}^- when S_1 and S_9 are on.
 - (c) Find V_{out}^+ and V_{out}^- when S_4 and S_{11} are on.
 - (d) Sketch the input-output characteristic.
- 9.8. Prove that the DNL expression in Eq. (9.21) can be obtained by differentiating the INL in Eq. (9.18) with respect to j .
- 9.9. Suppose the resistors in Fig. 9.5 experience a linear gradient similar to that in Fig. 9.8. (a) Write expressions for V_{out}^+ and V_{out}^- in terms of the digital input, j . Assume j varies from zero to 7. (b) Sketch $V_{out}^+ - V_{out}^-$ as a function of j and compute the straight line passing through the end points. (c) Determine the INL profile and differentiate it with respect to j to find the DNL profile as well.
- 9.10. Suppose V_{REF} in Fig. 9.10 is halved. What happens to the INL if it is expressed in volts or in LSB?
- 9.11. We double the width and the length of the resistor segments in Fig. 9.10. Determine by what factor the INL drops.
- 9.12. Suppose the unit resistors in the ladder of Fig. 9.12 have a value of R_u . If the ladder contains m segments, determine the worst-case time constant of the circuit.
- 9.13. Figure 9.15 depicts a common reference buffer topology. Here, M_1 provides the current drawn by the resistor ladder and amplifier A_1 ensures that $V_X \approx V_{REF}$. (a) Determine the low-frequency output resistance of the buffer. (b) Suppose the amplifier exhibits a one-pole response, $A_1 = A_0/(1 + s/\omega_p)$. Determine the output impedance of the buffer and sketch its magnitude as a function of frequency. (c) If we reduce the ladder's resistance, does the overall buffer become more stable or less? (Hint: Consider how the loop gain changes.)

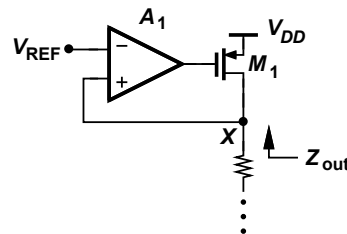


Figure 9.15 Reference buffer.

- 9.14. The RDAC of Fig. 9.1(c) is designed for a resolution of 8 bits. Using the model shown in Fig. 9.4, estimate the worst-case time constant.

- 9.15.** Suppose the differential RDAC of Fig. 9.5 must drive load capacitances equal to C_L at its outputs. Figure 9.16 shows the simplified circuit when S_1 and S_9 turn on. Using superposition, we ground the top of the ladder. Write KCLs at X and Y in the time domain to obtain two differential equations in terms of V_X and V_Y .

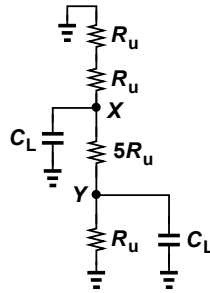


Figure 9.16 Simplified DAC.

Current-Steering DACs

As seen in Section 8.1, a digital input to a DAC can control a number of current sources so as to generate an analog output. In this chapter, we extend the idea to “current-steering DACs” (IDACs) and delve into their design. IDACs find wide usage as stand-alone functions in communication systems and have therefore been studied extensively. We begin with basic topologies and gradually elevate our understanding to more advanced architectures. We deal with matching issues, binary-weighted and “thermometric” topologies, R-2R DACs, and “segmented” structures. We then examine nonlinearity due to the output resistance of current sources. Next, we introduce the “matrix” architecture, and analyze dynamic errors. Some of the concepts presented here also apply to capacitor DACs (studied in the next chapter). This chapter is fairly long and invites the reader to take frequent breaks.

10.1 Basic Topologies

Let us begin with the IDAC implemented in Section 8.1, repeated in Fig. 10.1. We wish to examine the

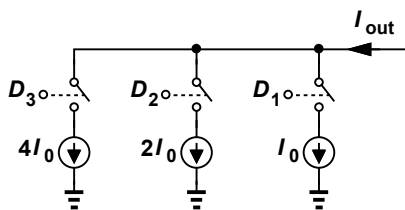


Figure 10.1 Simple 3-bit DAC.

operation of the circuit more closely and understand its limitations. We make three observations: (1) The DAC is capable of driving a resistive load. (2) The output voltage cannot swing to zero (or V_{DD}) because the current sources and the switches occupy some voltage headroom. Both of these properties stand in contrast to those of RDACs. (3) As explained in Section 8.2.1, mismatches among the current sources give rise to DNL and INL. This point requires proper sizing of the transistors and attention to the symmetry of the layout. We study the last two issues extensively in Section 10.7.

But the topology of Fig. 10.1 entails two other effects that degrade its performance. First, the finite output resistance of the current sources introduces additional DNL and INL. This can be readily seen from the small-signal model shown in Fig. 10.2(a), where the output resistance inversely scales with the nominal current.¹ Here, the currents flow *into* the output node for ease of illustration. For $D_3D_2D_1 = 001$, only the LSB current source is enabled, yielding $V_{out} = I_0(R_L || r_O)$. For $D_3D_2D_1 = 010$, on the other hand,

¹This is because doubling the drain current, for example, halves $r_O = \lambda/I_D$.

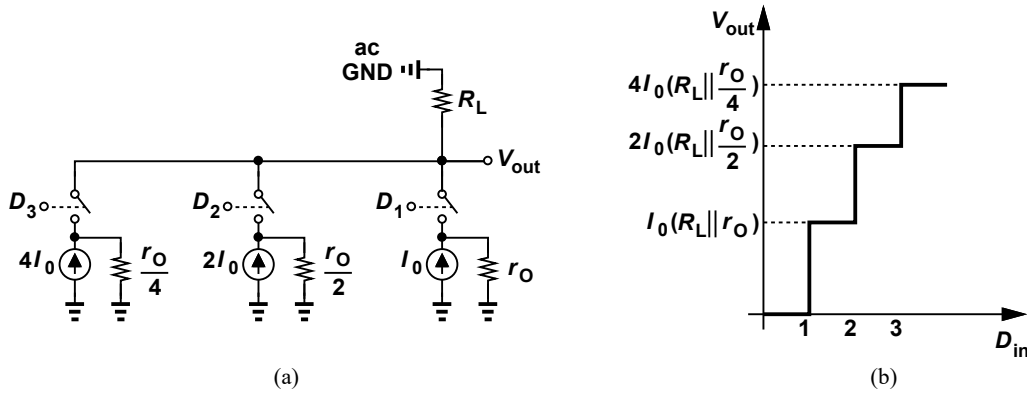


Figure 10.2 (a) DAC including output resistance of current sources, and (b) resulting characteristic.

$V_{out} = 2I_0[R_L || (r_O/2)]$, which is *not* twice the previous value. As depicted in Fig. 10.2(b), the output step height decreases as D_{in} increases because the total resistance at the output node falls. This phenomenon leads to DNL and INL and is formulated in Section 10.6.

Second, the switching actions in Fig. 10.1 produce significant glitches at the output. Consider the LSB branch, shown in Fig. 10.3, with the switch realized by M_2 . When D_1 is low, M_2 is off and M_1 in the deep triode region; C_X is thus discharged to zero. Now, when D_1 rises and M_2 turns on, C_X must charge through

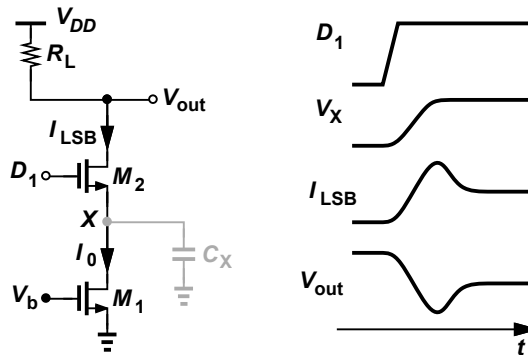


Figure 10.3 Effect of capacitance at X on a DAC output waveform.

M_2 and R_L . That is, M_2 initially carries I_0 and the current drawn by C_X , causing a glitch in V_{out} . This effect is more pronounced when $D_3D_2D_1$ goes from 000 to 111 (why?).

Example 10.1

Compute the output resistance of the current-switching branch in Fig. 10.3. Neglect body effect.

Solution

If both M_1 and M_2 operate in saturation, the cascode structure exhibits an output resistance equal to $(1 + g_{m2}r_{O2})r_{O1} + r_{O2}$, a relatively high value, where body effect is neglected. However, if V_{out} is so low as to drive M_2 into the deep-triode region, then this switch introduces only its on-resistance, R_{on} , in the path, yielding a total output resistance of $r_{O1} + R_{on}$, a substantially lower value. In other words, to suppress the effect illustrated in Fig. 10.2(b), we wish to operate M_2 in the saturation region. Whether that is possible or not will be studied below.

We deal with the glitch first. We wish to prohibit node X in Fig. 10.3 from falling to zero when M_2 turns off. This can be accomplished by “differential switching,” i.e., by “steering” the current of M_1 to another path as M_2 is disabled. Illustrated in Fig. 10.4(a), the idea is to turn on another device, M'_2 , so that V_X is held at a desirable value and C_X remains charged. Ideally, V_X must not change as D_1 falls and $\overline{D}_1$ rises, but in reality,

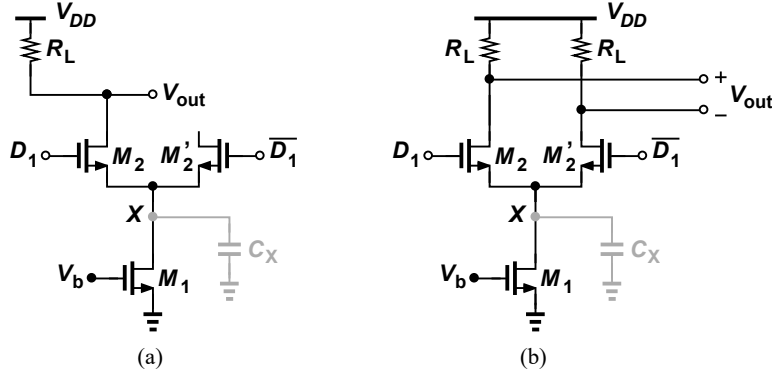


Figure 10.4 (a) Use of current steering to reduce tail voltage swing, and (b) fully-differential cell.

it does to some extent. We recognize that M_2 and M'_2 resemble a differential pair, introducing little change at their tail node if their gate voltage swings are small.²

To what node should we connect the drain of M'_2 in Fig. 10.4(a)? We can tie it to V_{DD} , but at the cost of “wasting” the current. A better approach is to add another R_L and connect this drain to it [Fig. 10.4(b)]. The DAC now naturally delivers a differential output.

Shown in Fig. 10.5 is a 3-bit example of the differential IDAC developed thus far. Each “cell” consists of a tail current source and a differential switching pair controlled by an input bit and its complement. We examine V_A and V_B as the input varies from 000 to 111. For $D_3D_2D_1 = 000$, the left switches in the cells are off and $V_A = V_{DD}$, while the right switches are on, drawing a total current of $7I_0$ and generating $V_B = V_{DD} - 7I_0R_L$. If $D_3D_2D_1 = 001$, then the LSB current is steered to the left, flowing through R_{L1} and yielding $V_A = V_{DD} - I_0R_L$ and $V_B = V_{DD} - 6I_0R_L$. That is, V_A and V_B vary differentially and cross each other at $D_3D_2D_1 = 100$.

We expect that differential operation improves the IDAC’s performance considerably. As explained in Chapter 2, differential circuits exhibit less nonlinearity. Moreover, the output is much less sensitive to noise on V_{DD} in Fig. 10.5 than in Fig. 10.3. In addition, and interestingly, the differential DAC *generates* less noise on V_{DD} , as studied in the following example.

Example 10.2

We wish to compare a 3-bit DAC using the single-ended cell of Fig. 10.3 with the differential structure of Fig. 10.5(a). Suppose the V_{DD} line in Figs. 10.3 and 10.5 suffers from a small series inductance, L_{DD} . Explain what happens in each DAC when D_{in} jumps from 000 to 111.

Solution

As shown in Fig. 10.6(a), the single-ended DAC draws a current of $7I_0$ from V_{DD} , causing a large transient voltage drop across L_{DD} equal to $L_{DD}dI_{DD}/dt$. We have

$$V_{out} = V_{DD} - L_{DD} \frac{dI_{DD}}{dt} - 7I_0R_L. \quad (10.1)$$

²As seen below, this is not exactly the case.

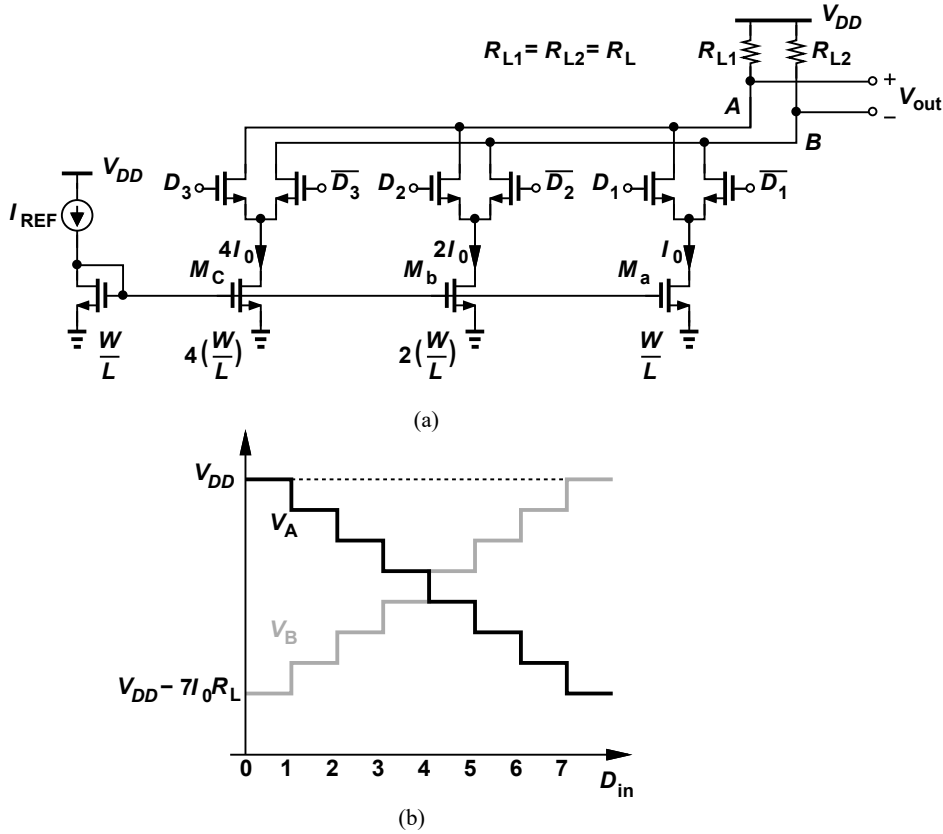


Figure 10.5 A 3-bit differential DAC and its characteristic.

Thus, V_{out} experiences a large undershoot. In practice, due to parasitic capacitances at the output node, the output may “ring” for some time before it settles.

Now, consider the differential topology in Fig. 10.6(b). We recognize that, when $D_{in} = 000$, the left transistors in the differential pairs are on, the right ones are off, and a current of $7I_0$ flows from V_{DD} . When D_{in} jumps to 111, the left transistors turn on and the supply current still settles to $7I_0$. Consequently, the large $L_{DD}dI_{DD}/dt$ drop is absent here and the DAC’s supply quieter.

In practice, I_{DD} in Fig. 10.6(b) does change to some extent because of the tail capacitance within the differential pairs. Illustrated in Fig. 10.6(c), this effect stems from the finite rise and fall times of the digital input, during which V_X drops. The reader can prove that this drop occurs. We note that a fraction of I_0 flows from C_X between t_1 and t_2 and hence does not reach the DAC output. Nevertheless, this phenomenon is much less significant than that in Fig. 10.6(a).

The differential current-steering DAC architecture of Fig. 10.5 forms the foundation for our study of IDACs. In the next sections, we delve into more details.

10.2 Basic Matching Considerations

As seen in Section 8.2.1, the DNL and INL of IDACs depend on the mismatches among their current sources. This means that the design must strive to minimize these effects. In this section, we study both deterministic and random mismatches.

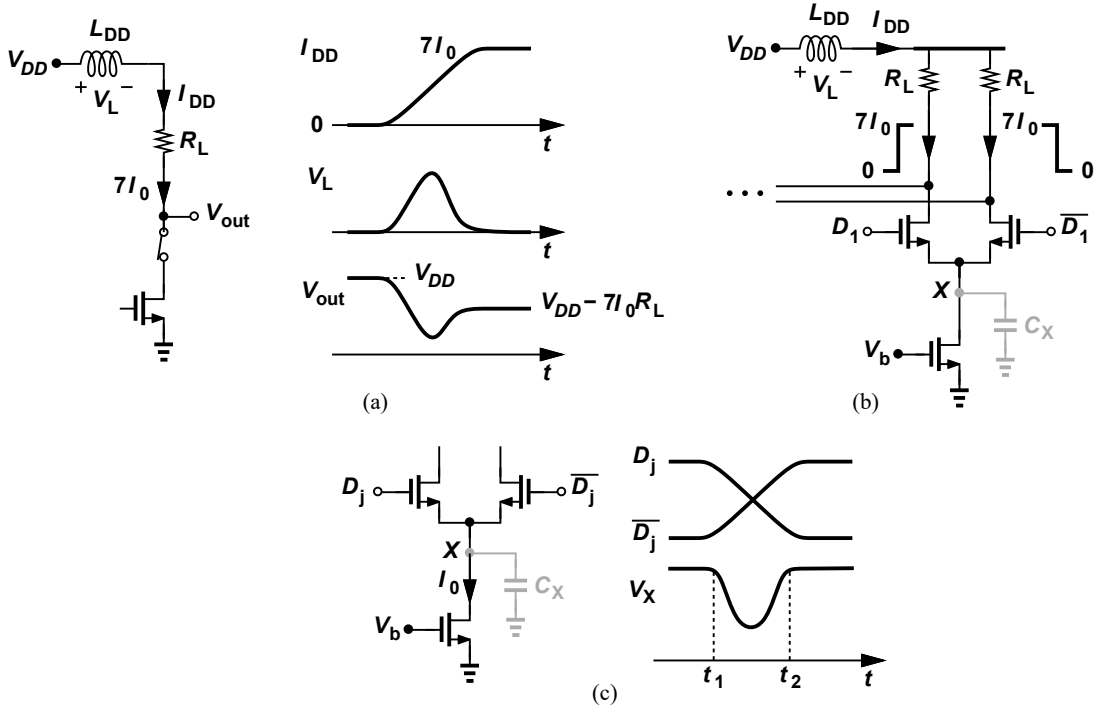


Figure 10.6 (a) Supply inductance in a single-ended DAC, (b) supply inductance in a differential DAC, and (c) tail voltage fluctuation.

10.2.1 Deterministic Mismatches

Consider the structure shown in Fig. 10.7(a). How do we ensure proper matching between these two branches? That is, how do we ensure that the current controlled by S_2 is, very accurately, twice that controlled by S_1 ? We may simply choose a transistor twice as wide for $2I_0$ compared to that for I_0 [Fig. 10.7(b)]. This, however, does not guarantee proper current scaling. As depicted in Fig. 10.7(c), the gate of a MOSFET must climb on the two edges, creating an “effective” width slightly less than the “drawn” width, W . Thus, the device with a drawn width of $2W$ does not carry exactly twice the current, suffering from a deterministic mismatch.

For this reason, we design $2I_0$ as two identical unit current sources in parallel [Fig. 10.7(d)]. Now, all units have the same width and length and incur only random mismatches. An 8-bit IDAC, therefore, requires 255 unit current sources, grouped so as to provide $I_0, 2I_0, 4I_0, \dots, 128I_0$.

Another type of deterministic mismatch arises from channel-length modulation. Suppose S_1 and S_2 in Fig. 10.7(a) are realized by *identical* devices. Illustrated in Fig. 10.8(a) for a differential architecture, this choice inevitably leads to unequal drain-source voltages for M_1 and M_2 because S_1 and S_2 carry unequal currents. Writing

$$I_{D1} \propto (V_b - V_{TH})^2 (1 + \lambda V_{DS1}), \quad (10.2)$$

$$I_{D2} \propto 2(V_b - V_{TH})^2 (1 + \lambda V_{DS2}), \quad (10.3)$$

we observe that $I_{D2} < 2I_{D1}$ because $V_{DS2} < V_{DS1}$. The foregoing issue is resolved by scaling the widths of the switches according to that of the current sources. In fact, for ease of layout, we first construct a “switching unit” consisting of a current source and a differential pair, and then duplicate this unit to obtain the higher bits [Fig. 10.8(b)]. In this case, all unit current sources have equal drain-source voltages regardless of the value of the digital input.

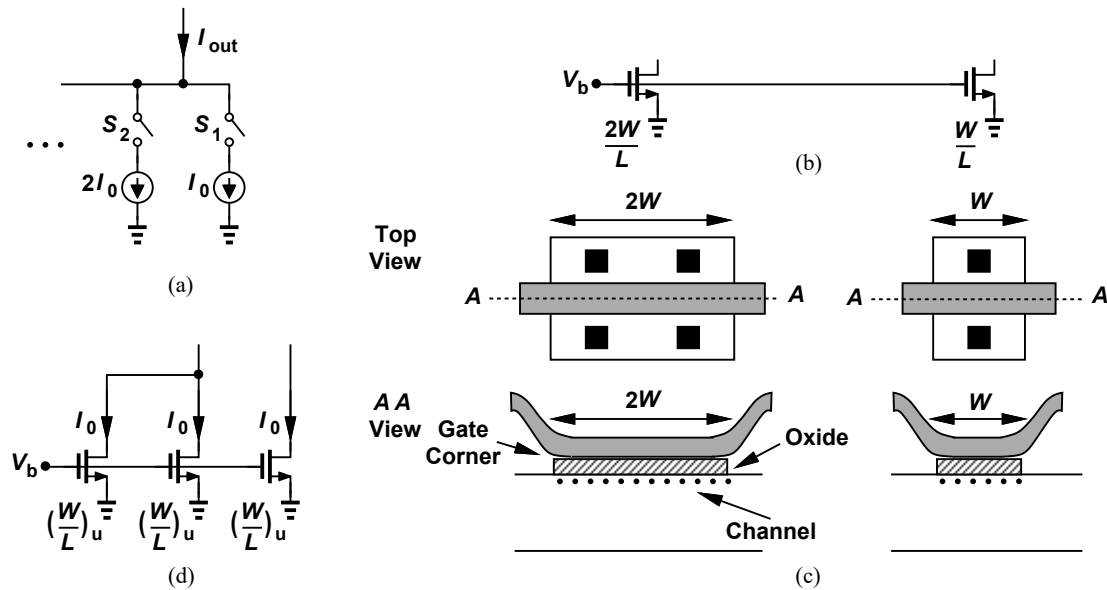


Figure 10.7 (a) Conceptual IDAC, (b) MOS width scaling for current sources, (c) problem of edge effects, and (d) proper scaling to avoid edge effects.

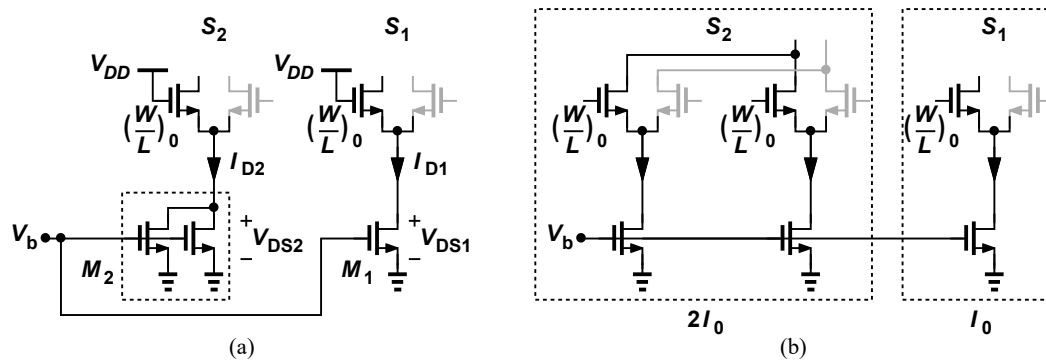


Figure 10.8 IDAC implementation (a) without, and (b) with current switch scaling.

10.2.2 Random Mismatches

Let us quantify the random mismatch between the two nominally identical current sources shown in Fig. 10.9(a). Our analysis of mismatches in Chapter 6 determined their contributions to the input-referred offset

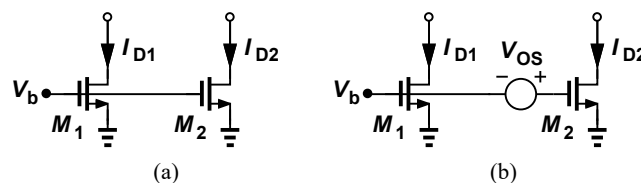


Figure 10.9 (a) Two nominally identical current sources, and (b) inclusion of offset voltage.

voltage, but it can also yield the output *current* mismatch. In fact, viewing the circuit as in Fig. 10.9(b), we

have $\Delta I_D = I_{D1} - I_{D2} = g_m V_{OS} \approx [(2I_D)/(V_{GS} - V_{TH})]V_{OS}$. Moreover, from Eq. (6.18),

$$\Delta I_D^2 = I_D^2 \left(\frac{\Delta\beta}{\beta} \right)^2 + \frac{4I_D^2}{(V_{GS} - V_{TH})^2} \Delta V_{TH}^2. \quad (10.4)$$

It follows that

$$\frac{\Delta I_D^2}{I_D^2} = \left(\frac{\Delta\beta}{\beta} \right)^2 + \frac{4\Delta V_{TH}^2}{(V_{GS} - V_{TH})^2}. \quad (10.5)$$

Recall from Section 6.1.2 that both $\Delta\beta/\beta$ and ΔV_{TH} decrease as the transistor channel area increases. From (6.27) and (6.28),

$$\frac{\Delta I_D^2}{I_D^2} = \frac{A_\beta^2}{WL} + \frac{4}{(V_{GS} - V_{TH})^2} \frac{A_{V_{TH}}^2}{WL}. \quad (10.6)$$

This result suggests that the current mismatch can be lowered by increasing WL and the overdrive voltage. We still face one question: how much $\Delta I_D/I_D$ is acceptable for a desired DAC resolution? We return to this point in Section 10.8.

10.3 Binary-Weighted and Thermometric IDACs

The IDAC of Fig. 10.5 employs “binary-weighted” current sources to convert a digital binary input to an analog output. This architecture suffers from a critical issue that makes it ill-suited to resolutions above roughly 5 or 6 bits.

In the presence of mismatches, the DAC exhibits a large DNL for the transition from 011 to 100. This point can be understood by noting that, in Fig. 10.5, for $D_{in} = 011$, the $2I_0$ and I_0 current sources are activated, whereas for $D_{in} = 100$, the $4I_0$ source is. If this source incurs a mismatch of ΔI , then, as explained in Example 8.4, a DNL equal to ΔI appears in the characteristic. In other words, to ensure $\text{DNL} < 0.5 \text{ LSB}$, we must guarantee that the mismatch between the MSB current source and the *sum* of the remaining current sources is less than 0.5 LSB. This matching requirement proves difficult to meet at high resolutions. For example, a 12-bit design demands that the MSB current, $2048I_0$, and the sum of the LSB currents, $2047I_0$, match to within $0.5I_0$. This translates to an allowable mismatch of $0.5/2048 = 0.024\%$, a challenging constraint. We generalize the result for an N -bit binary-weighted IDAC as

$$\left| \frac{\Delta I}{I_0} \right| < \frac{1}{m}, \quad (10.7)$$

where $m = 2^N$. We have seen in Section 8.2.1 that a mismatch greater than 1 LSB can lead to nonmonotonicity. This occurs in our 12-bit example if the MSB source is equal to $2046.9I_0$ while the remaining sources add up to $2047I_0$.

Thermometric DACs The foregoing issue can be remedied through the use of “thermometric” (also called “unary”) IDACs. Presented in Fig. 10.10 is a 3-bit example: seven *identical* current-switching branches are tied to the output node and driven by an input thermometer code, $T_7T_6 \cdots T_1$. Recall from Section 8.3 that the number of ONEs in the thermometer code is equal to the decimal value, e.g., $T_7T_6T_5T_4T_3T_2T_1 = 0001111$ signifies an input equal to 4. In other words, as the input value goes from 0 to 1 to 2 to 3, etc., we have in Fig. 10.10, $I_{out} = 0$, $I_{out} = I_1$, $I_{out} = I_2 + I_1$, $I_{out} = I_3 + I_2 + I_1$, etc.

Returning to our 12-bit example, we recognize that, as D_{in} goes from $011 \cdots 1$ to $100 \cdots 0$, the DAC simply adds one more unit to an output current of $2047I_0$. DNL is thus equal to the difference between this unit and I_0 , remaining less than 1 LSB if

$$\left| \frac{\Delta I}{I_0} \right| < 0.5. \quad (10.8)$$

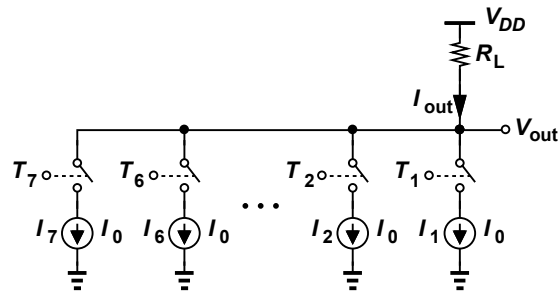


Figure 10.10 Thermometric DAC.

This result stands in sharp contrast to Eq. (10.7).

Another advantage of thermometric coding is that, for a 1-LSB input increment, the output always *increases*, i.e., the characteristic is guaranteed to be monotonic. For example, if the input goes from 3 to 4, only I_4 turns on and increases I_{out} .

Example 10.3

Seven nominally-equal current sources, I_1 – I_7 , are used to form a binary-weighted or a thermometric DAC [Fig. 10.11(a)]. We have $I_1 = I_2 = I_3 = I_0$, but, unfortunately, $I_4 = \dots = I_7 = I_0 - \Delta I$, where $\Delta I \approx 0.3I_0$. Plot the input-output characteristics of the two DACs.

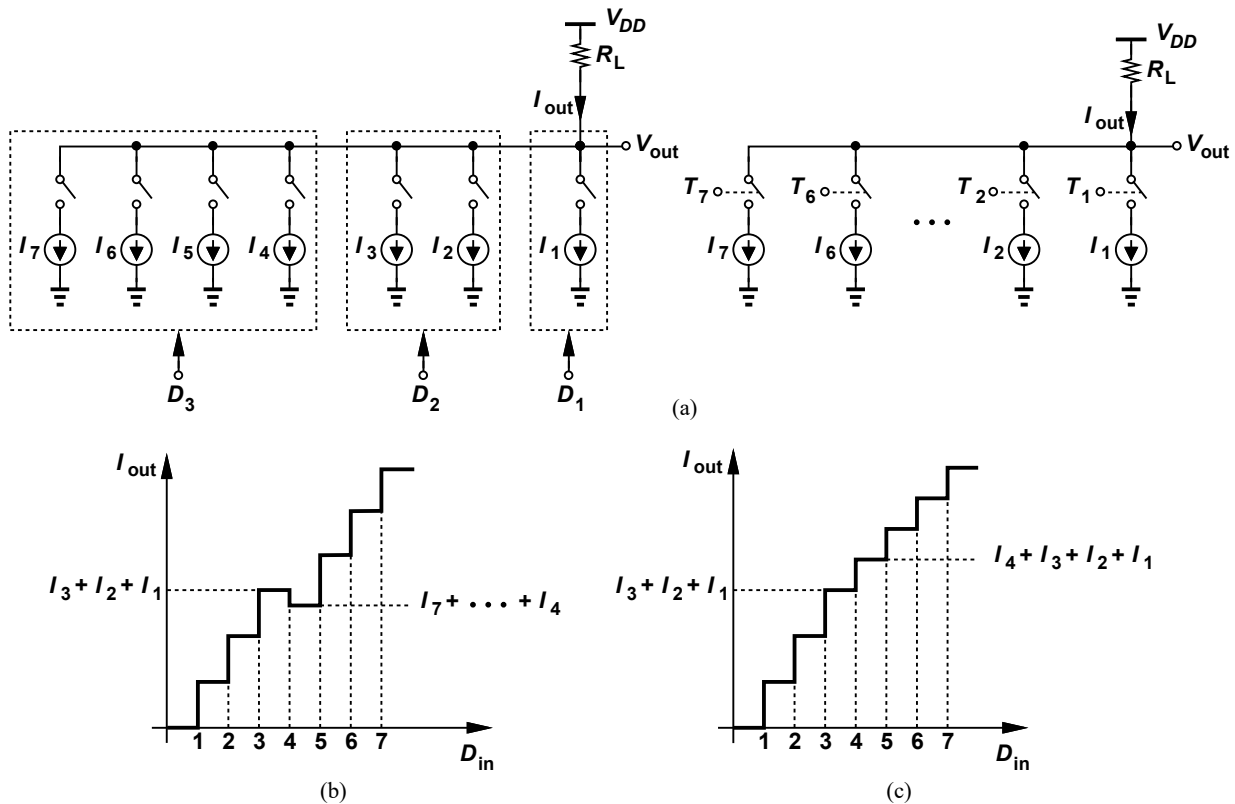


Figure 10.11 (a) Binary and thermometric groupings of seven current sources, (b) effect of mismatch on the former, and (c) effect of mismatch on the latter.

Solution

For the binary-weighted DAC, the output current jumps from $I_3 + I_2 + I_1 = 3I_0$ to $I_7 + \dots + I_4 = 2.8I_0$ as D_{in} goes from 3 to 4 [Fig. 10.11(b)]. By contrast, the thermometric DAC output simply rises from $I_3 + I_2 + I_1 = 3I_0$ to $I_4 + I_3 + I_2 + I_1 = 3.7I_0$, displaying a monotonic behavior.

The thermometric DAC's matching requirements are greatly relaxed as we only wish each current source in Fig. 10.10 to be sufficiently close to 1 LSB. As mentioned above, the mismatch between the current sources should be less than 50% for $DNL < 0.5$ LSB. Of course, the INL dictates tighter matching constraints.

Example 10.4

Compare the hardware requirements of binary-weighted and thermometric DACs.

Solution

As noted in Section 10.2, an N -bit binary-weighted architecture must employ $2^N - 1$ units for proper matching. Thus, the two architectures require approximately the same amount of hardware. Stand-alone thermometric DACs need additional logic to convert the binary input to a thermometer code, a small area penalty. The logic can consume substantial power at high speeds.

INL Calculation Suppose an N -bit thermometric DAC must achieve an INL of less than 0.5 LSB at mid-scale. We wish to estimate the corresponding matching requirements.

As shown in Fig. 10.12, the output current experiences random errors but is a monotonic function of

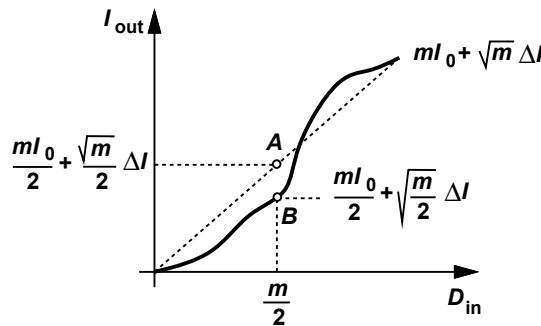


Figure 10.12 Computation of INL due to random mismatches.

D_{in} . We pass a line through the end points and seek the difference at the mid-scale. For $D_{in} = m/2$, approximately half of the current sources are turned on. Denoting the standard deviation (SD) of one source by ΔI , we recall from Section 6.1 that $m/2$ sources exhibit an SD of $\sqrt{m/2}\Delta I$, where $m = 2^N$. Moreover, the full-scale current is equal to mI_0 with an SD of $\sqrt{m}\Delta I$.

We may be tempted to find the INL as follows. The straight line yields a mid-scale value of $(m/2)I_0 + (\sqrt{m}/2)\Delta I$ whereas the actual characteristic gives $(m/2)I_0 + (\sqrt{m/2})\Delta I$. The difference should then be the INL. But this is not correct because subtracting two SDs assumes they represent completely *correlated* variables. Similarly, adding the variances assumes they are completely uncorrelated. In this case, we have neither: the first $m/2$ current sources contribute to both the mid-scale and the full-scale errors. It is possible to deal with these issues carefully but we take a different approach.

Let us assume that the full-scale current is approximately equal to the ideal value, mI_0 . The INL at the mid-scale is thus given by $(m/2)I_0 + \sqrt{m/2}\Delta I - (m/2)I_0 = \sqrt{m/2}\Delta I$. For this error to be

less than $0.5 \text{ LSB} = I_0/2$, we have

$$\frac{\Delta I}{I_0} < \frac{1}{\sqrt{2m}}. \quad (10.9)$$

This result merits two remarks. First, writing $\Delta I < I_0/\sqrt{2m}$ and hence $\sqrt{m}\Delta I < I_0/\sqrt{2}$, we recognize that the random component in the full-scale value is indeed much less than mI_0 for a large m . So our approximation for the full scale is valid. Second, the matching imposed by (10.9) is much less stringent than that by (10.7). That is, thermometric DACs are much simpler to design than binary-weighted topologies. For example, a 12-bit thermometric DAC can accommodate a mismatch of $1/\sqrt{2 \times 4096} = 1.1\%$ rather than $1/4096 = 0.024\%$.

Example 10.5

Does the condition stipulated by (10.9) apply to binary-weighted DACs as well?

Solution

Yes, it does. Recall from Section 10.2 that an N -bit binary-weighted DAC also incorporates $2^N - 1$ switching units to avoid deterministic mismatches. Thus, for a mid-scale binary input, $D_{in} = 100 \cdots 0$, half of the units are activated—just as in the thermometric topology. Of course, this condition does not guarantee monotonicity.

Table 10.1 summarizes our findings, underscoring the great advantage of thermometric DACs in terms of tolerable mismatches. We also note that, if the condition $\text{DNL} < 0.5 \text{ LSB}$ is met, so is the condition $\text{INL} < 0.5 \text{ LSB}$.

	Binary Weighted	Thermometric
DNL < 0.5 LSB	$\frac{\Delta I}{I_0} < \frac{1}{2^N}$	$\frac{\Delta I}{I_0} < 0.5$
INL < 0.5 LSB	$\frac{\Delta I}{I_0} < \frac{1}{\sqrt{2^{N+1}}}$	$\frac{\Delta I}{I_0} < \frac{1}{\sqrt{2^{N+1}}}$

Table 10.1 DNL and INL considerations in binary and thermometric DACs.

10.4 R-2R IDACs

The large number of switching units in binary-weighted DACs leads to large area consumption. It is possible to realize binary weighting in a more compact form by means of an “R-2R ladder.” Shown in Fig. 10.13(a) is a 4-bit example. The key point here is that the current sources are *equal*, avoiding the exponential dependence of the area upon the resolution. Note the periodic R-2R ladder structure, except that the rightmost resistor is equal to R .

To understand the operation, we assume S_1 – S_4 are on and prove that $I_{out} \propto I_4 + I_3/2 + I_2/4 + I_1/8$. Let us construct the Norton equivalent of I_1 and the two resistors attached to it [Fig. 10.13(b)]. This section can be modeled by a current of $I_1/2$ and a resistance of $2R$ (why?). Next, we include I_2 and its ladder resistors [Fig. 10.13(c)], obtaining a Norton equivalent consisting of $I_2/2 + I_1/4$ and $2R$. We then predict that the overall DAC reduces to a current source given by $I_4 + I_3/2 + I_2/4 + I_1/8$ and an output resistance equal to R [Fig. 10.13(d)] (why?).

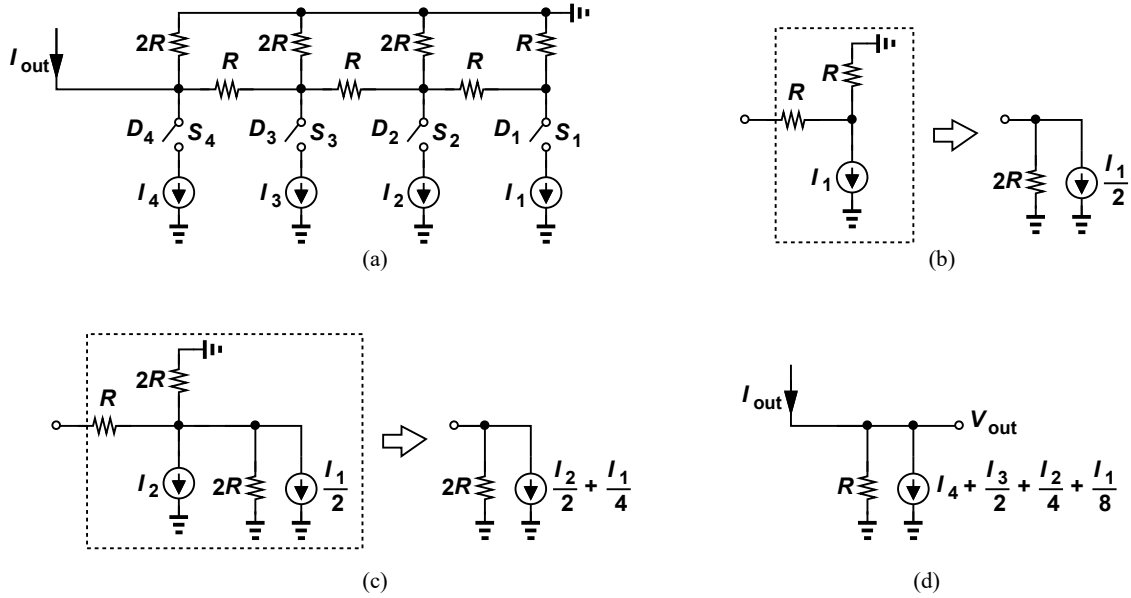


Figure 10.13 (a) Simple R-2R ladder, (b) reduction of rightmost section to a Norton equivalent, (c) one more reduction, and (d) overall equivalent circuit.

The IDAC of Fig. 10.13(a) merits four remarks. First, for a resolution of N bits, the (single-ended) circuit requires N unit current sources and $3N - 2$ unit resistors. Second, the DAC can be readily converted to a differential topology by realizing the switches as differential pairs and duplicating the R-2R ladder. Third, the binary weighting inherent in the DAC implies the same DNL and matching issues identified for the architecture shown in Fig. 10.5 and formulated by (10.7). For this reason, we limit the use of this architecture to about 4 bits.³ Fourth, the Norton model in Fig. 10.13(d) implies that $V_{out} = -R(I_4 + I_3/2 + I_2/4 + I_1/8)$; that is, the circuit also provides an output voltage, and I_{out} need not flow through another resistor.

In the presence of channel-length modulation, the IDAC of Fig. 10.13(a) suffers from deterministic mismatch. This issue arises because the R-2R ladder yields different voltages across the current sources. We formulate this mismatch in Problem 10.6. Nonetheless, the resulting nonlinearity is acceptable for resolutions up to 4 bits.

10.5 Segmented IDACs

We have seen that R-2R ladders afford a compact DAC implementation but limit the resolution to about 4 bits. Thermometric DACs, on the other hand, allow high precisions while requiring a large number of unit cells. We then surmise that combining the two architectures can reduce the complexity. The result is called a “segmented” DAC.

We demonstrate this point by a 10-bit example. The MSB current sources must be implemented as a thermometric array so as to relax the matching requirements. For example, the 7 MSBs can be realized in this form. Let us return to Fig. 10.13(d) and note that I_4 has a unity scaling factor with respect to the other current sources. We can then replace this current with 127 units, as shown in Fig. 10.14. The Norton model in Fig. 10.13(d) yields

$$V_{out} = -R \left(nI_0 + D_3 \frac{I_3}{2} + D_2 \frac{I_2}{4} + D_1 \frac{I_1}{8} \right), \quad (10.10)$$

³Since resistors typically exhibit smaller mismatches than current sources, the R-2R ladder itself does not limit the precision here.

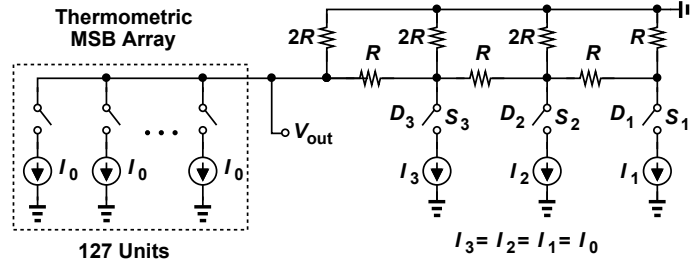


Figure 10.14 Segmented IDAC.

where n denotes the value of the thermometer code applied to the MSB array. We examine V_{out} as the overall digital input starts from zero: V_{out} changes from zero to $-RI_1/8 = -RI_0/8$ to $-RI_2/4 = -RI_0/4$, etc., until all three LSB sources turn on and $V_{out} = -7RI_0/8$. Now, we recognize that the ladder presents a Norton resistance equal to R to the MSB array. For the next code, $D_{in} = 8$, one unit in the MSB array turns on while other sources are off, yielding $V_{out} = -RI_0$. For the following seven codes, this source remains on and I_1 - I_3 play the previous role, eventually delivering $V_{out} = -R(I_0 + 7I_0/8) = -15RI_0/8$ at $D_{in} = 15$. The next code turns I_1 - I_3 off and turns on one more source in the MSB array so that $V_{out} = -2RI_0$. The full-scale voltage is equal to $-R(127 + 7/8)I_0$. We observe that 10 bits of resolution have been achieved through the use of 130 switching units.

It is instructive to determine the matching required of the current sources in the foregoing design. We first study the DAC's DNL. Consider a code transition such as $D_{in} = 00 \cdots 0111 \rightarrow 00 \cdots 1000$, where the LSB array turns off and one source in the MSB array turns on. Ideally, the output voltage goes from $-R(7I_0/8)$ to $-RI_0$ (Fig. 10.15). We thus require that the standard deviation (SD) of the difference between these

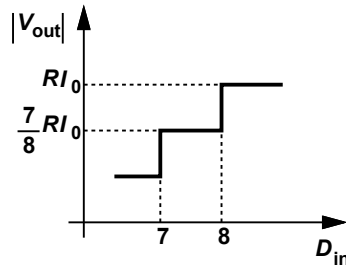


Figure 10.15 Segmented DAC output change.

two values be sufficiently small, e.g., less than 0.5 LSB. Let us examine the output current increment, $I_T - (I_3/2 + I_2/4 + I_1/8)$, where I_T denotes the unit in the MSB array. The four current sources appearing in this equation, namely, I_T , I_3 , I_2 , and I_1 , have a mean value of I_0 and a standard deviation of ΔI . Since they are uncorrelated, we find the overall SD by squaring and adding the SDs and taking the square root:

$$\Delta I_{out} = \sqrt{\Delta I^2 + \left(\frac{\Delta I^2}{4} + \frac{\Delta I^2}{16} + \frac{\Delta I^2}{64} \right)} \quad (10.11)$$

$$= \frac{\sqrt{85}}{8} \Delta I. \quad (10.12)$$

This is the DNL due to random mismatches. Noting that the LSB of the 10-bit DAC is equal to $I_0/8$, we wish the error to remain below 0.5 LSB:

$$\frac{\sqrt{85}}{8} \Delta I < \frac{I_0}{16} \quad (10.13)$$

and hence

$$\frac{\Delta I}{I_0} < \frac{1}{2\sqrt{85}} \approx 5.4\%. \quad (10.14)$$

Let us now turn to the INL of the segmented DAC and determine the maximum tolerable mismatch. To this end, we must repeat the calculations in Section 10.3, namely, we must assume an SD of ΔI for each current source, compute the full-scale current, pass a straight line through the end points, and obtain the difference between this line and the actual characteristic. The full-scale current is given by $127I_0 + I_3/2 + I_2/4 + I_1/8$. Recall from Section 10.3 that we can approximate the full-scale current by $128I_0$. If only the MSB array is considered, the mid-scale current has a mean value of $64I_0$ and an SD of $\sqrt{64}\Delta I$, departing from the straight line by $\sqrt{64}\Delta I$. For this INL error to remain below 0.5 LSB:

$$\sqrt{64}\Delta I < \frac{I_0}{16} \quad (10.15)$$

and hence

$$\frac{\Delta I}{I_0} < \frac{I_0}{128} \approx 0.78\%. \quad (10.16)$$

As expected, the INL demands tighter matching. Note that this calculation does not take the LSB array into account because it negligibly contributes to the INL.

The reader may wonder why we have used an R-2R ladder instead of simply adding current sources equal to $I_0/2$, $I_0/4$, etc. This is because the transistor widths cannot be reduced arbitrarily.

10.6 Nonlinearity Due to Output Impedance of Current Sources

The finite output resistance of the switching units in an IDAC translates to DNL and INL. We formulate this phenomenon here, beginning with a single-ended topology.

Single-Ended DAC Consider the thermometric IDAC shown in Fig. 10.16(a) and suppose j out of m units

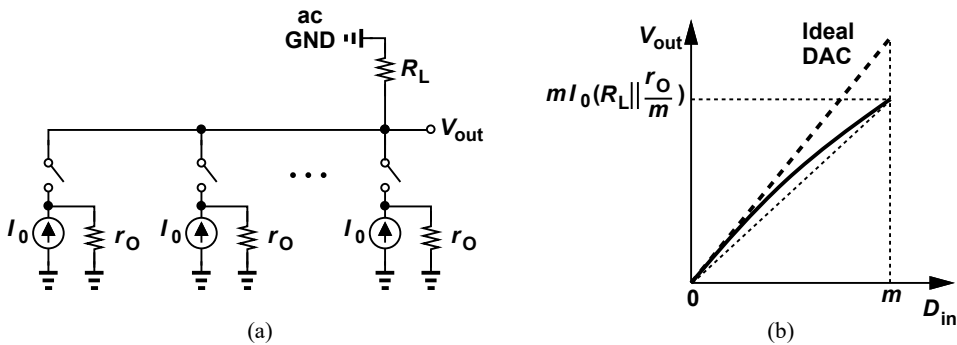


Figure 10.16 (a) IDAC including output impedance of current sources, and (b) resulting characteristic.

are activated. (For ease of illustration, the currents flow into the output node and R_L is tied to ground.) The output voltage is given by

$$V_{out}(j) = jI_0 \left(R_L \parallel \frac{r_O}{j} \right). \quad (10.17)$$

To compute the INL profile, we must find the straight line passing through the end points ($j = 0$ and $j = m$), and subtract it from $V_{out}(j)$ [Fig. 10.16(b)]. Since the line can be expressed as $jI_0[R_L \parallel (r_O/m)]$, the INL

profile is

$$\text{INL}_j = jI_0 \left(R_L \parallel \frac{r_O}{j} \right) - jI_0 \left(R_L \parallel \frac{r_O}{m} \right) \quad (10.18)$$

$$= jI_0 R_L \frac{(m-j)R_L r_O}{(jR_L + r_O)(mR_L + r_O)}. \quad (10.19)$$

As seen below, $r_O \gg mR_L$ in a well-designed DAC. It follows that $r_O \gg jR_L$ and

$$\text{INL}_j \approx \frac{I_0 R_L^2}{r_O} j(m-j). \quad (10.20)$$

The INL reaches a maximum of

$$\text{INL}_{\max} = \frac{m^2 I_0 R_L^2}{4r_O} \quad (10.21)$$

at $j = m/2$. To keep this below $0.5 \text{ LSB} \approx I_0 R_L/2$, we have

$$r_O > \frac{m^2 R_L}{2}. \quad (10.22)$$

Note that the assumption $r_O \gg mR_L$ is now valid if $m/2$ is large (why?).

It is interesting to note that, in Fig. 10.16(b), the gain of the DAC (the slope) drops as D_{in} increases. This is because the incremental resistance seen at the output falls. We say the DAC experiences “compression.”

The condition stipulated by (10.22) proves challenging for higher resolutions. The following example illustrates this point.

Example 10.6

An 8-bit IDAC must deliver a 0.5-V swing to a $50\text{-}\Omega$ load. Determine the minimum acceptable output resistance of each switching unit. Assume a single-ended design.

Solution

The unit current is equal to $10 \text{ mA}/256 = 39 \text{ }\mu\text{A}$. The tolerable output resistance is computed from (10.22) as $1.64 \text{ M}\Omega$. It is difficult to obtain such a high resistance from a single transistor and at this bias current.

We should remark that (10.22) also applies to binary-weighted DACs. This is because, at mid-scale, about half of the switching units in the DAC are activated, the same scenario as in a thermometric architecture.

Example 10.7

A digital sinusoid given by $(m/2) \cos \omega_1 t + m/2$ is applied to the single-ended DAC of Fig. 10.16(a). Determine the output harmonics. Assume $r_O \gg mR_L$.

Solution

The situation is depicted in Fig. 10.17. Let us write (10.17) as

$$V_{out} = \frac{jI_0 R_L r_O}{jR_L + r_O} \quad (10.23)$$

$$= \frac{jI_0 R_L}{1 + \frac{jR_L}{r_O}}. \quad (10.24)$$

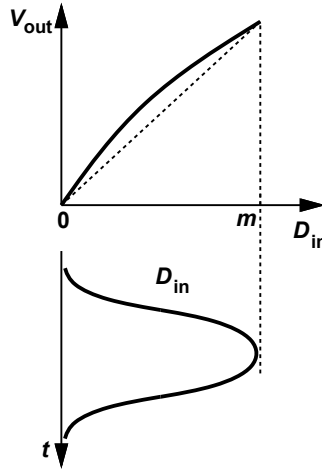


Figure 10.17 Illustration of harmonic distortion in a DAC.

It follows that

$$V_{out} \approx jI_0R_L \left(1 - \frac{jR_L}{r_O}\right). \quad (10.25)$$

Replacing j with $D_{in} = (m/2) \cos \omega_1 t + m/2$, we have

$$V_{out} \approx I_0R_L \left[\frac{m}{2} + \frac{m}{2} \cos \omega_1 t - \frac{R_L}{r_O} \left(\frac{m}{2} + \frac{m}{2} \cos \omega_1 t \right)^2 \right]. \quad (10.26)$$

Upon expanding the squared term, we note that the peak amplitude of the fundamental is given by $I_0R_Lm/2 - I_0R_L^2m^2/(2r_O) \approx I_0R_Lm/2$ (why?) and that of the second harmonic by $I_0R_L^2m^2/(8r_O)$. The normalized distortion emerges as

$$\text{HD}_2 = \frac{mR_L}{4r_O}. \quad (10.27)$$

The circuit generates only a second harmonic. This ratio can be viewed as one-fourth of the load resistance divided by the output resistance of m current sources. For the 8-bit DAC considered in the previous example, we can target $\text{HD}_2 = -50$ dB and hence $mR_L/(4r_O) = 0.32\%$. It follows that $r_O \approx 1$ M Ω .

An alternative method is to write V_{out} as the sum of the INL and the straight line passing through the end points and use Eq. (10.20) to obtain the output (Problem 10.10).

Differential DAC We expect that the output impedance of current sources leads to less nonlinearity in differential IDACs, thereby relaxing the design of the switching units. In such DACs, when j current sources are switched to one side, $m - j$ current sources are switched to the other [Fig. 10.18(a)]. We thus have

$$V_{out} = V_A - V_B \quad (10.28)$$

$$= jI_0 \left(R_L \parallel \frac{r_O}{j} \right) - (m - j)I_0 \left(R_L \parallel \frac{r_O}{m - j} \right). \quad (10.29)$$

It follows that

$$V_{out} = I_0R_Lr_O^2 \frac{2j - m}{(jR_L + r_O)[(m - j)R_L + r_O]}. \quad (10.30)$$

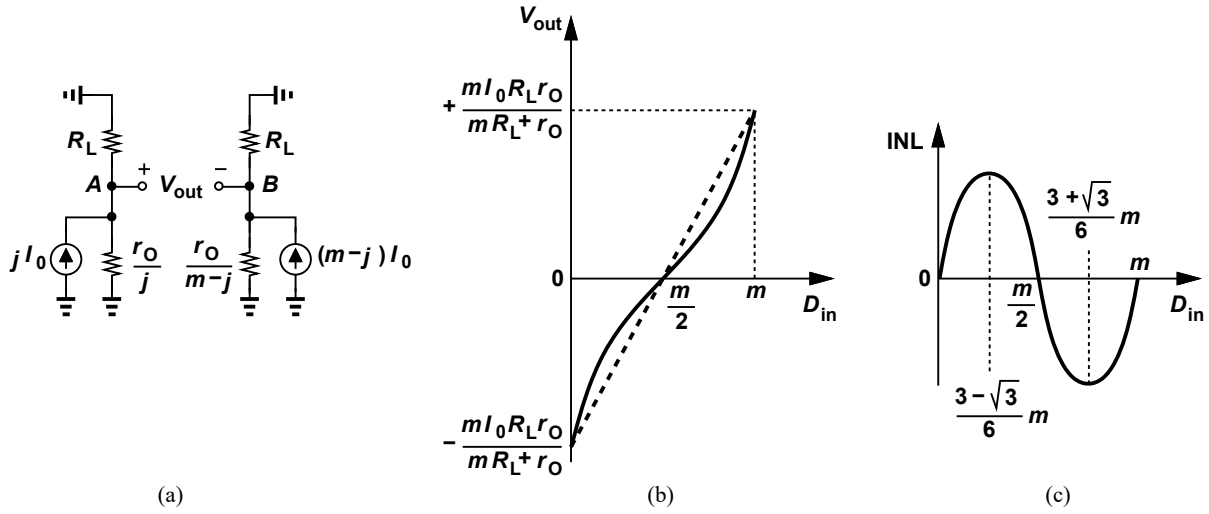


Figure 10.18 (a) Differential IDAC, (b) its nonlinearity due to the current sources' output impedance, and (c) corresponding INL profile.

Note that $V_{out}(j=0) = -V_{out}(j=m) = -mI_0R_Lr_O/(mR_L+r_O) = -mI_0[R_L||r_O]$. These points define the straight line passing through the two ends of the characteristic [Fig. 10.18(b)]. In contrast to the single-ended behavior in Fig. 10.16(b), this DAC's gain *increases* toward the end points. The reader can prove this result. Here, $1 \text{ LSB} \approx 2I_0R_L$ (why?).

The INL is given by

$$\text{INL}_j = \frac{I_0R_Lr_O^2(2j-m)}{(jR_L+r_O)[(m-j)R_L+r_O]} - \frac{I_0R_Lr_O(2j-m)}{mR_L+r_O}, \quad (10.31)$$

where the second term represents the straight line. We thus have

$$\text{INL}_j = \frac{I_0R_Lr_O(2j-m)(j^2-mj)R_L^2}{(jR_L+r_O)[(m-j)R_L+r_O](mR_L+r_O)}. \quad (10.32)$$

Note that $\text{INL}_j = 0$ at $j = 0, m/2$, and m . We assume $r_O \gg mR_L$, obtaining

$$\text{INL}_j \approx \frac{I_0R_L^3}{r_O^2}(2j-m)(j^2-mj). \quad (10.33)$$

The INL reaches a maximum at

$$j = \frac{3 \pm \sqrt{3}}{6}m, \quad (10.34)$$

as shown in Fig. 10.18(c). From (10.33),

$$|\text{INL}_{max}| = \frac{I_0R_L^3}{r_O^2} \frac{\sqrt{3}}{18} m^3. \quad (10.35)$$

For this error to remain below $0.5 \text{ LSB} \approx I_0R_L$, we require that

$$r_O > \sqrt{\frac{\sqrt{3}}{18}} m^{3/2} R_L. \quad (10.36)$$

Compared to (10.22), this constraint is a function of $m^{3/2}$ and hence less stringent. For example, with $R_L = 50 \Omega$, an 8-bit differential DAC necessitates $r_O > 63.5 \text{ k}\Omega$, a factor of 26 lower than that in Example 10.6. It also delivers twice the voltage swing.

Example 10.8

Repeat Example 10.7 for the differential IDAC of Fig. 10.18(a). Assume $r_O \gg mR_L$.

Solution

We write (10.30) as

$$V_{out} \approx I_0 R_L \frac{2j - m}{\left(1 + \frac{jR_L}{r_O}\right) \left[1 + \frac{(m-j)R_L}{r_O}\right]} \quad (10.37)$$

$$\approx I_0 R_L (2j - m) \left(1 - j \frac{R_L}{r_O}\right) \left[1 - \frac{(m-j)R_L}{r_O}\right]. \quad (10.38)$$

We can identify a j^3 term and hence a third harmonic. But we can also see a j^2 term. Does this mean that the differential DAC generates a second harmonic? No, it does not. Note that the input, $j = (m/2) \cos \omega_1 t + m/2$, is symmetric around $m = j/2$ (Fig. 10.19). As explained in Chapter 2, such an input does not lead to even-order harmonics. From (10.38), we write

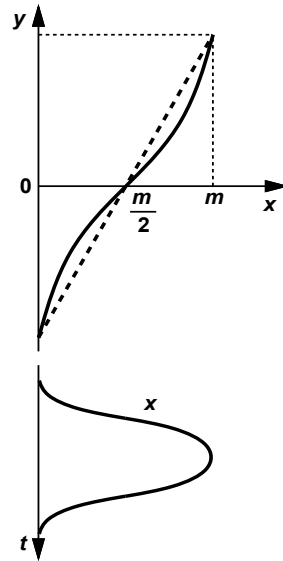


Figure 10.19 Effect of symmetry on distortion.

$$V_{out} \approx I_0 R_L \left[-2j^3 \frac{R_L^2}{r_O^2} + 3j^2 m \frac{R_L^2}{r_O^2} + j \left(2 - 2m \frac{R_L}{r_O} - m^2 \frac{R_L^2}{r_O^2} \right) - m + m^2 \frac{R_L}{r_O} \right]. \quad (10.39)$$

This result can be viewed as a polynomial, $\alpha_0 + \alpha_1 j + \alpha_2 j^2 + \alpha_3 j^3$. We are interested in the cubic term (for the third harmonic) and the linear term (for the fundamental). With $j = (m/2) \cos \omega_1 t + m/2$, we recall from Chapter 2 that $\alpha_3 j^3$ yields a third harmonic amplitude equal to $(\alpha_3/4)(m/2)^3$. Equation (10.39) gives $\alpha_3 = -2I_0 R_L^3 / r_O^2$ and hence a third harmonic amplitude of $-I_0 R_L^3 m^3 / (16r_O^2)$. To find the fundamental amplitude, we neglect all terms in (10.39) but the $2j$ term, obtaining $2I_0 R_L (m/2)$. It follows that the normalized

third-order harmonic is given by

$$HD_3 \approx \left(\frac{mR_L}{4r_O} \right)^2. \quad (10.40)$$

Comparing (10.27) and (10.40), we observe the dramatic reduction of distortion as we go from a single-ended topology to a differential realization.

Raising the Output Resistance The output resistance problem can be alleviated through the use of two techniques. First, the current source in each unit can be realized by a cascode structure [Fig. 10.20(a)], exhibiting an output resistance of

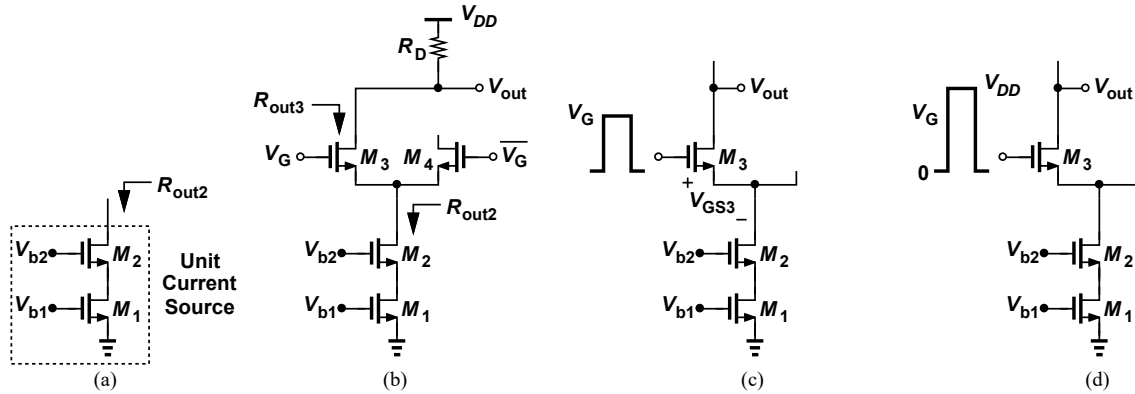


Figure 10.20 (a) Cascode current source, (b) current-steering unit using a cascode, (c) input swing for switching device to be in saturation, and (d) input swing reaching V_{DD} .

$$R_{out2} = (1 + g_{m2}r_{O2})r_{O1} + r_{O2}, \quad (10.41)$$

where body effect is neglected. If M_2 is a relatively long transistor, $g_{m2}r_{O2}$ is in the range of 10 to 20, raising the output resistance considerably. Of course, M_2 occupies additional voltage headroom and limits the DAC output voltage swing.

Second, the switching differential pair itself can act as a cascode, too. Shown in Fig. 10.20(b), such a structure ensures that M_3 and M_4 operate in saturation, requiring that $V_{out} \geq V_G - V_{TH}$. Under this condition, R_{out2} is boosted by a factor of $1 + g_{m3}r_{O3}$:

$$R_{out3} = (1 + g_{m3}r_{O3})R_{out2} + r_{O3}. \quad (10.42)$$

Transistors M_3 and M_4 typically use the minimum channel length so as to minimize capacitances at their three terminals. Thus, $g_{m3}r_{O3}$ is in the range of 5 to 10. The allowable output voltage swing is now given by:

$$V_{DS1} + V_{DS2} + V_{DS3} < V_{out} < V_{DD}. \quad (10.43)$$

For example, with $V_{DS} \approx 100$ mV and $V_{DD} = 1$ V, we obtain a single-ended output swing of $0.7 V_{pp}$. But how should V_G and $\overline{V}_G$ be generated to accommodate this swing? With $V_{TH} \approx 0.3$ V, we note from Fig. 10.20(c) that $V_G = V_{TH} + V_{DS3} + V_{DS2} + V_{DS1} \approx 0.6$ V, dictating that the data path deliver pulses with a height *less than* V_{DD} . This creates difficulty in the data path design, e.g., demanding CMOS inverters running from a 0.6-V supply.

To simplify the data path, we allow V_G and $\overline{V}_G$ to reach V_{DD} [Fig. 10.20(d)]. The cost is a smaller output swing, from V_{DD} to $V_{DD} - V_{TH}$ if M_3 must remain in saturation.

10.7 Transistor Sizing

We have seen in previous sections that the current sources in an IDAC introduce nonlinearity through two imperfections: random mismatches and the finite output resistance. In this section, we decide how the switching units' transistors should be sized to meet a given linearity requirement. As an example, we target a differential 8-bit thermometric DAC delivering 0.5-V_{pp} single-ended swings to $50\text{-}\Omega$ loads. From the previous section, we require an output resistance of $63.5\text{ k}\Omega$.

The design begins with the main current-source transistor(s), e.g., M_1 in Fig. 10.21. Striving for simplicity,

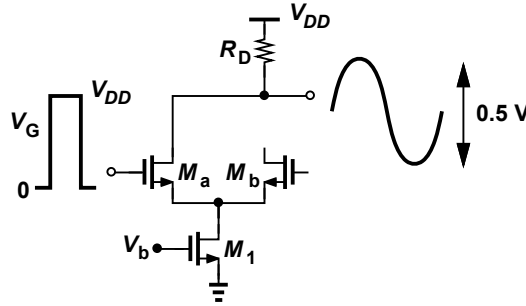


Figure 10.21 Unit cell for sizing considerations.

we prefer to avoid a cascode device and also allow rail-to-rail swings at the gates of the switching devices. With a 0.5-V output swing and $V_{TH} \approx 0.3\text{ V}$, these devices thus enter the triode region and do not much boost the output resistance. In other words, we must ensure that M_1 itself exhibits $r_O > 63.5\text{ k}\Omega$ with a drain current of $39\text{ }\mu\text{A}$. This is accomplished by increasing the *length* to the point of meeting this bound. If this endeavor leads to excessively long devices, we must resort to a cascode topology.

Next, we select the width of M_1 in Fig. 10.21 so as to fulfill the random mismatch condition stipulated by Eq. (10.9): $\Delta I/I_0 < 1/\sqrt{2m}$. For $m \approx 256$, we have $\Delta I/I_0 < 4.4\%$. Returning to Eq. (10.6), we have

$$(4.4\%)^2 = \frac{A_\beta^2}{WL} + \frac{1}{(V_{GS} - V_{TH})^2} \frac{A_{V_{TH}}^2}{WL}. \quad (10.44)$$

The overdrive, $V_{GS} - V_{TH}$, is dictated by the output voltage swing and is typically around 100 mV . We can then choose W so as to meet this condition if A_β and $A_{V_{TH}}$ are known. For example, if $A_\beta \approx 0.05\text{ }\mu\text{m}$ and $A_{V_{TH}} \approx 2.5\text{ mV} \cdot \mu\text{m}$, we require $WL = 1.6\text{ }\mu\text{m}^2$.

10.8 Matrix Architecture

10.8.1 Basic Idea

Our study of IDACs in the previous sections indicates a large number of switching units for high resolutions, most of which must be driven by a thermometer code so as to ease matching requirements. Suppose a segmented IDAC incorporates 63 units in its MSB array. How do we arrange these units in the layout so that the digital signals can be distributed efficiently? Where do we place the binary-thermometer decoder(s)? Figure 10.22(a) offers one possibility where the switching units and their decoders are arranged in a linear array. The input digital lines, $D_6 \cdots D_1$, travel horizontally to reach all of the decoders. It can be seen that this layout style leads to very disproportionate x and y dimensions, possibly creating difficulty in a larger system. We will also see in Section 10.8.3 that such a long array would face severe IR drops on the ground or supply line.

A better solution is the matrix architecture introduced in [1]. Illustrated in Fig. 10.22(b) for our 6-bit thermometric example, the idea is to arrange the units in an 8×8 matrix so as to achieve comparable x and y dimensions. Moreover, the thermometer code is produced in two steps, thus greatly simplifying the routing of the signals. We now describe the details of this structure.

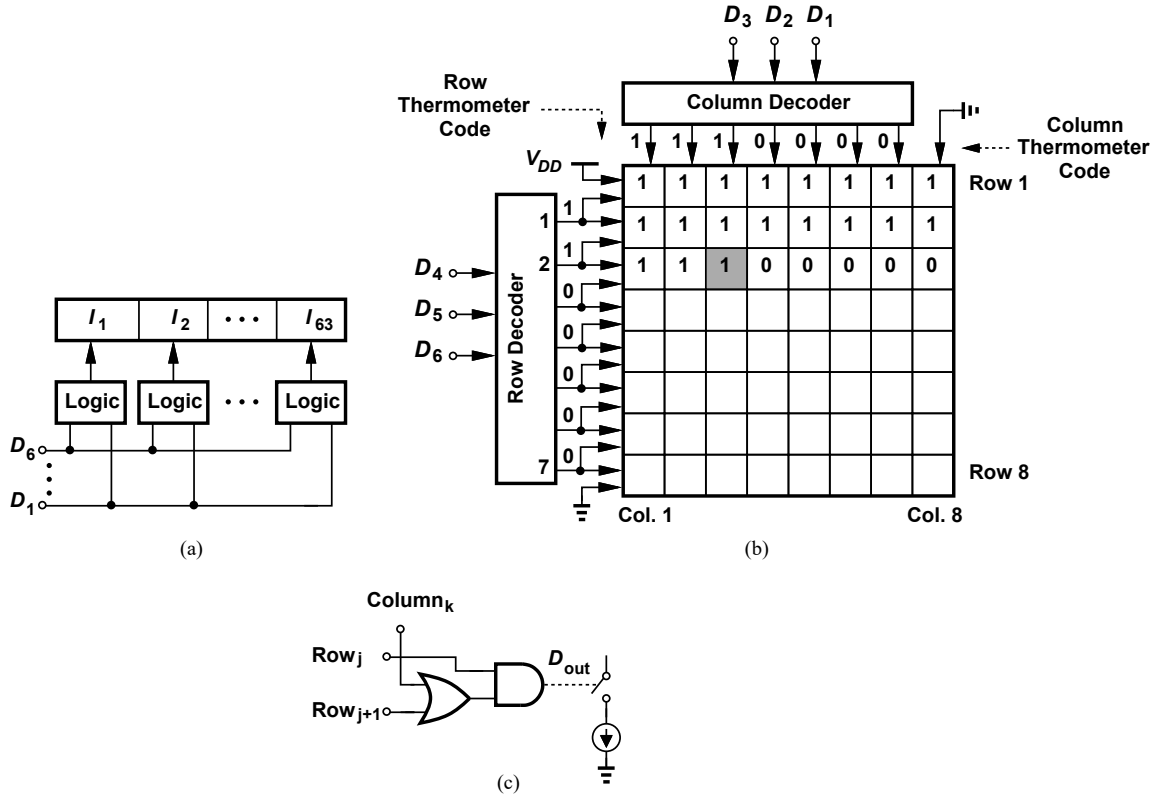


Figure 10.22 (a) Linear placement of current units, (b) matrix architecture, and (c) local logic implementation.

The matrix architecture consists of 63 cells, each containing both a switching unit and a “local” decoder. In addition, the DAC includes a “row thermometer decoder” and a “column thermometer decoder.” The former receives the binary MSBs, $D_6 D_5 D_4$, and generates eight thermometer signals that travel across each row. The latter is driven by the LSBs, $D_3 D_2 D_1$, and delivers its thermometer outputs to the eight columns. The local decoder in each cell senses the row and column thermometer codes and decides whether the current source in that cell must be on or off. The design of the decoders is described in Chapter 8. We assume that the column thermometer code “fills up” from left to right. Also, the row thermometer code fills up from top to bottom.

To design the local decoder logic, we first recognize that the rows can have one of three possibilities: (1) all of the cells in a row are on, (2) all of the cells in a row are off, and (3) some of the cells are on and some are off. The local logic thus emerges as shown in Fig. 10.22(c), where $Column_k$ denotes a column thermometer code bit and Row_j and Row_{j+1} represent two row thermometer code bits. The current source is switched on if $Row_j = 1$ and $Row_{j+1} = 1$, or if $Row_j = 1$ and $Column_k = 1$. That is, if both row values are equal to 1, then $D_{out} = 1$, and if $Row_{j+1} = 0$, then $Column_k$ determines the output.⁴ For example, the cell shaded in gray receives $Column_3 = 1$, $Row_2 = 1$, and $Row_3 = 0$, generating $D_{out} = 1$. The next cell on the right, on the other hand, senses $Column_4 = 0$, delivering $D_{out} = 0$.

The unit cells in Fig. 10.22(b) match well only if they all “see” the same environment. For example, each cell in the middle of the array is surrounded by four others—but not the cells on the edges. To ensure matching, we place “dummy” cells on all four sides, as shown in Fig. 10.23. These cells are identical to the main ones but with all of their input and output terminals grounded.

⁴Note that output Row_j of the decoder feeds cells in both Row_j and Row_{j+1} .

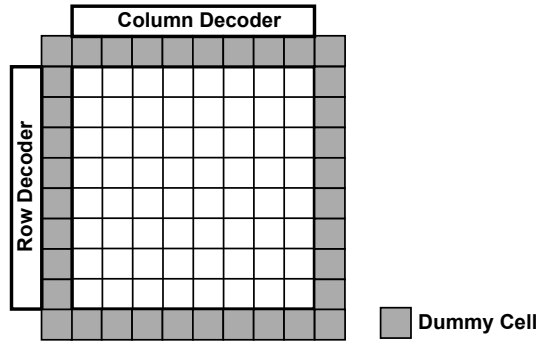


Figure 10.23 Matrix architecture including dummy cells.

10.8.2 Effect of Gradients

IDAC arrays, and in particular, the matrix architecture face an important issue: gradients along the x and y dimensions tend to accumulate and produce large INL. Recall from Section 9.4 that the value of a component can vary systematically from one side of the chip to the other side, an effect often modeled by a linear change. The following example studies this phenomenon for an IDAC.

Example 10.9

The IDAC of Fig. 10.22(b) experiences a linear gradient along the x -axis, i.e., the current source in column j is given by $I_0 + (j - 1)\Delta I$. Sketch the input-output characteristic and compute the gain error and the maximum INL.

Solution

Suppose the digital input begins from zero and rises in steps of 1 LSB, turning on the current sources in row number 1 from the left to the right. When j units are enabled, the output current is

$$I_{out} = I_0 + (I_0 + \Delta I) + (I_0 + 2\Delta I) + \cdots + [I_0 + (j - 1)\Delta I] \quad (10.45)$$

$$= jI_0 + \frac{j(j - 1)\Delta I}{2}. \quad (10.46)$$

We observe that the error accumulates quadratically [Fig. 10.24(a)], reaching $28\Delta I$ at the end of the row.

Figure 10.24(b) sketches the overall behavior as more rows are enabled: the mismatches accumulated in the rows add up, yielding increasingly greater errors. Note that each row contributes a total current of $8I_0 + 28\Delta I$, except for row number 8, which contains seven units adding up to $7I_0 + 21\Delta I$. But, for the sake of simplicity, we assume the last row also consists of eight units. The full-scale current is therefore equal to $8(8I_0 + 28\Delta I) = 64I_0[1 + 7\Delta I/(2I_0)]$. It follows that the normalized gain error is $7\Delta I/(2I_0)$.

To compute the INL profile, we first express the straight line passing through the end points as $jI_0[1 + 7\Delta I/(2I_0)]$, where j varies from zero to 64. We also recognize that points A , B , C , etc., in Fig. 10.24(b) lie on this line (why?). Thus, the INL profile emerges as shown in Fig. 10.24(c).⁵ The maximum absolute value can be found simply from the first row: we return to Eq. (10.46) and subtract $jI_0[1 + 7\Delta I/(2I_0)]$ from it. We then have

$$\text{INL}_j = \frac{j(j - 1)\Delta I}{2} - j\frac{7\Delta I}{2} \quad (10.47)$$

$$= \frac{j(j - 8)}{2}\Delta I. \quad (10.48)$$

⁵Note that we have defined the INL as the actual characteristic minus the straight line; hence the *negative* peaks.

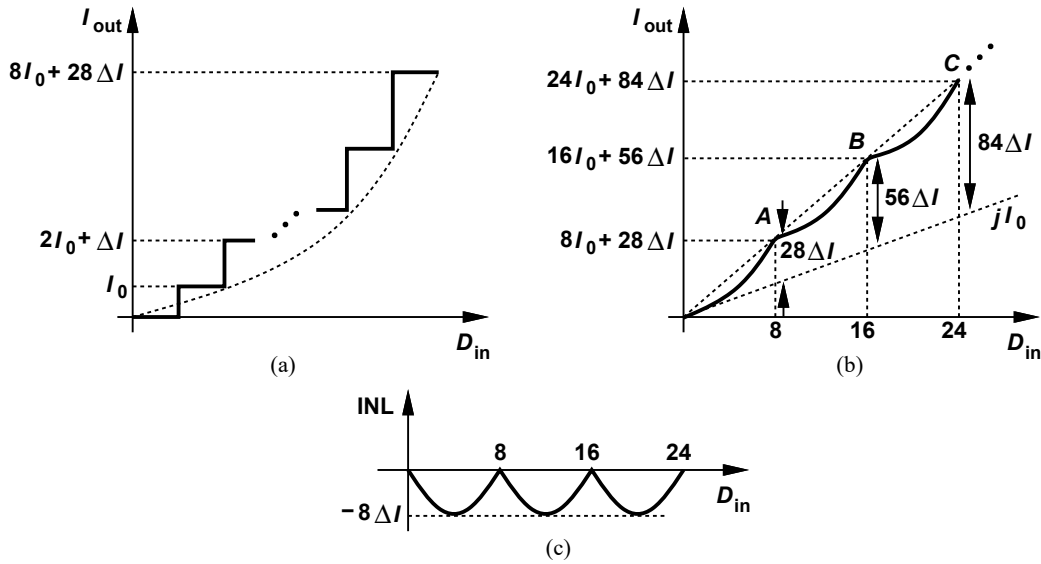


Figure 10.24 (a) Effect of gradient on one row's characteristic, (b) effect on overall characteristic, and (c) resulting INL profile.

As expected, the maximum occurs at $j = 4$ and has an absolute value of $8\Delta I$. This error should be maintained below 0.5 LSB of the system. In Problem 10.14, we extend this analysis to a case where gradients are present along both x and y dimensions.

It is instructive to examine the effect of gradients on the DNL as well. Recall from Chapter 8 that $DNL_j = A_{j+1} - A_j$. For the case studied in Example 10.9, we observe that the first transition in Fig. 10.24(a) is equal to I_0 , the second equal to $I_0 + \Delta I$, the third equal to $I_0 + 2\Delta I$, etc. The DNL thus rises to a maximum of $7\Delta I$ (Fig. 10.25), which must be kept below 0.5 LSB. We conclude that the INL and DNL requirements impose nearly the same constraint on ΔI , a point of contrast to the case of random mismatches (Section 10.3). It is possible to reduce the effect of gradients by modifying the sequence in which the units

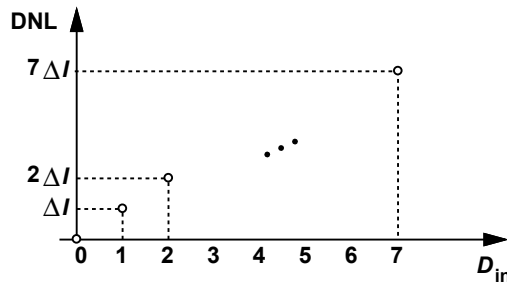


Figure 10.25 DNL profile for a linear gradient.

are enabled. As an example, consider the method depicted in Fig. 10.26(a), where the currents in a row are activated according to the order shown (and still by a thermometer code); first, the leftmost unit turns on, next the rightmost unit is added, etc. We wish to determine the INL in this case.

Let us plot the input-output characteristic for the first row [Fig. 10.26(b)]. We observe that I_{out} is still equal to $8I_0 + 28\Delta I$ for $D_{in} = 8$. Thus, the overall DAC exhibits the same gain error as found above, and the straight line still follows the expression $jI_0[1 + 7\Delta I/(2I_0)]$. The INL profile is plotted in Fig. 10.26(c), exhibiting a peak value of $-3.5\Delta I$. The new sequence therefore reduces the maximum INL by about a factor

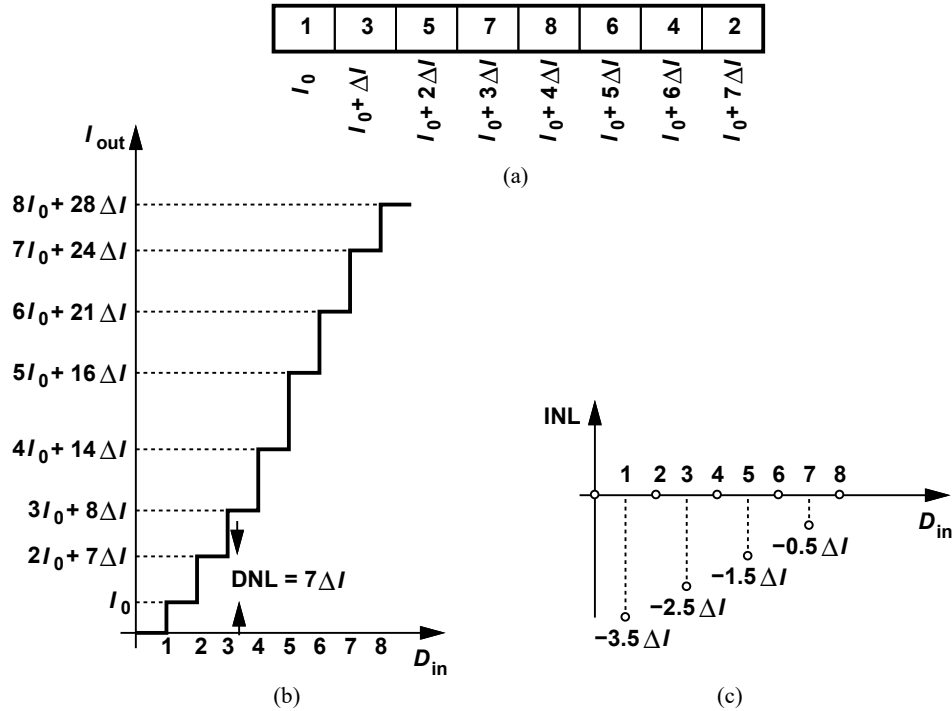


Figure 10.26 (a) Unit placement for reducing effect of gradient, (b) corresponding characteristic (vertical transitions not to scale), and (c) resulting INL.

of 2.3. Unfortunately, however, this sequence still introduces a peak DNL of $7\Delta I$ [Fig. 10.26(b)], which must be maintained below 0.5 LSB.

10.8.3 Signal and Power Distribution

The distribution of signal and power lines across large DAC arrays faces a number of issues that can degrade the performance. Let us first consider the digital (thermometric) signals traveling horizontally and vertically in Fig. 10.22(b). Suppose the output current must go from $16I_0$ to $17I_0$. The reader can show that, for the first unit in the next row to turn on, the column thermometer code must change from $111 \dots 1$ to $100 \dots 0$ while the row thermometer code goes from $1100 \dots 0$ to $1110 \dots 0$.

How does the DAC respond if these transitions do not occur at exactly the same time? Consider two possibilities. First, the column code changes before the row code does. That is, for some time, the former is equal to $100 \dots 0$ while the latter remains at $1100 \dots 0$. As shown in Fig. 10.27(a), the first row still turns on all of its units because the local decoder senses Row_1 and Row_2 thermometer codes. In the second row, on the other hand, only one unit is enabled. This “race” condition causes I_{out} to fall by $7I_0$ before it rises by I_0 .

The second case occurs if the column code changes after the row code does. Depicted in Fig. 10.27(b), this misalignment turns on all of the units in the third row for a short time, forcing I_{out} to rise by $7I_0$ before it falls to its steady-state value.

Other race conditions can also appear among the digital signals. For example, a code change experiences a finite delay as it travels horizontally or vertically within the matrix.

In order to reduce the effect of race conditions, the unit cell can incorporate a latch that is driven by the global clock. Illustrated in Fig. 10.28, the idea is to present the output of the local logic to the current source after the row and column codes have settled. If all of the cells are clocked simultaneously, their output currents change at the same time, creating no glitches. This method, of course, assumes that the delays along the clock lines running across the DAC array are small, or at least smaller than that of the digital signals. Depending on the actual design and layout, this may or may not be true. Thus, the decision to include latches within the

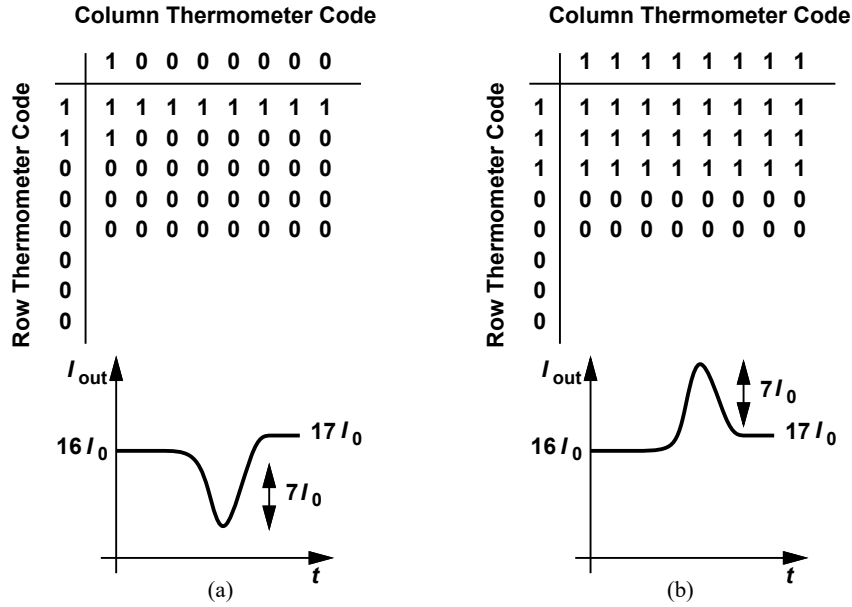


Figure 10.27 IDAC response when (a) column code changes first, and (b) row code changes first.

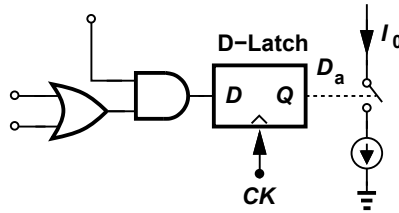


Figure 10.28 Addition of latch to local logic.

unit cells must be made after careful extraction and simulation of the layout parasitics.

Another source of error in IDACs is the voltage drop along the ground line tied to the current sources.⁶ Consider the arrangement shown in Fig. 10.29(a), where the global ground is attached on the left and r_u denotes the resistance of the ground wire within each DAC unit. The dc drops along the ground accumulate, introducing a large error in V_A . To estimate this error, we construct the equivalent circuit shown in Fig. 10.29(b), where the unit currents are assumed approximately equal to I_0 , as we expect of a good design. Beginning from the right side, we add up the voltage drops for n units:

$$V_A = I_0 r_u + 2I_0 r_u + \cdots + (n-1)I_0 r_u \quad (10.49)$$

$$= \frac{n(n-1)}{2} I_0 r_u. \quad (10.50)$$

For example, in an 8×8 matrix employing one ground line per row, this error reaches a maximum of $28I_0 r_u$ at the end of every row.

We wish to compute the INL of the DAC due to this voltage drop. We first recognize some similarity between this phenomenon and the linear gradient in Section 10.8.2: in both cases, the error reaches a maximum at the end of the rows. We therefore need only derive the INL profile for one row. Returning to Fig.

⁶Or along the V_{DD} line if PMOS current sources are used.

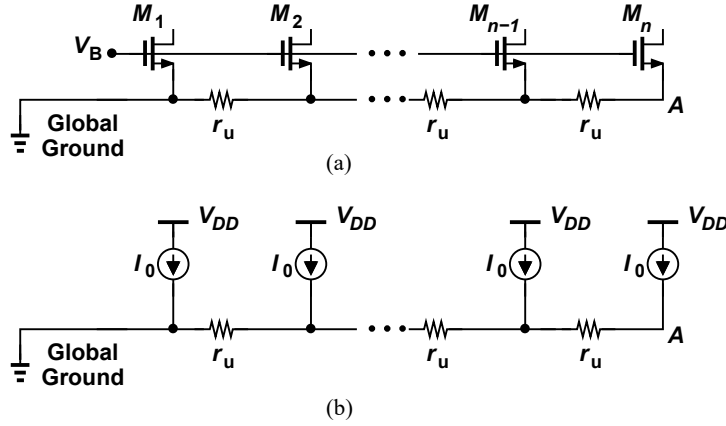


Figure 10.29 (a) Voltage drops along GND line, and (b) equivalent circuit.

In 10.29(a), we observe that M_1 and M_2 suffer from a gate-source voltage difference equal to $(n-1)I_0r_u$ (why?). Their current difference is then approximately equal to $g_m[(n-1)I_0r_u]$. It follows that $I_{D1} = I_0$, $I_{D2} = I_0 - g_m(n-1)I_0r_u$, $I_{D3} = I_0 - g_m(n-1)I_0r_u - g_m(n-2)I_0r_u$, ..., and $I_{Dn} = I_0 - g_mV_A = I_0 - g_m[n(n-1)/2]I_0r_u$. Figure 10.30 sketches the general behavior of the source voltages and drain currents as a function of the transistor number, j . Also sketched is the DAC output current.

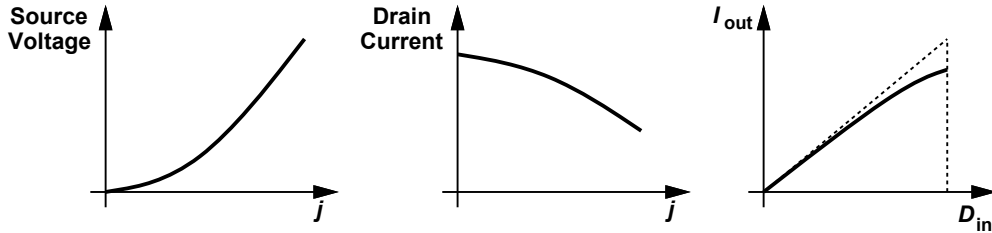


Figure 10.30 Illustration of how ground voltage drops lead to nonlinearity.

To calculate the INL profile, we write the drain current of transistor number j in Fig. 10.29(a) as

$$I_{Dj} = I_0 - g_m(n-1)I_0r_u - g_m(n-2)I_0r_u - \dots - g_m(n-j+1)I_0r_u \quad (10.51)$$

$$= I_0 - g_m(j-1)\frac{2n-j}{2}I_0r_u. \quad (10.52)$$

For an input digital value equal to j , $M_1, M_2, \dots, M_j$ turn on, delivering a current equal to

$$I_{out}(j) = jI_0 - \sum_{k=1}^j \left[g_m(k-1)\frac{2n-k}{2} \right] I_0r_u. \quad (10.53)$$

Noting that

$$\sum_{k=1}^j k^2 = \frac{j(2j+1)(j+1)}{6}, \quad (10.54)$$

In Problem 10.16, we obtain the INL profile. Let us proceed with a numerical example.

Example 10.10

A 10-bit segmented IDAC must deliver an output current of 10 mA. Configured as a 6-bit thermometric array and a 4-bit binary array, the DAC employs an 8×8 matrix architecture for the former. Compute the maximum allowable ground resistance per cell if the INL must be less than 0.5 LSB. Assume $V_{GS} - V_{TH} \approx 300$ mV for the current sources.

Solution

When all eight units in a row are enabled, we have from Eq. (10.53):

$$I_{out}(8) = 8I_0 - g_m I_0 r_u (7 + 13 + 18 + 22 + 25 + 27 + 28) = 8I_0 - 140g_m I_0 r_u = 8I_0(1 - 17.5g_m r_u). \quad (10.55)$$

Note a gain error of $17.5g_m r_u$. The straight line passing through the end points is then given by $I_0(1 - 17.5g_m r_u)j$. Moreover, the error in I_{out} is equal to $-g_m I_0 r_u$ multiplied by 7, 20, 38, 60, 85, 112, and 140 as j varies from 2 to 8 (Table 10.2). The table also shows the difference between the actual characteristic and the straight line, revealing a maximum INL of $-32.5g_m I_0 r_u$. With $I_0 \approx 10$ mA/64 and $g_m \approx 2I_0/(V_{GS} - V_{TH}) = 0.001 \Omega^{-1}$, this INL is equal to $-5.1 \times 10^{-6} r_u$, whose magnitude must be less than 10 mA/(2048). The tolerable r_u is then around 1Ω per unit cell. Of course, matrices with higher resolutions further tighten this constraint.

j	1	2	3	4	5	6	7	8
Error in I_{out} (Normalized to $-g_m r_u I_0$)	0	7	20	38	60	85	112	140
Straight-Line Value (Normalized to $g_m r_u I_0$)	17.5	35	52.5	70	87.5	105	122.5	140
Difference between Straight Line and Characteristic (Normalized to $-g_m r_u I_0$)	17.5	28	32.5	32	27.5	20	10.5	0

Table 10.2 Errors due to ground resistance.

Example 10.11

A student surmises that the effect of the ground voltage drop can be alleviated by “tapering” the width of the ground line. Illustrated in Fig. 10.31(a), the idea is to provide less resistance for the sections that carry a greater current. Is this method preferred over no tapering?

Solution

In this example, each section of the line sustains a voltage drop of $I_0 r_u$ (why?) and hence the rightmost unit incurs an error of $g_m(4I_0 r_u)$ with respect to the leftmost unit. Now, consider the untapered line in Fig. 10.31(b), where each section exhibits a resistance of $r_u/4$. The error between the first and the last units is given by $g_m(4 + 3 + 2 + 1)I_0(r_u/4) = g_m(2.5I_0 r_u)$, indicating a wide trace is preferred over a tapered one.

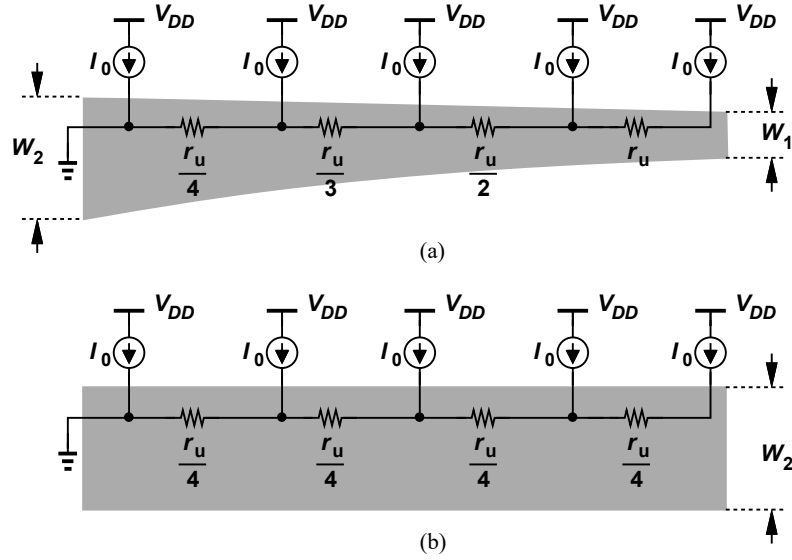


Figure 10.31 IDAC (a) with, and (b) without ground tapering.

The key point here is that, if the unit cell can accommodate a line width of W_2 for the ground, then a rectangular geometry with this width in fact yields less error than does a linearly-tapered structure.

10.9 Speed and Reference Bounce Issues

The speed of IDACs is determined by two fairly independent factors. First, the time constant at the output node, τ , places a lower bound on the settling time. As depicted in Fig. 10.32, each switching unit contributes a

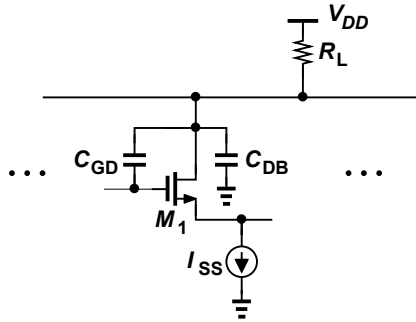


Figure 10.32 Unit cell showing capacitances at the output node.

gate-drain capacitance and a drain-bulk capacitance to the output node. The former is minimal if M_1 remains in saturation. We have

$$\tau = m(C_{GD} + C_{DB})R_L \quad (10.56)$$

for m units. In practice, the output wire also contributes significantly to τ . Note that the transistor serving as I_{SS} must be large enough to meet random mismatch constraints, but M_1 need only have sufficient width to carry I_{SS} with a reasonable V_{GS} . Thus, τ is typically small, yielding a fast settling behavior.

The second factor—the more critical one—is the reference bounce. Consider the situation shown in Fig. 10.33, where D_j and $\overline{D_j}$ switch, turning M_1 on and M_2 off. Recall from Fig. 10.6(c) that V_X drops momen-

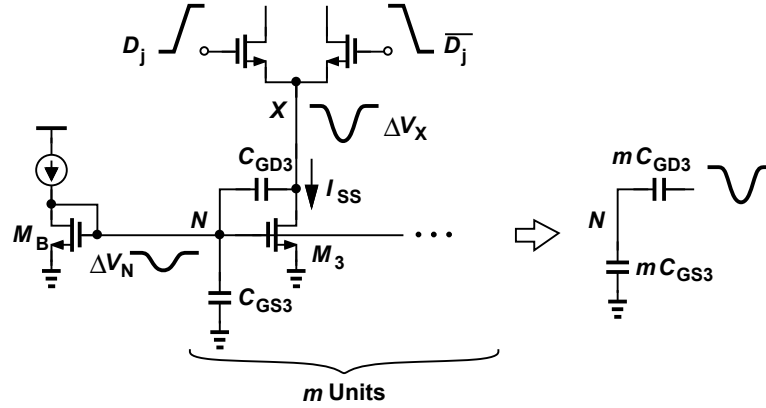


Figure 10.33 Effect of tail voltage swings on the gate bias voltage.

tarily during this transition. This drop couples through C_{GD3} to the common bias line, N , disturbing this voltage for *all* of the switching units. Consequently, the total output current fluctuates for some time. We say the reference, V_N , experiences a “bounce.”

Let us examine this phenomenon more closely. The worst case occurs when the digital input jumps from zero to its maximum or vice versa. We wish to compute the disturbance in V_N if V_X changes by ΔV_X . For m cells, the simple capacitive divider suggests that⁷

$$\Delta V_N = \frac{mC_{GD3}}{mC_{GD3} + mC_{GS3}} \Delta V_X \quad (10.57)$$

$$= \frac{C_{GD3}}{C_{GD3} + C_{GS3}} \Delta V_X. \quad (10.58)$$

This voltage change yields a total output current change equal to

$$\Delta I_{out} = mg_{m3} \Delta V_N \quad (10.59)$$

$$= \frac{2m\Delta V_X}{(V_{GS} - V_{TH})_3} \frac{C_{GD3}}{C_{GD3} + C_{GS3}} I_{SS}, \quad (10.60)$$

where $g_{m3} = 2I_{SS}/(V_{GS} - V_{TH})_3$. The key point here is that ΔV_X is equivalently multiplied by m . For example, if $m = 256$, $\Delta V_X = 10$ mV, $(V_{GS} - V_{TH})_3 = 200$ mV, and $C_{GD3} = 0.2C_{GS3}$, we have $\Delta I_{out} = 4.3I_{SS}$. That is, the total output current of $256I_{SS}$ experiences a glitch of $4.3I_{SS}$, increasing the settling time.

The problem of reference bounce in IDACs is somewhat similar to that in RDACs (Section 9.6). That is, we must decide whether or not node N in Fig. 10.33 should be bypassed to ground by a large capacitor. Let us consider the arrangement in Fig. 10.34, where the bypass capacitor, C_b , is much greater than $mC_{GD3} + mC_{GS3}$. Noting that $\Delta V_N \approx (mC_{GD3}/C_b)\Delta V_X$, we rewrite Eq. (10.59) as

$$\Delta I_{out} = mg_{m3} \Delta V_N \quad (10.61)$$

$$= \frac{2m^2 C_{GD3}}{C_b} \frac{\Delta V_X}{(V_{GS} - V_{TH})_3} I_{SS}. \quad (10.62)$$

⁷We assume that the change in ΔV_X lasts for a short time compared to the time constant at N . We therefore neglect the effect of M_B .

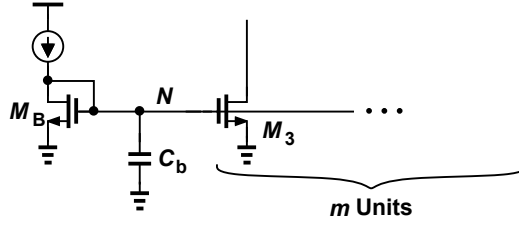


Figure 10.34 Addition of capacitance to gate like to reduce the bounce.

For this change to remain below $0.5 \text{ LSB} = I_{SS}/2$, we must have

$$C_b > (4m^2 C_{GD3}) \frac{\Delta V_X}{(V_{GS} - V_{TH})_3}. \quad (10.63)$$

For the previous numerical example, we have

$$C_b > 1.31 \times 10^4 C_{GD3}. \quad (10.64)$$

As explained in Section 10.7, the dimensions of the tail current source, M_3 , are dictated by both channel-length modulation and matching requirements. Thus, C_{GD3} can be in the range of 5 to 10 fF, yielding a C_b value from 66 pF to 130 pF. Such a large capacitor is inevitable if ΔI_{out} must be kept small. Alternatively, we can avoid the use of C_b altogether if the glitch in V_X subsides quickly and hence does not materially affect the output settling time. These two choices must be evaluated by careful simulations.

10.10 Dynamic Errors in IDACs

10.10.1 Output Impedance at High Frequencies

As observed in Section 10.6, the finite output resistance of the unit current sources in an IDAC leads to nonlinearity and harmonic distortion. For high-frequency operation, we must revisit this phenomenon while examining the output *impedance* of the units.

Consider the basic switching unit depicted in Fig. 10.35(a), where M_1 is on and in saturation. This tran-

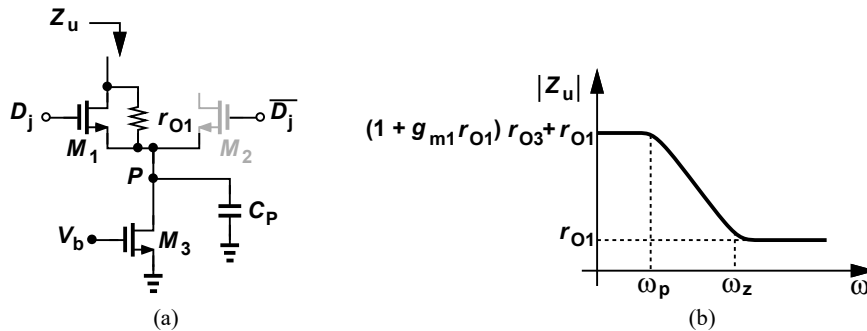


Figure 10.35 (a) Current unit with tail capacitance, and (b) its output impedance vs. frequency.

sistor and M_3 act as a cascode structure, but node P incurs a parasitic capacitance, C_P , equal to

$$C_P = C_{GD3} + C_{DB3} + C_{GS1} + C_{SB1} + C_{GS2} + C_{SB2}, \quad (10.65)$$

where the subscript B stands for “bulk.” Note that C_{GS2} is relatively small because M_2 is off. The impedance seen at the drain of M_1 is expressed as

$$Z_u = (1 + g_{m1}r_{O1})Z_T + r_{O1}, \quad (10.66)$$

where $Z_T = r_{O3} || [1/(C_{Ps})]$. That is,

$$Z_u = \frac{(1 + g_{m1}r_{O1})r_{O3} + r_{O1} + r_{O1}r_{O3}C_{Ps}}{r_{O3}C_{Ps} + 1}, \quad (10.67)$$

exhibiting a pole, ω_p , and a zero, ω_z . Sketched in Fig. 10.35(b), $|Z_u|$ falls as the frequency exceeds $\omega_p = 1/(r_{O3}C_P)$.

If the DAC operates well beyond ω_p , but also well below ω_z , and if $g_m r_O \gg 1$, we can approximate Z_u as

$$Z_u \approx \frac{g_{m1}r_{O1}r_{O3}}{r_{O3}C_{Ps}} \quad (10.68)$$

$$\approx \frac{g_{m1}r_{O1}}{C_{Ps}}. \quad (10.69)$$

As expected of a cascode, the impedance tied to the source of M_1 is “boosted” by a factor of $g_{m1}r_{O1}$. Alternatively, we can say the equivalent capacitance seen at the drain of M_1 is approximately equal to $C_P/(g_{m1}r_{O1})$.

We predict that the drop of the output impedance at high frequencies translates to distortion. Let us examine this effect for the thermometric IDAC shown in Fig. 10.36(a). Here, $C_a \approx C_P/(g_{m1}r_{O1})$, as suggested by (10.69). Suppose the DAC must generate a full-scale sinusoidal output. As shown in Fig. 10.36(b), for V_{out} near zero, only a few current sources are switched to the output and so are a few capacitances. For V_{out}

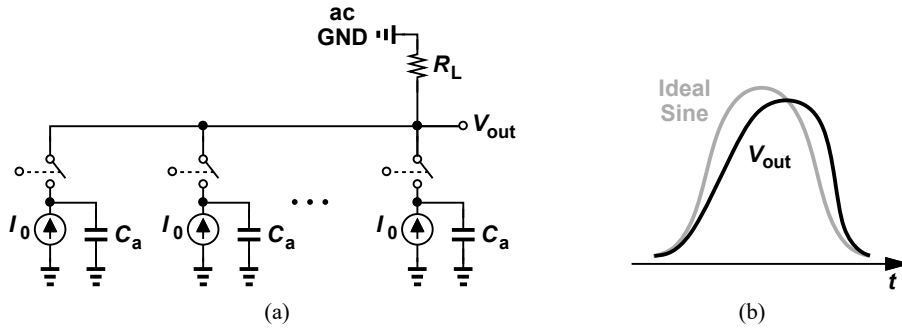


Figure 10.36 (a) Simplified diagram of an IDAC with tail capacitances, and (b) effect of these capacitances on a sinusoidal input.

near the peak, on the other hand, nearly all of the capacitances are switched to the output. This variation of the output time constant is reminiscent of that in sampling circuits (Chapter 3) and lends itself to a similar analysis. We say V_{out} experiences phase distortion and amplitude distortion. We perform the analysis for both single-ended and differential architectures.

Single-Ended DAC The topology of Fig. 10.36(a) can be simplified if we recognize that, as the digital code varies, some integer multiple of I_0 flows from the output node (a static effect) and some more capacitances are switched in or switched out (a dynamic effect). Thus, immediately after a code change, the circuit reduces to that in Fig. 10.37(a): a current of jI_0 is produced and j unit capacitances are switched in. Now suppose the digital input is given by $(m/2) + (m/2)\cos\omega_1 t$, where m denotes the total number of units. This input varies from zero to m .⁸ Drawing the Thevenin equivalent for greater intuition [Fig. 10.37(b)], we observe a

⁸This is a continuous-time approximation of the discrete-time input, j .

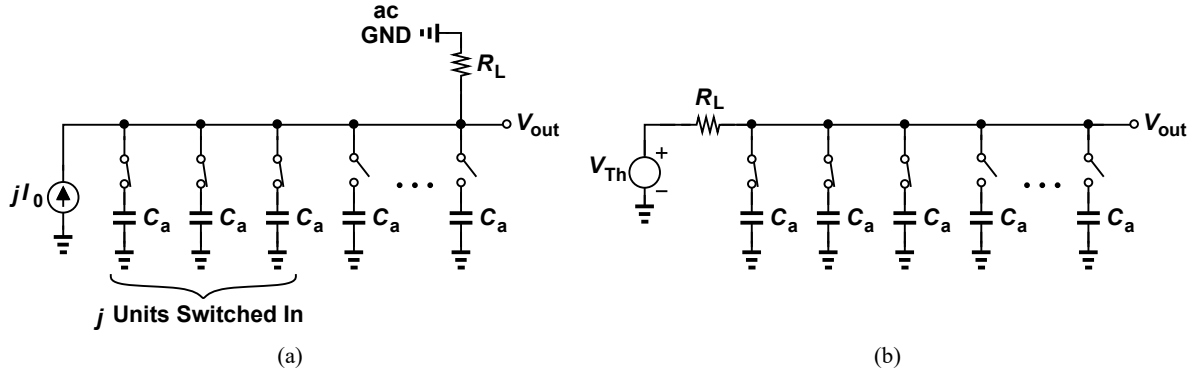


Figure 10.37 (a) Single-ended IDAC, and (b) its equivalent circuit.

low-pass filter whose corner frequency varies with the input, thus introducing input-dependent phase shift. We can express the total capacitance switched to the output as $[(m/2) + (m/2) \cos \omega_1 t] C_u$. Neglecting the effect of the capacitances on the output amplitude and following the analysis in Chapter 3, we have

$$V_{out} \approx \frac{m}{2} I_0 R_L + \frac{m}{2} I_0 R_L \cos \left\{ \omega_1 t - \tan^{-1} \left[\left(\frac{m}{2} + \frac{m}{2} \cos \omega_1 t \right) C_u R_L \omega_1 \right] \right\}. \quad (10.70)$$

For a small phase shift, θ , we have $\tan^{-1} \theta \approx \sin \theta \approx \theta$. Expanding the cosine term in (10.70) gives

$$V_{out} \approx \frac{m}{2} I_0 R_L + \frac{m}{2} I_0 R_L \cos \omega_1 t + \frac{m}{2} I_0 R_L \sin(\omega_1 t) \left(\frac{m}{2} \cos \omega_1 t \right) C_u R_L \omega_1 + \dots \quad (10.71)$$

The second harmonic amplitude normalized to that of the fundamental emerges as

$$\text{HD}_2 = \frac{m R_L C_u \omega_1}{4}. \quad (10.72)$$

For example, if $m = 64$, $R_L = 50 \, \Omega$, $C_u = C_P / (g_{m1} r_{O1}) = 2 \, \text{fF}$, and $\omega_1 = 2\pi(10 \, \text{GHz})$, then $\text{HD}_2 \approx 0.1 = -20 \, \text{dB}$. We recognize the severity of this problem: a 6-bit, 10-GHz IDAC should exhibit an HD_2 less than about $-38 \, \text{dB}$ so that its distortion is comparable to its SNR (Chapter 1). Note that (10.72) can be interpreted as the ratio of the input frequency, ω_1 , and four times the output bandwidth, $1/(m R_L C_u)$, in the presence of all of the unit capacitances.

The reader may wonder why we have neglected the effect of code-dependent capacitance upon the output amplitude. After all, Fig. 10.36(b) suggests that V_{out} experiences greater *attenuation* around its maximum value than around its minimum. The following example formulates this effect.

Example 10.12

For a simple first-order low-pass stage, the output amplitude is proportional to $1/\sqrt{1 + R^2 C^2 \omega^2}$. Assuming $R^2 C^2 \omega^2 \ll 1$, use this expression to estimate amplitude distortion in Fig. 10.37(b).

Solution

For the IDAC, the capacitance varies as $[(m/2) + (m/2) \cos \omega_1 t] C_u$. Thus,

$$V_{out} \approx \frac{mI_0 R_L}{2} + \frac{mI_0 R_L}{2\sqrt{1 + R_L^2 C_u^2 \omega_1^2 \left(\frac{m}{2} + \frac{m}{2} \cos \omega_1 t\right)^2}} \cos \omega_1 t, \quad (10.73)$$

where the phase variation is neglected. Noting that $1/\sqrt{1 + \epsilon} \approx 1 - \epsilon/2$ for $\epsilon \ll 1$, we have

$$V_{out} \approx \frac{mI_0 R_L}{2} + \frac{mI_0 R_L}{2} \left[1 - \frac{R_L^2 C_u^2 \omega_1^2}{2} \left(\frac{m}{2} + \frac{m}{2} \cos \omega_1 t \right)^2 \right] \cos \omega_1 t. \quad (10.74)$$

We now observe that the second term within the square brackets is proportional to $R_L^2 C_u^2 \omega_1^2$, and so are the harmonics obtained by expanding V_{out} . By comparison, the right-hand side of (10.72) is proportional to $R_L C_u \omega_1$ and hence dominant.

Differential DAC We predict that differential operation greatly reduces the distortion. The model in Fig. 10.37(a) is revised to that in Fig. 10.38, where $I_1 = (m/2) + (m/2) \cos \omega_1 t$ and $I_2 = (m/2) - (m/2) \cos \omega_1 t$. This means that the differential circuit delivers twice as much voltage swing as its single-ended counterpart. Let us first compute the third harmonic in V_{out1} . Interestingly, the approximations leading to Eq. (10.71) im-

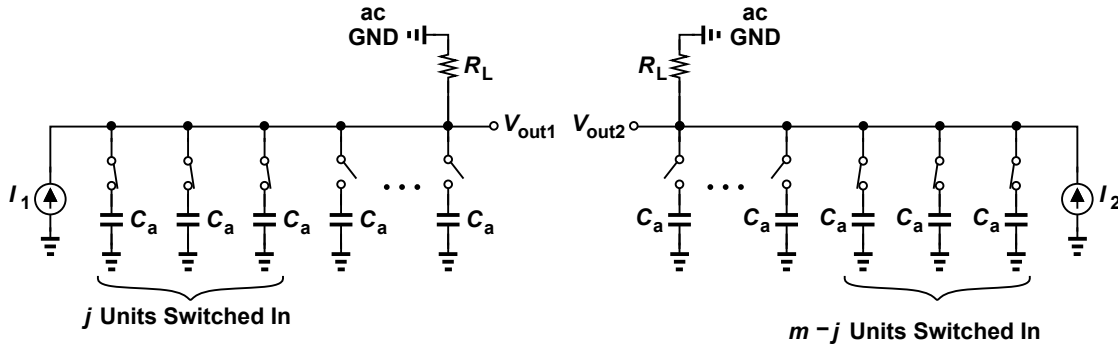


Figure 10.38 Differential IDAC model.

ply that the code-dependent phase does not yield HD_3 (why?). But how about the code-dependent amplitude? Returning to (10.74) and expanding the right-hand side, we have

$$V_{out1} \approx \frac{mI_0 R_L}{2} \cos \omega_1 t - \frac{mI_0 R_L}{2} \frac{R_L^2 C_u^2 \omega_1^2}{2} \left(\frac{m^2}{4} \cos^2 \omega_1 t \right) \cos \omega_1 t + \dots \quad (10.75)$$

Since $\cos^3 \omega_1 t = (\cos 3\omega_1 t + 3 \cos \omega_1 t)/4$, the peak amplitude of the third harmonic normalized to that of the fundamental is

$$HD_3 \approx \frac{1}{2} \left(\frac{m R_L C_u \omega_1}{4} \right)^2. \quad (10.76)$$

If $m = 64$, $R_L = 50 \Omega$, $C_u = 2$ fF, and $\omega_1 = 2\pi(10 \text{ GHz})$, we have $HD_3 = -46$ dB, a dramatic improvement over the single-ended example.⁹ Writing (10.75) for V_{out2} , the reader can readily show that

⁹But not necessarily sufficient in some applications.

the differential output exhibits the same HD_3 . In summary, we have observed that HD_2 arises from phase distortion and HD_3 from amplitude distortion.

Example 10.13

If we improve the approximations leading to (10.71), do we find a third harmonic resulting from phase distortion?

Solution

Let us write $\tan^{-1} \theta = \theta - \theta^3/3$. In Eq. (10.70), we must raise $[(m/2) + (m/2) \cos \omega_1 t] C_u R_L \omega_1$ to the power of 3. But this yields terms that are multiplied by $C_u^3 R_L^3 \omega_1^3$, which will be much less than the HD_3 in (10.76). Thus, phase distortion produces negligible third-order nonlinearity in this case.

10.10.2 Nonlinear Memory Effect

Recall that our earliest study of IDACs in Section 10.1 suggested the use of a differential pair for switching so that the voltage across the current source does not collapse. We also noted in Section 10.6 that one can choose the differential swings controlling the switching action to be less than rail-to-rail excursions so as to minimize the voltage change at the tail node. In this section, we delve into issues related to this node.

Consider the situation depicted in Fig. 10.39(a), where a particular value of the digital input has established

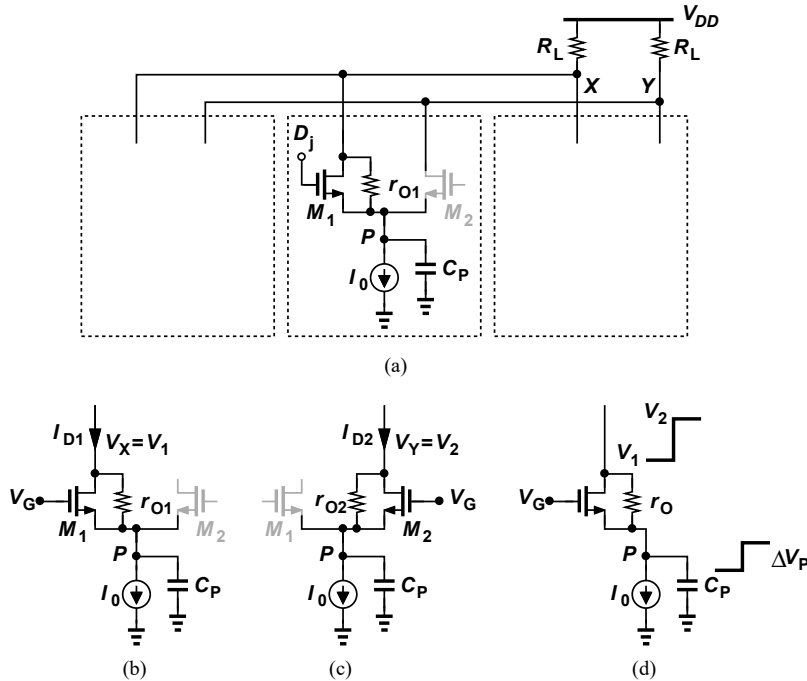


Figure 10.39 (a) Unit circuit showing tail capacitance, (b) tail node voltage when M_1 is on, (c) tail node voltage when M_2 is on, and (d) equivalent circuit.

a voltage of V_1 at node X . We examine one of the units, assuming that M_1 is on and resides in the saturation region. Owing to channel-length modulation, this transistor allows V_X to affect V_P . This can be seen by writing the total drain current in Fig. 10.39(b) as

$$I_{D1} = \frac{1}{2} \mu_n C_{ox} \left(\frac{W}{L} \right)_1 (V_G - V_P - V_{TH})^2 [1 + \lambda(V_X - V_P)]. \quad (10.77)$$

Since the value of I_{D1} is dictated by I_0 , changes in V_X cause V_P to change. The key point here is that C_P sustains a voltage that is a function of V_X .

Next, suppose the DAC's digital input changes and the unit in Fig. 10.39(a) switches, i.e., M_1 turns off and M_2 turns on. As shown in Fig. 10.39(c), the voltage on C_P now changes so as to agree with V_Y , thereby drawing charge from M_2 and hence from the output node. We conclude that C_P creates a “memory effect,” injecting into the output an amount of charge proportional to a previous digital value [2]. This injection can be estimated as follows. We note that the transition from Fig. 10.39(b) to Fig. 10.39(c) is equivalent to having a *single* transistor tied to P and changing its drain voltage by $V_2 - V_1$ [Fig. 10.39(d)]. Neglecting the current drawn by C_P for now, the reader can prove that V_P changes through r_O by an amount equal to

$$\Delta V_P = \frac{1/g_m}{1/g_m + r_O} (V_2 - V_1) \quad (10.78)$$

$$= \frac{1}{1 + g_m r_O} (V_2 - V_1), \quad (10.79)$$

where body effect is neglected. Consequently, C_P must draw charge in the amount of $C_P \Delta V_P$ from node Y . This charge depends on the *past* value of V_X .

Does this phenomenon introduce distortion? Let us visualize the charge delivery to V_Y as shown in Fig. 10.40(a). We know that the charge is proportional to V_X , hence $Q = \alpha V_X$, and this result is delayed by some

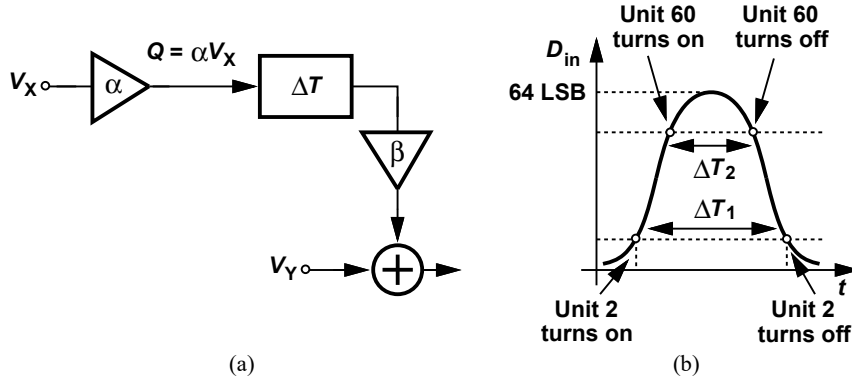


Figure 10.40 (a) Equivalent system for studying charge delivery, and (b) effect of different current sources turning on.

time, ΔT , before the differential pair in Fig. 10.39(a) switches. Finally, the charge is scaled by another factor β to translate to voltage. Thus, an amount equal to $\alpha\beta V_X(t - \Delta T)$ is added to V_Y .¹⁰ It appears that all of the operations in this model are linear and unable to create distortion. But we recognize that ΔT is a function of the digital output. To understand this point, suppose the digital input varies sinusoidally between zero and 64 LSB [Fig. 10.40(b)] while driving a thermometric array. We observe that unit 2 turns on soon after D_{in} rises from zero, remaining on for nearly one input period. Unit 60, on the other hand, turns on only briefly. Thus, ΔT is code-dependent, causing distortion.

The third-order distortion resulting from this phenomenon is given by [2]:

$$HD_3 = \frac{m\pi R_L C_P}{4(1 + g_m r_O)} f, \quad (10.80)$$

where m denotes the number of units and f the output frequency. For example, if $m = 64$, $R_L = 50 \Omega$, $C_P = 10$ fF, and $g_m r_O = 5$, then HD_3 reaches $4.2\% \equiv -27.5$ dB at 10 GHz.

The memory effect studied above can be alleviated by adding cascode transistors to the basic unit and drawing a small, constant current from them. The details are described in [3].

¹⁰This quantity is subsequently dissipated by R_L , but we ignore this for now.

References

- [1]. T. Miki et al, "An 80-MHz 8-bit CMOS D/A converter," *IEEE Journal of Solid-State Circuits*, vol. 21, pp. 983-988, Dec. 1986.
- [2]. P. Palmers and M. S. Stayeert, "A 10Bit 1.6-GS/s 27-mW current-steering D/A converter with 550-MHz 54-dB SFDR bandwidth in 130-nm CMOS," *IEEE Transactions on Circuits and Systems*, vol. 57, pp. 2870-2879, Nov. 2010.
- [3]. C. Lin and K. Bult, "A 10-b, 500-MSample/s CMOS DAC in 0.6 mm²," *IEEE Journal of Solid-State Circuits*, vol. 33, pp. 1948-1958, Dec. 1998.

Problems

- 10.1. We design the IDAC of Fig. 10.2 for a resolution of 8 bits. Determine the output voltage change when D_{in} rises from zero by 1 LSB or from 254 by 1 LSB.
- 10.2. We double the widths and the lengths of the current sources in Fig. 10.9(a). Determine by what factor their current mismatch drops.
- 10.3. In the 3-bit thermometric DAC of Fig. 10.10, current source I_2 is accidentally omitted. Plot the input-output characteristic and compute the offset, gain error, DNL, and INL.
- 10.4. We wish to design a 6-bit IDAC. Determine the matching requirements for binary-weighted and thermometric implementations.
- 10.5. We consider a 3-bit IDAC using the R-2R ladder technique depicted in Fig. 10.13(a). Assume each current source has an output resistance of r_O . Determine the overall Norton equivalent in a form similar to that in Fig. 10.13(d).
- 10.6. Compute the voltages across the current sources in Fig. 10.13(a) for $D_4D_3D_2D_1 = 0011, 0111$, and 1111 .
- 10.7. Consider a 2-bit version of the DAC shown in Fig. 10.13(a) and suppose the rightmost resistor is chosen equal to $2R$ rather than R . Determine the overall Norton equivalent in a form similar to that in Fig. 10.13(d).
- 10.8. We wish to design the segmented DAC of Fig. 10.14 for a resolution of 12 bits. Determine the matching required of the current sources if the thermometric array incorporates 127 units.
- 10.9. The single-ended DAC of Fig. 10.16(a) is designed for a resolution of 10 bits with $R_L = 50\ \Omega$. Determine the minimum tolerable r_O if the second harmonic must be 60 dB below the first.
- 10.10. We can express the DAC characteristic of Fig. 10.17 as the sum of the straight line and the INL profile given by

$$\text{INL}_j = \frac{I_0 R_L^2}{r_O} j(m-j). \quad (10.81)$$

Replace j with $(m/2) \cos \omega_1 t + m/2$ and obtain the second harmonic.

- 10.11. Suppose the output resistance of a current source scales linearly with its current; e.g., doubling the current halves the resistance. We can design the DAC of Fig. 10.18(a) with $R_L = 50\ \Omega$ and a full-scale current of I_{FS} or with $R_L = 25\ \Omega$ and a full-scale current of $2I_{FS}$. Does either design provide a lower INL_{max} ?
- 10.12. A differential 10-bit IDAC must deliver a single-ended swing of $0.5\ V_{pp}$ to $50\text{-}\Omega$ loads. (a) Compute the necessary unit cell output resistance for $\text{INL}_{max} \leq 0.5\ \text{LSB}$. (b) Considering only random mismatches, determine the channel area, WL , of M_1 in Fig. 10.21 if $A_\beta = 0.05\ \mu\text{m}$, $A_{VTH} = 2.5\ \text{mV} \cdot \mu\text{m}$, and $V_{GS} - V_{TH} = 100\ \text{mV}$.
- 10.13. In the matrix architecture of Fig. 10.22(b), determine the number of current sources that are turned on in each row if the digital input is equal to 29, 39, or 49.
- 10.14. Suppose the current sources in Fig. 10.22(b) suffer from linear gradients in both x and y directions. In a given row, the unit current increases by ΔI as we travel to the right. Also, in each column, the unit current increases by ΔI as we travel to the bottom. (a) Assign a value of I_0 to the unit on the top right and find expressions for the currents in the rows and columns. (b) Sketch the input-output characteristic and compute the gain error and the maximum INL.
- 10.15. Consider the unit placement shown in Fig. 10.26(a), which we denote by 13578642. Suppose we instead arrange the current sources as 75312468. Sketch the input-output characteristic and the INL.
- 10.16. Using Eqs. (10.53) and (10.54), compute the INL profile due to the voltage drop along the ground line.
- 10.17. Suppose the ground line in Fig. 10.31(a) is tapered such that the unit resistance decreases linearly, from r_u to $r_u - \Delta r$ to $r_u - 2\Delta r$, etc. Estimate the voltage drop across the entire ground (the right side with respect to the left side).
- 10.18. A student mistakenly tapers the ground in Fig. 10.31(a) such that the leftmost resistance is r_u , the next one is $r_u/2$, etc. Find the voltage drop across the entire ground line.
- 10.19. The IDAC of Fig. 10.37(a) is designed to deliver a maximum of $10\ \text{mA}$ to $R_L = 50\ \Omega$. Assuming a resolution of 8 bits and $C_u = 1\ \text{fF}$, determine the maximum sinusoidal output frequency that the DAC can provide while maintaining an HD_2 of $-50\ \text{dB}$.
- 10.20. A student allows a large voltage swing at the output of the unit cell shown in Fig. 10.35(a). Thus, M_1 enters the deep-triode region. Explain how the HD_2 expression changes in this case.

Capacitor DACs

Some ADC architectures employ capacitor DACs (CDACs) for their low power consumption and rail-to-rail output swings. In this chapter, we analyze and design CDACs in preparation for using them within ADCs in the subsequent chapters. We first deal with basic CDAC topologies, and then with binary-weighted and thermometric architectures. We also analyze CDACs' speed and precision limitations and study the problem of reference bounce. The reader is encouraged to review Chapter 2 before beginning this chapter. We also draw upon many concepts developed for RDACs and IDACs.

11.1 Basic Topologies

Recall that resistor-ladder DACs subdivide their reference to uniformly-spaced voltages and deliver one to the output. We can ask whether reference division can be performed by capacitors. Let us begin with Fig. 11.1(a), where $V_{out} = 0$ at $t = 0$. We then turn S_{1a} off and turn S_{1b} on, recognizing that the left plate of C_1 jumps

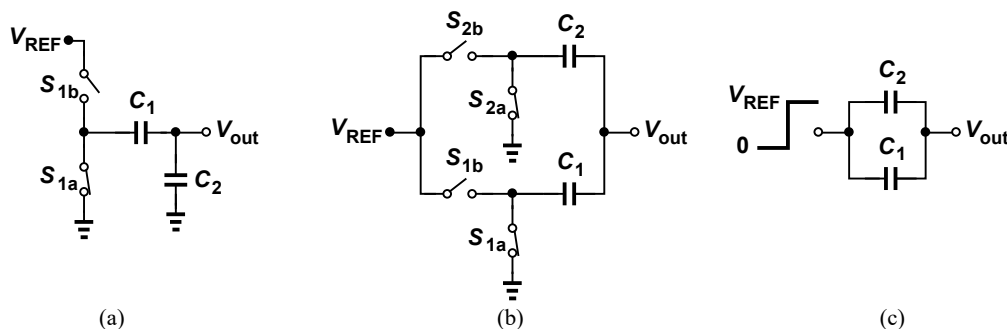


Figure 11.1 (a) A capacitive divider, (b) simple DAC, and (c) equivalent circuit when both left plates switch to V_{REF} .

from zero to V_{REF} . As explained in Chapter 2, the capacitive divider consisting of C_1 and C_2 generates an output *change* given by

$$\Delta V_{out} = \frac{C_1}{C_1 + C_2} V_{REF}. \quad (11.1)$$

For example, if $C_1 = C_2$, the circuit can deliver two output levels equal to zero and $V_{REF}/2$, acting as a 1-bit DAC.

Now, we consider the arrangement in Fig. 11.1(b). If S_{2a} is always on, the circuit is equivalent to that in Fig. 11.1(a). But suppose we turn off S_{1a} and S_{2a} and turn on S_{1b} and S_{2b} . As depicted in Fig. 11.1(c), the left plates of both capacitors travel from zero to V_{REF} . How much is the change in V_{out} ? Since the output

node sees no capacitance to ground, there is no voltage division and hence

$$\Delta V_{out} = V_{REF}. \quad (11.2)$$

The circuit can then produce three levels equal to zero, $V_{REF}/2$, and V_{REF} if $C_1 = C_2$.

Let us next study the topology shown in Fig. 11.2(a), where the switches are controlled by a thermometer code ranging from 0 to 3. If only the left plate of C_1 swings from zero to V_{REF} , then the circuit reduces to that in Fig. 11.2(b), yielding an output change equal to $V_{REF}/4$. Similarly, if only the left plates of C_1 and C_2 are switched, the output jumps by $2V_{REF}/4$, etc. The structure therefore provides output levels equal to $kV_{REF}/4$, where $k = 0, \dots, 3$, acting as a DAC with an output LSB of $V_{REF}/4$. Figure 11.2(c) plots the input-output characteristic as the digital input varies from 0 to 3.

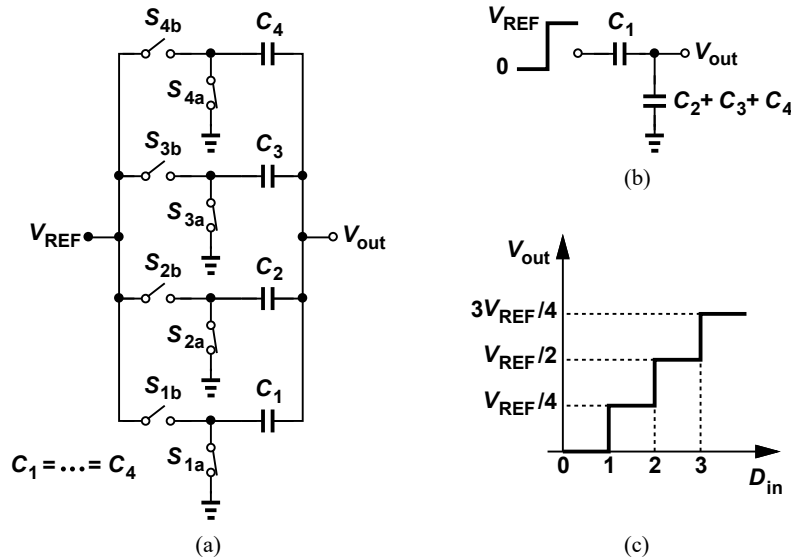


Figure 11.2 (a) A CDAC using four units, (b) equivalent circuit, and (c) input-output characteristic.

Example 11.1

What is the resolution of the DAC shown in Fig. 11.2(a)?

Solution

Since the circuit provides five output levels, the resolution is given by $\log_2 5 = 2.3$ bits. If we wish to deliver only four levels, we configure the branches as depicted in Fig. 11.3, where C_1 is not switched. The output values are now given by 0, $V_{REF}/4$, $2V_{REF}/4$, and $3V_{REF}/4$. What happens if C_1 is removed?

The CDACs of Fig. 11.2(a) and Fig. 11.3 offer several desirable properties: (1) their output can swing from zero to V_{DD} ; (2) they do not suffer from the nonlinearity observed in IDACs due to the output impedance of the current sources; and (3) they consume much less power than do RDACs and IDACs.

The drawbacks of CDACs are as follows: they cannot drive resistive loads (why?) and they incur gain error in the presence of capacitive loads. For example, if, in Fig. 11.3, a load capacitance of C_L appears in parallel with C_1 , we write from Fig. 11.4

$$V_{out} = \frac{kC_0}{4C_0 + C_L} V_{REF} \quad (11.3)$$

$$\approx \frac{k}{4} \left(1 - \frac{C_L}{4C_0} \right) V_{REF}. \quad (11.4)$$

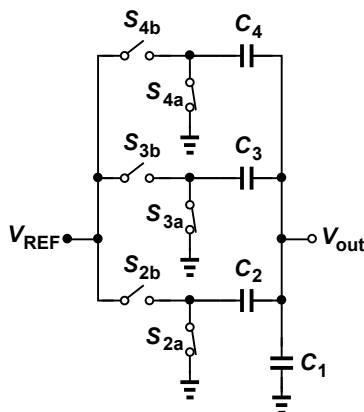


Figure 11.3 CDAC with one grounded capacitor.

Here, we have assumed that $C_L \ll 4C_0$. It follows that the gain error is equal to $C_L/(4C_0)$, a critical issue in some ADC architectures that employ this type of DAC.

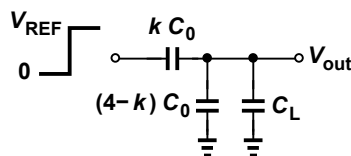


Figure 11.4 Equivalent CDAC circuit for a digital input equal to k .

With typical values of V_{REF} , e.g., 0.6 V, we expect that the “ a ” and “ b ” switches in Fig. 11.2 are realized by NMOS and PMOS devices, respectively. We also surmise that the a and b groups should have equal resistances for optimum settling, a point revisited in Section 11.3.

In the analysis of the foregoing CDACs, we assumed that V_{out} begins from zero. How do we guarantee this? We must attach a reset switch from the output node to ground (or a common-mode level appropriate for the next stage) (Fig. 11.5). The reset action must occur in every conversion cycle to counteract the leakage of S_R ,¹ leading to the “return-to-zero” output waveform shown in Fig. 11.5. This behavior is relatively benign if the DAC serves in an ADC environment. Nonetheless, the parasitic capacitances of S_R introduce gain error and nonlinearity (Chapter 15).

11.2 Binary-Weighted and Thermometric CDACs

The disadvantages outlined for binary IDACs in Section 10.3 also apply to binary CDACs. In particular, to relax the matching requirements for a given DNL, we prefer thermometric CDACs for resolutions higher than roughly 4 bits.

The CDAC shown in Fig. 11.2 and repeated in Fig. 11.6(a) is in fact driven by a thermometer code. The binary counterpart is depicted in Fig. 11.6(b). Here, S_{1a} and S_{1b} control the LSB capacitor, and S_{2a} and S_{2b} the MSB capacitor.

How do we select the switch dimensions in Fig. 11.6(b)? We expect that larger capacitors require smaller series resistances so that the time constants of all of the branches are approximately equal (Section 11.3).

¹ A MOS device exhibits leakage through its junctions and through its channel.

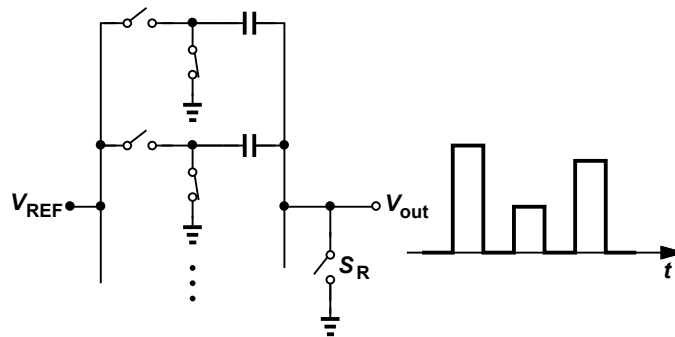


Figure 11.5 DAC output behavior with a reset switch.

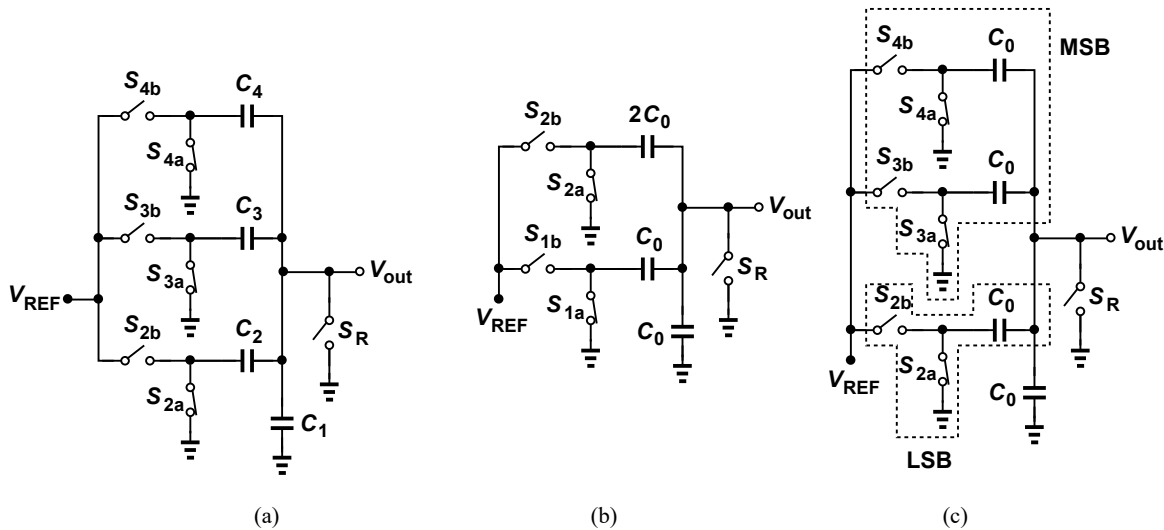


Figure 11.6 (a) A 2-bit CDAC, (b) binary-weighted counterpart, and (c) proper unit design for maximum speed.

Thus, as in IDACs, we construct this CDAC as an array of switching units that are grouped to realize binary weighting [Fig. 11.6(c)].

Example 11.2

Explain how the topology of Fig. 11.6(c) ensures equal time constants for the MSB and LSB paths.

Solution

Suppose the MSB capacitors are switched to V_{REF} , as shown in Fig. 11.7. We observe that the two unit branches act as one containing a resistance of $R_{on}/2$ in series with a capacitance of $2C_0$ (why?). The time constant is therefore equal to that of the LSB branch.

Both binary-weighted and thermometric DACs can readily accommodate differential operation. Shown in Fig. 11.8(a) is a 2-bit binary-weighted example. Each side requires an extra unit tied to ground. Note that, in the reset mode, the “a” switches on the left and the “b” switches on the right are on, and the output nodes are tied to a common-mode level, V_{CM} . For the DAC to evaluate the digital input, D_{in} , we turn off S_{R1} and S_{R2} and control the state of the remaining switches. If $D_{in} = 0$, the switches do not change and $V_{out} = 0$. If $D_{in} = 1$, the left plate of C_1 swings from zero to V_{REF} while the right plate of C_3 goes from V_{REF} to zero.

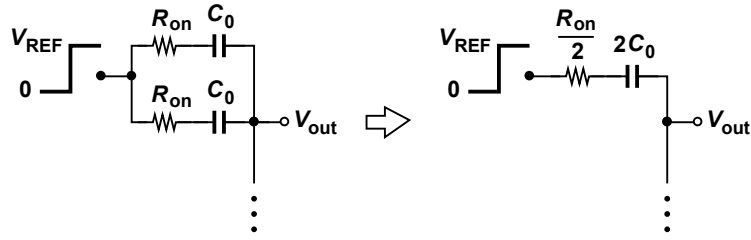


Figure 11.7 Equivalent CDAC circuit for settling time calculation.

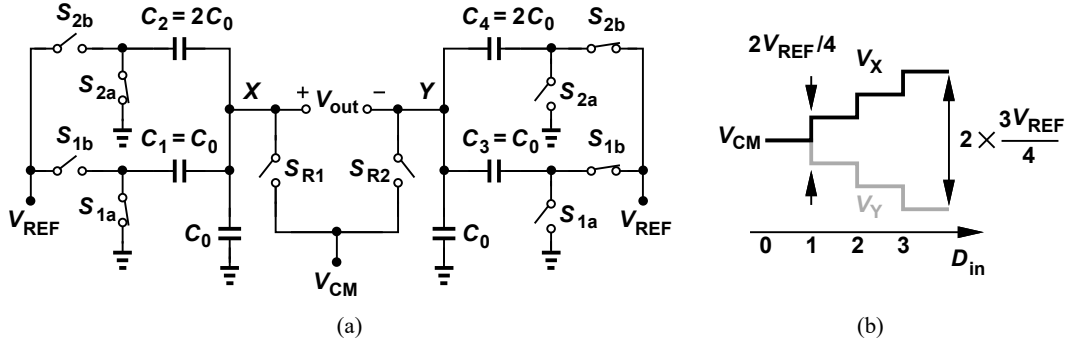


Figure 11.8 (a) Differential CDAC, and (b) its input-output characteristic.

Thus, as illustrated in Fig. 11.8(b), V_X jumps up by $V_{REF}/4$ and V_Y jumps down by the same amount. We conclude that the output differential LSB size is $2 \times V_{REF}/4$, an expected result because, in comparison to Fig. 11.2(c), the output full scale has risen to $2(3V_{REF}/4)$.

11.3 Speed Limitations

Suppose the digital input of a CDAC jumps from zero to k at $t = 0$. How long does the analog output take to settle to the desirable precision? As depicted in Fig. 11.9(a), k units switch to V_{REF} while $m - k$ units remain connected to zero. Here, m denotes the total number of unit branches, e.g., $m = 256$ for an 8-bit design. We should remark that the dummy unit also contains a switch that is permanently on. Assuming that the resistance of each switch is R_{on} , we draw the equivalent circuit in Fig. 11.9(b), noting that the parallel branches collapse into one branch [Fig. 11.9(c)]. Let us examine the output voltage at $t = 0^+$ and at $t = \infty$. With the input voltage jump, the capacitors act as a short circuit, reducing the circuit to a resistive divider consisting of R_{on}/k and $R_{on}/(m - k)$. That is,

$$V_{out}(0^+) = \frac{\frac{R_{on}}{m-k}}{\frac{R_{on}}{k} + \frac{R_{on}}{m-k}} V_{REF} \quad (11.5)$$

$$= \frac{k}{m} V_{REF}. \quad (11.6)$$

This is the desired value. At $t = \infty$, the currents drop to zero, the resistors sustain no voltage, and the circuit resembles a capacitive divider. Thus,

$$V_{out}(\infty) = \frac{kC_0}{kC_0 + (m-k)C_0} V_{REF} = \frac{k}{m} V_{REF}. \quad (11.7)$$

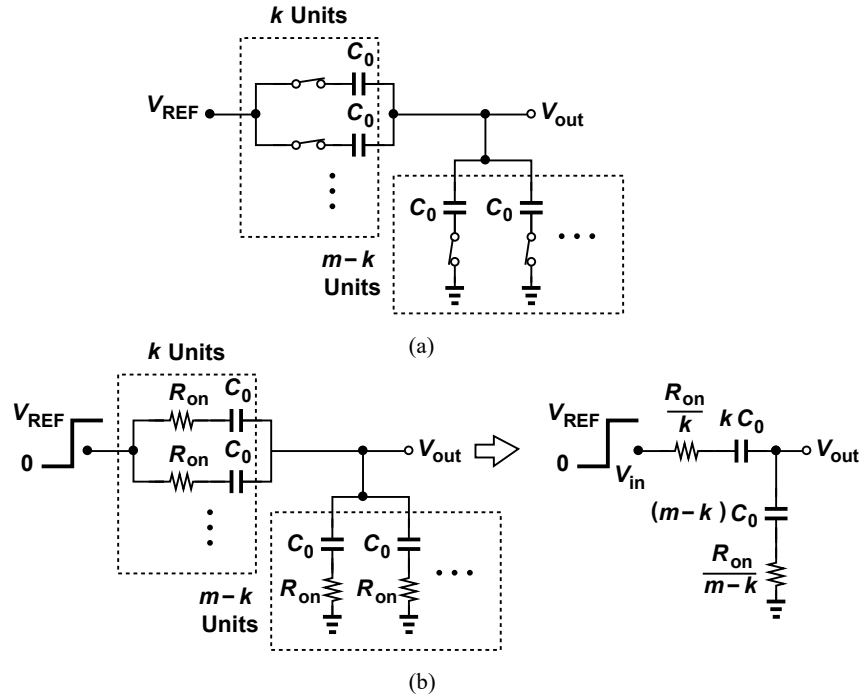


Figure 11.9 (a) CDAC with a digital input equal to k , and (b) its equivalent circuit.

Interestingly, the output settles *immediately*. How is that possible? In Problem 11.8, we show that the transfer function from V_{in} to V_{out} in Fig. 11.9(b) contains a pole and a zero that cancel each other, giving $V_{out}/V_{in} = k/m$. Unique to CDACs, this infinite speed unfortunately does not materialize in practice due to the finite output impedance of the reference generator.

Example 11.3

Compute the current drawn from the reference in Fig. 11.9(b).

Solution

At $t = 0^+$, the circuit reduces to only two resistors, namely, R_{on}/k and $R_{on}/(m-k)$, drawing a current equal to

$$I_{REF}(0^+) = \frac{V_{REF}}{\frac{R_{on}}{k} + \frac{R_{on}}{m-k}} \quad (11.8)$$

$$= \frac{k(m-k)}{mR_{on}} V_{REF}. \quad (11.9)$$

The highest current occurs for $k = m/2$ and is given by $mV_{REF}/(4R_{on})$. The reference current falls with a time constant of

$$\tau = \left(\frac{R_{on}}{k} + \frac{R_{on}}{m-k} \right) \frac{(kC_0)[(m-k)C_0]}{kC_0 + (m-k)C_0} \quad (11.10)$$

$$= R_{on}C_0. \quad (11.11)$$

We recognize that the maximum initial current, $mV_{REF}/(4R_{on})$, can be large, causing significant reference bounce (Section 11.6) and leading to long settling.

Given that the intrinsic speed of CDACs does not depend on the on-resistance of the switches, and given that the reference current does, the reader may wonder whether we can select a *large* value for R_{on} so as to lessen the reference bounce. Such a remedy faces two issues. First, from the time the switches turn on until the final settling, their on-resistance varies because their V_{GS} is not constant. Thus, at least for some fraction of the settling period, the assumption of equal resistances is not valid, making settling a function of R_{on} . Second, as explained in Chapters 14 and 17, CDACs typically also serve as samplers in ADCs, requiring sufficient acquisition bandwidth and hence a small time constant.

11.4 Effect of Random Mismatches

Consider the thermometric CDAC shown in Fig. 11.10(a). We expect that capacitor mismatches introduce

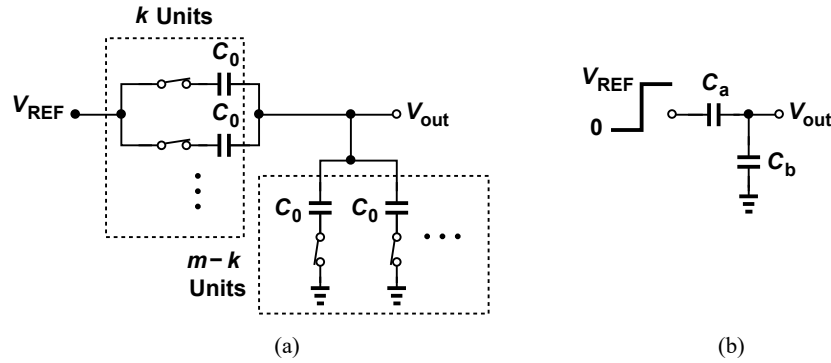


Figure 11.10 (a) CDAC for mismatch study, and (b) its equivalent circuit.

DNL and INL. But we also note that $V_{out} = 0$ if no capacitors switch to V_{REF} , and $V_{out} = V_{REF}$ if all do. That is, regardless of mismatches, the two end points of the characteristic remain equal to their ideal values. We then surmise that the highest INL is more likely to occur around the mid-scale, in a manner similar to that of RDACs (Section 9.5).

Let us repeat the RDAC INL analysis of Example 9.6 for CDACs. Suppose, of the m units in Fig. 11.10(a), half switch to V_{REF} . Each capacitor has a mean value equal to C_0 and a standard deviation (SD) equal to ΔC . For $m/2$ capacitors, the mean and SD are given by $mC_0/2$ and $\sqrt{m/2}\Delta C$, respectively. From Fig. 11.10(b), we have

$$V_{out} = \frac{C_a}{C_a + C_b} V_{REF}, \quad (11.12)$$

where $V_{out}(0^-)$ is assumed to be zero, and C_a and C_b have a mean value of $(m/2)C_0$ and equal SDs. Denoting their errors by ΔC_a and ΔC_b , respectively, we write

$$V_{out} = \frac{\frac{m}{2}C_0 + \Delta C_a}{\frac{m}{2}C_0 + \Delta C_a + \frac{m}{2}C_0 + \Delta C_b} V_{REF} \quad (11.13)$$

$$= \frac{\frac{m}{2}C_0}{mC_0} \cdot \frac{1 + \frac{2\Delta C_a}{mC_0}}{1 + \frac{\Delta C_a + \Delta C_b}{mC_0}} V_{REF}. \quad (11.14)$$

If the errors are small,

$$V_{out} \approx \left(1 + \frac{2\Delta C_a}{mC_0}\right) \left(1 - \frac{\Delta C_a + \Delta C_b}{mC_0}\right) \frac{V_{REF}}{2} \quad (11.15)$$

$$\approx \frac{V_{REF}}{2} + \frac{\Delta C_a - \Delta C_b}{mC_0} \cdot \frac{V_{REF}}{2}. \quad (11.16)$$

We can say V_{out} is a random variable with a mean value of $V_{REF}/2$ and an “error” of $[(\Delta C_a - \Delta C_b)/(mC_0)](V_{REF}/2)$. (Our assumption is that, if the capacitors follow a Gaussian distribution, so does V_{out} .) This error represents the INL (why?). We now treat the errors as standard deviations. Since the mismatches in C_a and C_b are uncorrelated, we must add the squares of ΔC_a and ΔC_b to find the SD of the error:

$$INL \approx \frac{\sqrt{\Delta C_a^2 + \Delta C_b^2}}{mC_0} \frac{V_{REF}}{2} \quad (11.17)$$

$$\approx \frac{\sqrt{m}\Delta C}{mC_0} \frac{V_{REF}}{2} \quad (11.18)$$

$$\approx \frac{1}{\sqrt{4m}} \frac{\Delta C}{C_0} V_{REF}. \quad (11.19)$$

Note the similarity between this result and that for RDACs [Eq. (9.31)]. For the INL to remain below 1 LSB = V_{REF}/m ,

$$\frac{\Delta C}{C_0} < \frac{2}{\sqrt{m}}. \quad (11.20)$$

11.5 Capacitor Structures

The implementation of capacitors in CMOS technology has been under research for decades—not only for CDACs but also for analog filters, op amp compensation, oscillators, supply regulation, etc. In DAC design, we seek capacitor structures that exhibit proper matching and yet do not occupy a large area.

In a given CMOS process, MOS transistors provide the densest capacitors by virtue of their high channel capacitance, e.g., about 20 fF/ μm^2 in 28-nm technology. Can we utilize such devices to realize CDACs? Recall from our study in the previous sections that the voltage across a CDAC’s capacitor changes from zero to a finite value, and not all of the units sustain the same voltage. Thus, if the capacitance *varies* with voltage, the DAC suffers from nonlinearity. As shown in Fig. 11.11, this is the case for MOS devices, making them ill-suited to CDAC design.

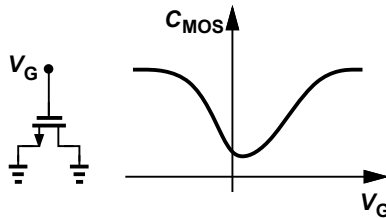


Figure 11.11 A MOS cap and its C-V characteristic.

We must therefore resort to capacitances between metal layers. Depicted in Fig. 11.12(a) is a parallel-plate example consisting of metal-9 (M9) and metal-8 (M8) layers. This structure exhibits very high linearity, serving resolutions up to 16 or even 18 bits. Unfortunately, its capacitance density is very low, in the vicinity of 50 aF/ μm^2 .

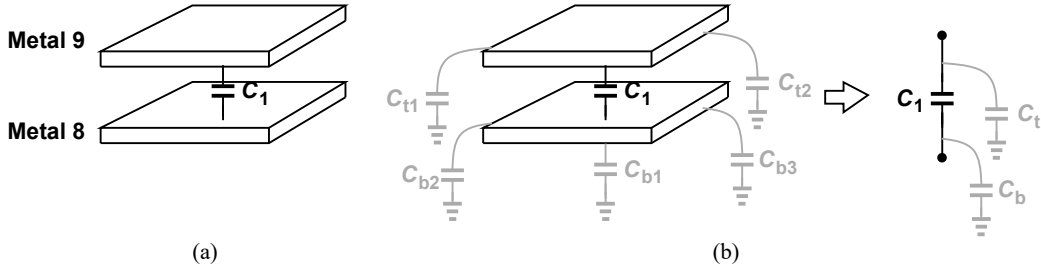


Figure 11.12 (a) Parallel-plate capacitor, and (b) its parasitics.

Example 11.4

An 8-bit differential CDAC employs a unit capacitor equal to 10 fF. Estimate the DAC's area if the structure in Fig. 11.12(a) is used.

Solution

Each unit occupies an area of $1000 \mu\text{m}^2$, e.g., $31.6 \mu\text{m} \times 31.6 \mu\text{m}$. The differential DAC incorporates 512 such units, which can be arranged in a 16×32 array. The DAC thus occupies about $500 \mu\text{m} \times 1000 \mu\text{m}$. In reality, the capacitors must have a minimum spacing to satisfy the design rules, further increasing the area.

Capacitor Parasitics The geometry of Fig. 11.12(a) suffers from parasitic capacitances. As illustrated in Fig. 11.12(b), the fringe electric field lines emanating from the top plate and ending on the substrate give rise to a “top-plate parasitic,” C_t , which appears on all four sides of the plate. In addition, the bottom plate and the substrate act as a parallel-plate structure, introducing a “bottom-plate parasitic,” C_b , which also includes fringe components. We normalize C_t and C_b to C_1 . For an M9-M8 topology, $C_t \approx 1\%$ and $C_b \approx 10\%$. Whether such values are acceptable or not depends on the effect of these parasitics on the performance of CDACs.

Let us first examine how the top-plate parasitic alters the DAC's behavior. Redrawing the circuit of Fig. 11.9(a) in Fig. 11.13(a) and assuming the top plates of the capacitors are tied to the output node, we observe

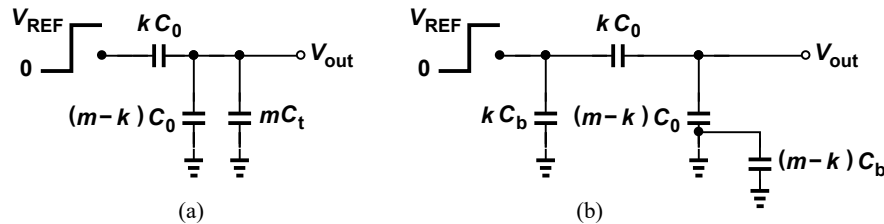


Figure 11.13 A CDAC including (a) top-plate, and (b) bottom-plate parasitic capacitance.

a total top-plate capacitance of mC_t to ground. As expected from Eq. (11.1), the DAC incurs a gain error. We have

$$V_{out} = \frac{kC_0}{mC_0 + mC_t} V_{REF} \quad (11.21)$$

$$\approx \frac{k}{m} \left(1 - \frac{C_t}{C_0} \right) V_{REF}. \quad (11.22)$$

This result shows why we prefer to connect the top plates (with their smaller parasitics) to the output. In some ADC architectures, the DAC gain error becomes critical (Chapter 18).

We now study the effect of bottom-plate parasitics. Figure 11.13(b) shows how these capacitances, kC_b and $(m - k)C_b$, appear in the equivalent circuit, the latter playing no role. Is V_{out} affected by kC_b ? Since the left-plate voltage of kC_0 still jumps by an amount equal to V_{REF} and since capacitive dividers transfer only voltage changes (Chapter 2), V_{out} is still equal to

$$V_{out} = \frac{kC_0}{mC_0} V_{REF}. \quad (11.23)$$

That is, kC_b only raises the transient current drawn from V_{REF} but does not affect the DAC's precision. In CDAC design, therefore, the bottom-plate parasitic only degrades the speed to some extent (and raises the power consumption).

Multilayer Capacitors The low density of the two-layer capacitor shown in Fig. 11.12(a) motivates us to seek better structures. With the availability of nine or ten metal layers in most CMOS processes, we surmise that an arrangement such as that in Fig. 11.14(a) provides a more compact solution. Here, the capacitances between different plates—made of metal 6 to metal 9—appear in parallel, roughly tripling the capacitance.

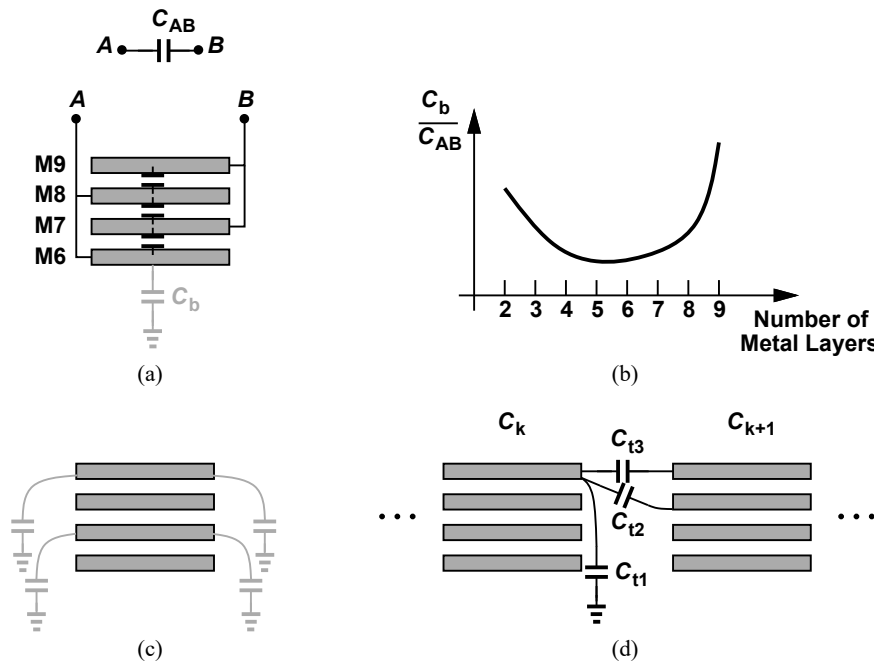


Figure 11.14 (a) Multilayer capacitor, (b) behavior of normalized bottom-plate parasitic, (c) effect of fringe capacitances, and (d) fringe components between unit capacitors.

In this case, the term “top plate” refers to that terminal of the capacitor which exhibits less capacitance to the ground, namely, terminal B and hence the M7 and M9 plates. The other two form the “bottom plate” and are tied to terminal A .

Can we extend the idea to a stack of metal 1 to metal 9 layers? Yes, we can, but we must mind the bottom-plate parasitic, C_b . We observe that in Fig. 11.14(a), C_b stems from M6, which is closer to the substrate than M8 in Fig. 11.12(a). Thus, as we stack more layers, C_b increases. In practice, the normalized parasitic, C_b/C_{tot} , varies as depicted in Fig. 11.14(b),² reaching a minimum of about 5% for the top four or five metal layers.

²The metal layers start from M9 and increase downward.

As seen previously, the bottom-plate capacitance does not affect the precision, but the top-plate one does. In the geometry of Fig. 11.14(a), the M9 and M7 layers form the top plates and incur fringe capacitances to the substrate [Fig. 11.14(c)]. The situation becomes more complex if an array of unit capacitors is formed [Fig. 11.14(d)]. Each plate now suffers from parasitics to the plates of other units.

Example 11.5

Explain how C_{t1} , C_{t2} , and C_{t3} in Fig. 11.14(d) affect the DAC's behavior.

Solution

Appearing between the output node and ground, C_{t1} introduces gain error. Capacitance C_{t2} , on the other hand, goes from the top plate of C_k to the *second* plate of C_{k+1} (Fig. 11.15). That is, C_{t2} simply adds to C_{k+1} , causing mismatch. Of course, another parasitic goes from the top plate of C_{k+1} to the second plate of C_k , canceling this mismatch. Capacitance C_{t3} is benign (why?).

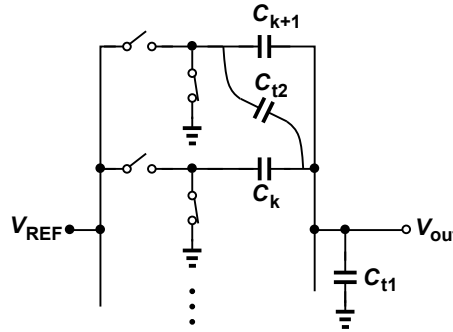


Figure 11.15 Equivalent circuit including fringe capacitance.

How are M9 and M7 (and M8 and M6) connected to each other in Fig. 11.14(a)? This is accomplished by extending each plate sideways and placing a column of “vias” (contacts) on it to reach the higher layer. As shown in Fig. 11.16, one dimension of the capacitor increases by $2S_1 + 2S_2$, where S_1 denotes the minimum allowable horizontal spacing between metal lines, and S_2 the minimum allowable width of the vias.

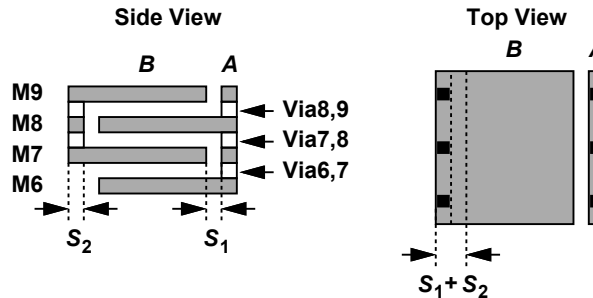


Figure 11.16 Actual structure of a multilayer capacitor.

The undesirable coupling effects seen in Example 11.5 can be suppressed by enclosing each unit capacitor in a Gaussian surface, i.e., in a metal “cage.” Illustrated in Fig. 11.17(a), the idea is to ensure that all electric field lines emanating from the “top” plates terminate only on the “bottom” plates and not on the substrate or on another unit. Here, a wall made of vias (contacts) surrounds the core and connects to the topmost and bottommost plates and the M7 plate. We then consider the M9/M7/M5 node and their associated vias as the “bottom” plate. Note how the structures in Figs. 11.16 and 11.17(a) differ: the “top” plate in the latter consists of M8 and M6.

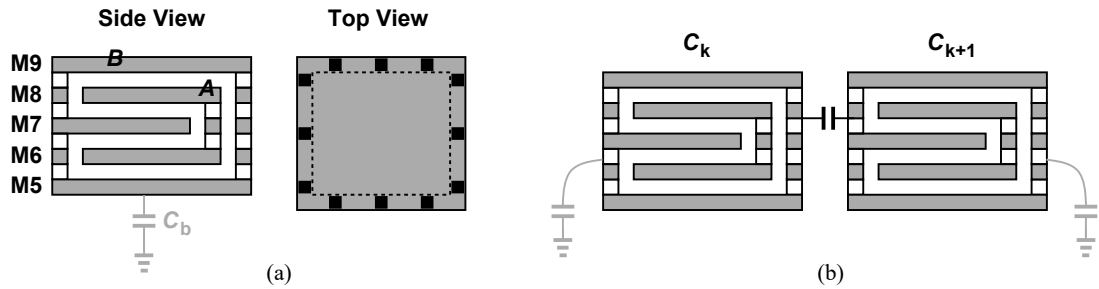


Figure 11.17 (a) Multilayer capacitor in a cage, and (b) effect of fringe capacitances.

It is instructive to examine the fringe capacitances in Fig. 11.17(a). We recognize from Fig. 11.17(b) that the M8 and M6 plates experience fringe to the walls. This simply adds to the total capacitance, i.e., the top-plate parasitic to ground is very small. The fringe capacitance outside the walls appears to the walls of adjacent units and to ground [Fig. 11.17(b)], only raising the “bottom-plate” capacitance to ground (why?).

Lateral Capacitors It is possible to achieve higher capacitance densities by exploiting the fringe fields between metal lines. With the scaling of CMOS technology to smaller geometries, the minimum allowable spacing between the lines has also shrunk, yielding greater lateral (fringe) capacitances. Consider the two metal-9 lines shown in Fig. 11.18(a), noting the capacitance, C_{AB} , between them. Even with a minimum

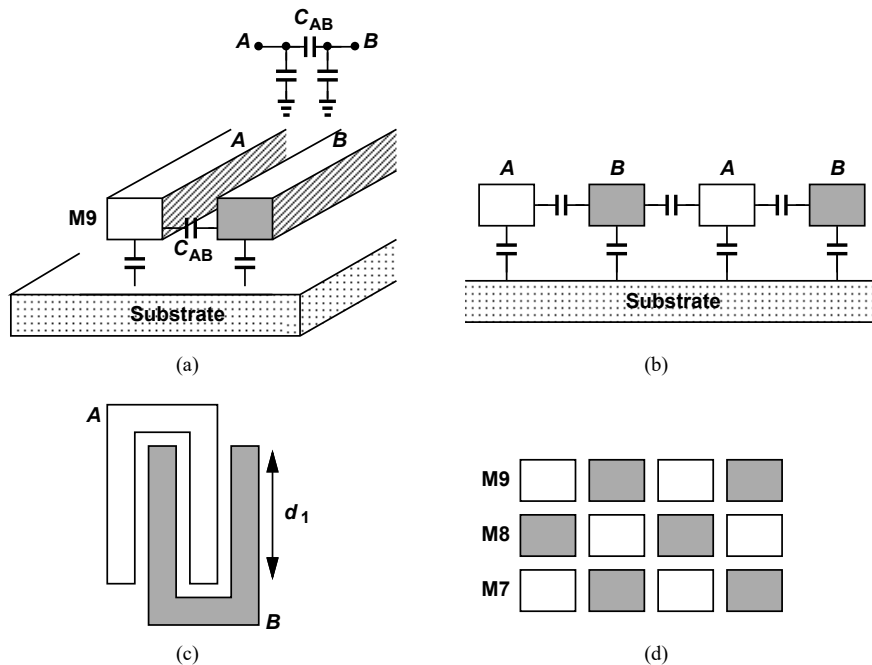


Figure 11.18 (a) Simple lateral capacitor, (b) structure having four fingers (side view), (c) top view, and (d) a multilayer structure (side view).

width, each line also displays capacitance to the substrate. Now, suppose we repeat this structure laterally and connect the lines alternately [Fig. 11.18(b)].³ Thus, C_{AB} increases proportionally. The top view in Fig. 11.18(c) shows the actual connections.

³The shading of the metal lines is to distinguish between the A and the B plates.

Next, we repeat the structure in multiple metal layers [Fig. 11.18(d)], utilizing the “vertical” capacitances as well. Called a “fringe” or “lateral” capacitor, this arrangement provides a higher density than the multi-layer structure in Fig. 11.16.

The fringe capacitor of Fig. 11.18(d) entails two issues. First, unlike the geometry of Fig. 11.17(a), it exhibits nearby *equal* “top-plate” and “bottom-plate” parasitic capacitances because it is relatively symmetric with respect to its two terminals. Depicted in Fig. 11.19(a), these components appear from both A and B to ground.⁴ To minimize the top-plate parasitic (and the DAC gain error), we can enclose the entire arrangement in a Gaussian surface, as in Fig. 11.17(a). Second, since the topmost metal layer, e.g., M9, is typically thicker, it requires a greater width and a greater lateral spacing than do the lower layers. Thus, the fringe capacitor of Fig. 11.18(d) is modified to, for example, that in Fig. 11.19(b).

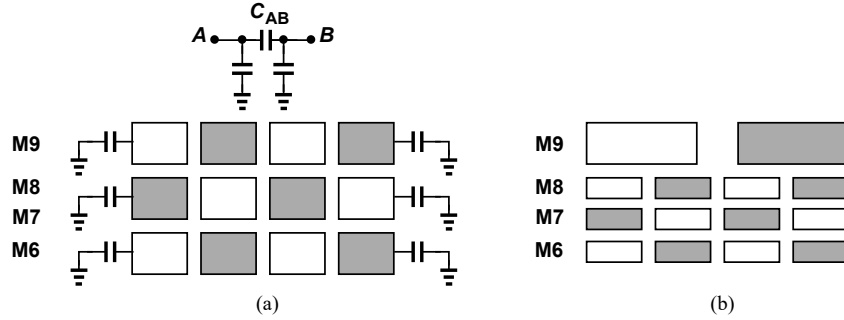


Figure 11.19 (a) Effect of parasitics on a lateral structure, and (b) a more practical realization.

11.6 Problem of Reference Bounce

In Section 9.6, we studied the reference bounce in the context of RDACs. We expect that the issue arises in CDACs, too, and perhaps more severely.

From Fig. 11.9(b), we can construct the simple equivalent circuit shown in Fig. 11.20(a) for $t = 0^+$, disregarding C_1 for now. Here, the reference exhibits an output resistance, R_{REF} . We observe that the large

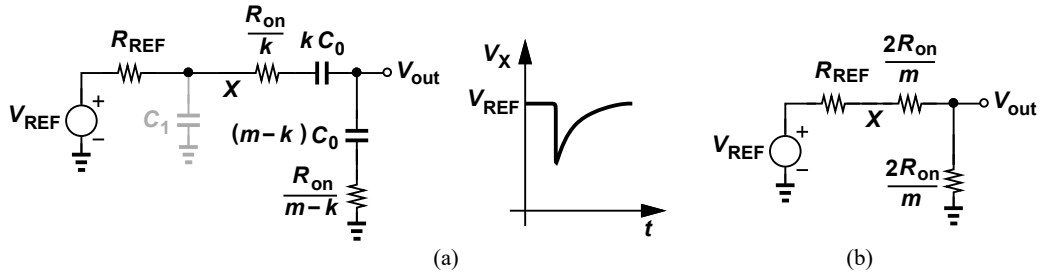


Figure 11.20 (a) CDAC driven by a finite reference resistance while k units switch to X , and (b) equivalent circuit at $t = 0^+$.

current drawn by the DAC causes V_X to drop considerably. Then, V_X recovers with a time constant *greater* than that in Eq. (11.11) because the series resistance in (11.8) now includes R_{REF} . In particular, if half of the unit capacitors switch to the reference, the circuit reduces to that in Fig. 11.20(b) at $t = 0^+$, yielding

$$V_X = \frac{\frac{4R_{on}}{m}}{\frac{4R_{on}}{m} + R_{REF}} V_{REF}, \quad (11.24)$$

⁴Note that half of the bottommost fingers are attached to A and the other half to B ; so the parallel-plate parasitics are also equal.

and a time constant given by

$$\tau = \left(\frac{4R_{on}}{m} + R_{REF} \right) \frac{m}{4} C_0. \quad (11.25)$$

As explained in Example 9.8, it is possible to reduce the reference bounce by tying a sufficiently large capacitor, C_1 , from node X in Fig. 11.20(a) to ground. Recall from that example that we must have $C_1 \approx (2m)C_L$, where C_L is the series combination of the capacitors in Fig. 11.20(a) and is equal to $(m/4)C_0$ for $k = m/2$. That is, C_1 must be as large as $(m^2/2)C_0$, occupying a greater area than the DAC itself!

Example 11.6

In Fig. 10.6, we observed that a differential IDAC draws much lower transient currents from the supply than its single-ended counterpart. Does this hold for CDACs and their reference voltage?

Solution

Unfortunately, it does not. Returning to Fig. 11.8(a), suppose D_{in} jumps from 00 to 10. Then, C_2 switches to V_{REF} while C_4 switches to ground. The circuit therefore does not provide any cancellation of the reference current.

If $C_1 \approx (m^2/2)C_0$ is impractically large, we resort to a reference buffer (Fig. 11.21). We recognized in Section 9.6, however, that the buffer's output impedance climbs with frequency, failing to hold V_X constant

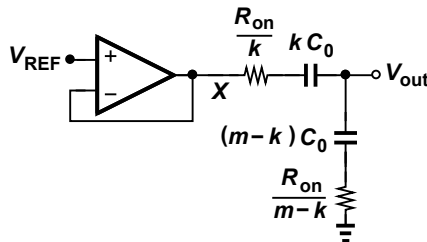


Figure 11.21 CDAC driven by a reference buffer.

when the DAC switches. Compared to the RDAC buffer in Fig. 9.14(a), this case proves more severe because the CDAC presents a far greater capacitance to the buffer. As explained in Section 9.6, we typically add some capacitance from X to ground so as to lower the buffer's output impedance at high frequencies, but this capacitance may need to be very large.

Example 11.7

Figure 11.22(a) shows a simple unity-gain buffer implementation for driving CDACs. Capacitor C_A is typically necessary for frequency compensation. Determine (a) the small-signal output resistance of the buffer, and (b) the minimum supply voltage for the buffer. Neglect all other capacitances.

Solution

The closed-loop output impedance of the buffer is given by $1/g_{m5}$ divided by 1 plus the loop gain, where channel-length modulation and body effect of M_5 are neglected. These assumptions lead to a unity gain for the source follower and hence a loop gain⁵ equal to $g_{m1,2}(r_{O2}||r_{O4})/[(r_{O2}||r_{O4})C_A s + 1]$ (why?). The

⁵When the loop gain is a function of the frequency, it is called the “loop transmission.”

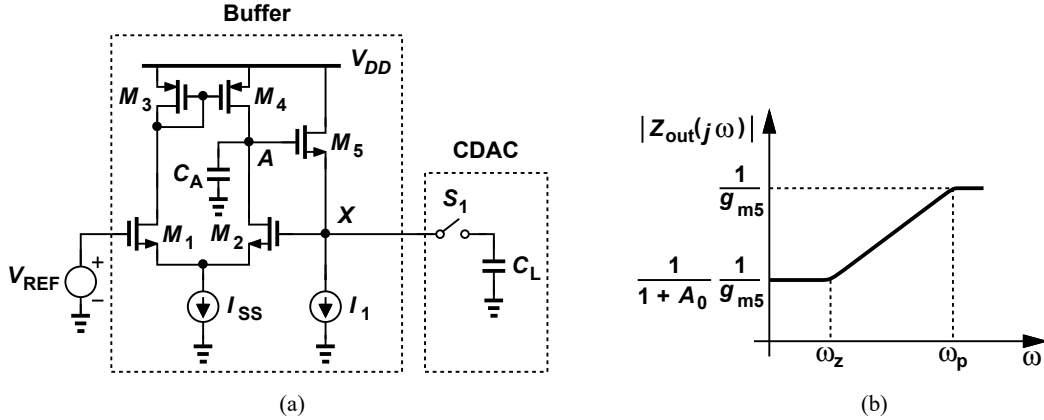


Figure 11.22 (a) Five-transistor amplifier with a source follower acting as a reference buffer, and (b) behavior of its output impedance.

closed-loop output impedance thus emerges as

$$Z_{out}(s) = \frac{1}{g_{m5}} \frac{1}{1 + \frac{g_{m1,2}(r_{O2}||r_{O4})}{(r_{O2}||r_{O4})C_A s + 1}} \quad (11.26)$$

$$= \frac{1}{g_{m5}} \frac{(r_{O2}||r_{O4})C_A s + 1}{(r_{O2}||r_{O4})C_A s + 1 + g_{m1,2}(r_{O2}||r_{O4})}. \quad (11.27)$$

We write the low-frequency open-loop gain as $A_0 = g_{m1,2}(r_{O2}||r_{O4})$. Sketched in Fig. 11.22(b), $|Z_{out}|$ begins at $(1/g_{m5})/[1 + g_{m1,2}(r_{O2}||r_{O4})]$ at low frequencies and begins to rise beyond the zero frequency, $\omega_z = 1/[(r_{O2}||r_{O4})C_A]$. We observe that Z_{out} has an inductive behavior up to the pole frequency, $\omega_p \approx g_{m1,2}/C_A$, above which it remains approximately equal to $1/g_{m5}$.

The minimum supply voltage is given by $V_{ISS} + V_{GS2} + V_{GS5} + |V_{DS4}|$, where V_{ISS} denotes the voltage necessary for I_{SS} . Thus, V_{DD} may need to be as high as 1.2 to 1.5 V. The key point here is that, while CDACs consume minimal power, their reference buffers do not.

The reference buffer in Fig. 11.22(a) merits two other remarks. First, when C_L has a zero initial condition and switches to node X, V_X jumps down momentarily because the charging current for C_L must be provided by M_5 . If large enough, this jump turns off M_2 , causing node A to slew (why?). That is, the large-signal transient response is quite different from the small-signal behavior predicted by Fig. 11.22(b). Second, the output noise of the buffer travels to the CDAC output, degrading the performance. The reader is encouraged to compute this noise.

11.7 Split CDACs

At high resolutions, CDACs occupy a large area and present severe signal distribution challenges (Section 11.9). For example, an 8-bit differential thermometric CDAC employs 510 switching units driven by a thermometer code. We can reduce the complexity through the use of “split CDACs.”

Let us begin with the 3-bit binary-weighted CDAC shown in Fig. 11.23(a). We wish to construct the Thevenin equivalent of the section within the dashed box. Disconnecting C_1 , we express the Thevenin voltage as

$$V_{Thev} = D_3 \left(\frac{4}{7} V_{REF} \right) + D_2 \left(\frac{2}{7} V_{REF} \right) + D_1 \left(\frac{1}{7} V_{REF} \right). \quad (11.28)$$

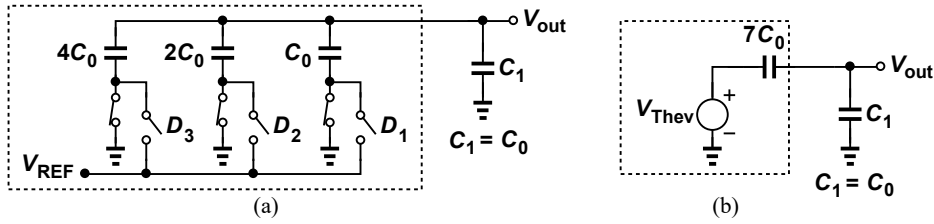


Figure 11.23 (a) A 3-bit CDAC, and (b) its equivalent circuit.

For the Thevenin impedance, we keep the bottom plates of the capacitors at zero or V_{REF} and recognize that the circuit reduces to $7C_0$. Figure 11.23(b) depicts the result. The reader is encouraged to verify that V_{out} jumps by $V_{REF}/2$ if $D_3D_2D_1 = 100$.

Next, consider the arrangement shown in Fig. 11.24(a), where α and β are integers and the output of

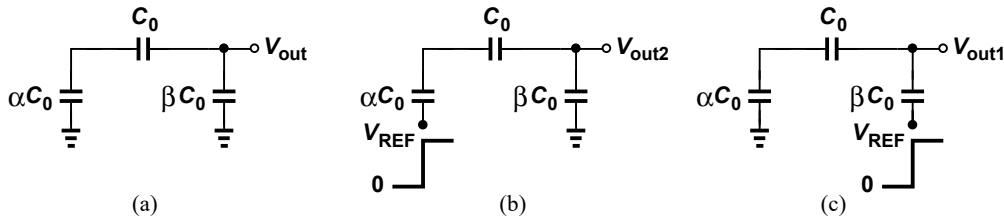


Figure 11.24 (a) A circuit using a bridge capacitor, C_0 , (b) a step applied to the bottom plate of αC_0 , and (c) a step applied to the bottom plate of βC_0 .

interest is sensed on the top plate of βC_0 . Suppose the bottom plate of αC_0 swings from zero to V_{REF} [Fig. 11.24(b)]. To obtain V_{out2} , we compute the series combination of αC_0 and C_0 as $[\alpha/(\alpha + 1)]C_0$, and write

$$V_{out2} = \frac{\frac{\alpha}{\alpha + 1}C_0}{\frac{\alpha}{\alpha + 1}C_0 + \beta C_0} V_{REF}. \quad (11.29)$$

Now, let us assume the bottom plate of βC_0 jumps by V_{REF} [Fig. 11.24(c)], and write

$$V_{out1} = \frac{\beta C_0}{\beta C_0 + \frac{\alpha}{\alpha + 1}C_0} V_{REF}. \quad (11.30)$$

It follows that

$$\frac{V_{out2}}{V_{out1}} = \frac{\alpha}{\alpha + 1} \frac{1}{\beta}. \quad (11.31)$$

As an example, we select $\alpha = \beta = 7$ and hence $V_{out2}/V_{out1} = 1/8$. The key point here is that proper choice of α and β yields binary-weighted attenuation factors for the bottom-plate steps.

At this point, we can utilize the section in the dashed box in Fig. 11.23(a) for both αC_0 and βC_0 in Fig. 11.24(a). Depicted in Fig. 11.25(a), the result is a 6-bit “split DAC” requiring only 15 unit capacitors. We call $C_S = C_0$ the “splitting” or the “bridge” capacitor. Note the symmetry of the circuit. Also, the unswitched capacitor, C_1 , in Fig. 11.23 is absent here. We examine the architecture by allowing the bottom plate of C_a or C_b to jump by V_{REF} . Due to the symmetry, the jump at X in the former case is equal to the jump at Y in

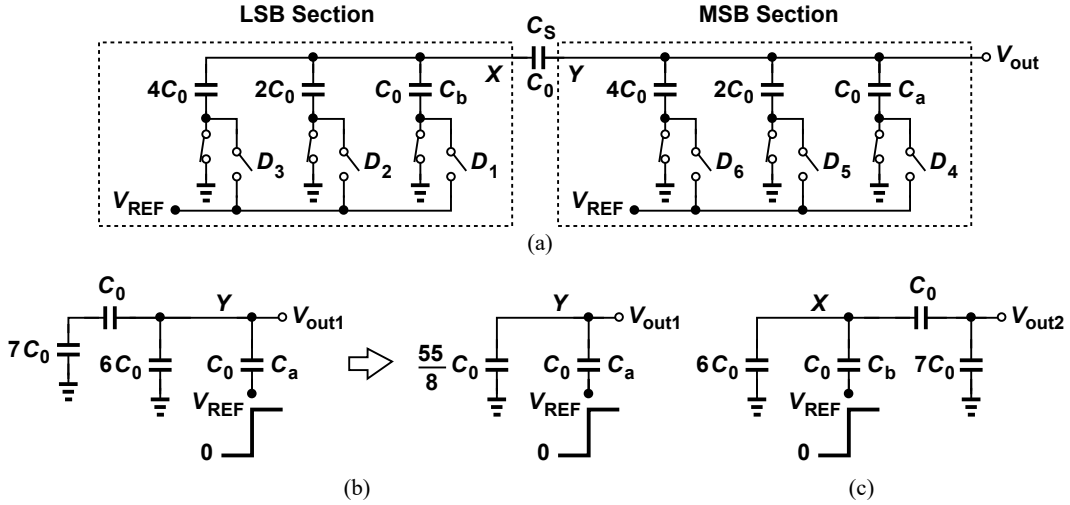


Figure 11.25 (a) A CDAC using a bridge capacitor, (b) effect of a step on the bottom plate of C_a , and (c) effect of a step on the bottom plate of C_b .

the latter. For the former, the equivalent circuit in Fig. 11.25(b) yields

$$V_{out1} = \frac{C_0}{C_0 + \frac{55}{8}C_0} V_{REF} \quad (11.32)$$

$$= \frac{8}{63} V_{REF}. \quad (11.33)$$

For the latter, we have from Fig. 11.25(c):

$$V_X = \frac{8}{63} V_{REF} \quad (11.34)$$

and hence

$$V_{out2} = \frac{C_0}{8C_0} V_X \quad (11.35)$$

$$= \frac{1}{63} V_{REF}. \quad (11.36)$$

It follows that the C_a and C_b branches bear a weighting factor of 8. The other branches display similar relationships.

We thus have for V_{out} in Fig. 11.25(a):

$$V_{out} = D_6 \frac{32}{63} V_{REF} + D_5 \frac{16}{63} V_{REF} + \cdots + D_1 \frac{1}{63} V_{REF}. \quad (11.37)$$

This expression reveals two interesting differences between the split CDAC and a topology such as that in Fig. 11.3. (1) The output LSB size is equal to $V_{REF}/63$ rather than $V_{REF}/64$. (2) The output full scale, i.e., V_{out} for $D_6 D_5 \cdots D_1 = 11 \cdots 1$, is equal to $(63/63)V_{REF} = V_{REF}$ rather than $(63/64)V_{REF}$.

Example 11.8

Redesign the split CDAC of Fig. 11.25(a) for a resolution of 8 bits.

Solution

Recall from Section 10.3 that binary-weighted DACs require tighter matching than their thermometric counterparts. We thus configure the MSB section as a thermometric array, arriving at the arrangement shown in Fig. 11.26. The circuit incorporates 31 unit capacitors. The reader is encouraged to repeat the calculations leading to (11.33) and (11.36) for this design.

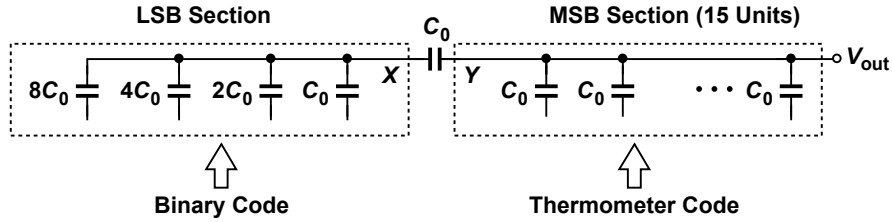


Figure 11.26 An 8-bit CDAC using a bridge capacitor.

Effect of Parasitics The principal drawback of split CDACs stems from the parasitic capacitances on the top plates of the MSB and LSB sections. Recall that the switching units suffer from a finite top-plate parasitic. Moreover, the splitting capacitor, C_S in Fig. 11.25(a), introduces its own top- and bottom-plate parasitics at nodes X and Y . To quantify the resulting error, let us redraw the circuits in Figs. 11.25(b) and 11.25(c) and compute the ratio of the outputs while these capacitances, C_X and C_Y , are included. As shown in Fig. 11.27(a), C_X and C_Y appear between their corresponding nodes and ground. We have

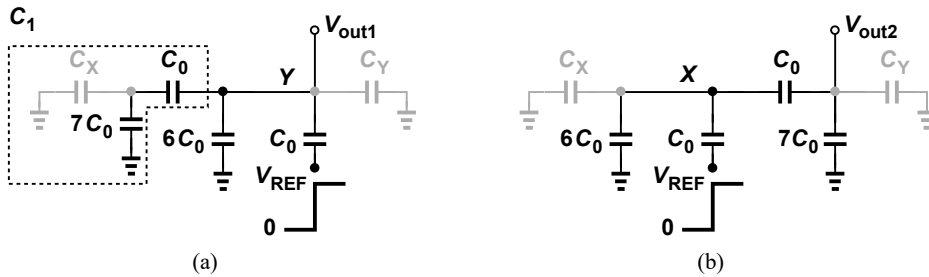


Figure 11.27 Effect of top-plate parasitics for a step applied to the bottom plate of (a) the capacitor on the right, and (b) the capacitor on the left.

$$C_1 = \frac{C_0(7C_0 + C_X)}{8C_0 + C_X} \quad (11.38)$$

and

$$V_{out1} = \frac{C_0}{C_0 + 6C_0 + C_1 + C_Y} V_{REF} \quad (11.39)$$

$$= \frac{C_0}{7C_0 + \frac{C_0(7C_0 + C_X)}{8C_0 + C_X} + C_Y} V_{REF} \quad (11.40)$$

$$= \frac{C_0(8C_0 + C_X)}{(7C_0 + C_Y)(8C_0 + C_X) + C_0(7C_0 + C_X)} V_{REF}. \quad (11.41)$$

Next, we turn to Fig. 11.27(b) and note that V_X can be obtained by swapping C_X and C_Y in (11.41) (why?):

$$V_X = \frac{C_0(8C_0 + C_Y)}{(7C_0 + C_X)(8C_0 + C_Y) + C_0(7C_0 + C_Y)} V_{REF}. \quad (11.42)$$

Also, $V_{out2} = [C_0/(8C_0 + C_Y)]V_X$. It follows that

$$V_{out2} = \frac{C_0^2}{(7C_0 + C_X)(8C_0 + C_Y) + C_0(7C_0 + C_Y)} V_{REF}. \quad (11.43)$$

The weighting factor between the MSB and the LSB sections is now given by

$$\frac{V_{out1}}{V_{out2}} = \frac{8C_0 + C_X}{C_0} \cdot \frac{(7C_0 + C_X)(8C_0 + C_Y) + C_0(7C_0 + C_Y)}{(7C_0 + C_Y)(8C_0 + C_X) + C_0(7C_0 + C_X)} \quad (11.44)$$

$$= 8 + \frac{C_X}{C_0}. \quad (11.45)$$

Note that C_Y introduces only gain error. Neglecting C_Y , we simplify V_{out1} and V_{out2} and write

$$V_{out} = (4D_6 + 2D_5 + D_4) \frac{8C_0 + C_X}{63C_0 + 8C_X} V_{REF} \quad (11.46)$$

$$+ (4D_3 + 2D_2 + D_1) \frac{C_0}{63C_0 + 8C_X} V_{REF}. \quad (11.47)$$

Example 11.9

Compute the DNL of the above CDAC as D_{in} goes from 000111 to 001000.

Solution

For $D_{in} = 000111$, Eq. (11.47) gives an output voltage of $[7C_0/(63C_0 + 8C_X)]V_{REF}$. For $D_{in} = 001000$, on the other hand, V_{out} is equal to $[(8C_0 + C_X)/(63C_0 + 8C_X)]V_{REF}$. We find the difference and subtract 1 LSB from it:

$$\text{DNL} = \frac{56C_X}{63^2C_0} V_{REF}. \quad (11.48)$$

Thus, in this 6-bit example, C_X/C_0 must be less than about 0.2 for $\text{DNL} < 0.2$ LSB. This restriction becomes tighter for higher resolutions.

It is instructive to sketch the input-output characteristic of the split CDAC shown in Fig. 11.25(a). We use Eq. (11.47) to predict how V_{out} rises with D_{in} and compute its undesirable jumps. From Example 11.9, we note that V_{out} follows the ideal trajectory as the input goes from 000000 to 000111 (Fig. 11.28). At the next code transition, DNL occurs. The characteristic then continues with uniform steps up to $D_{in} = 001111$, at which Eq. (11.47) yields

$$V_{out} = \frac{15C_0 + C_X}{63C_0 + 8C_X} V_{REF}. \quad (11.49)$$

For $D_{in} = 010000$, we have

$$V_{out} = \frac{16C_0 + 2C_X}{63C_0 + 8C_X} V_{REF}. \quad (11.50)$$

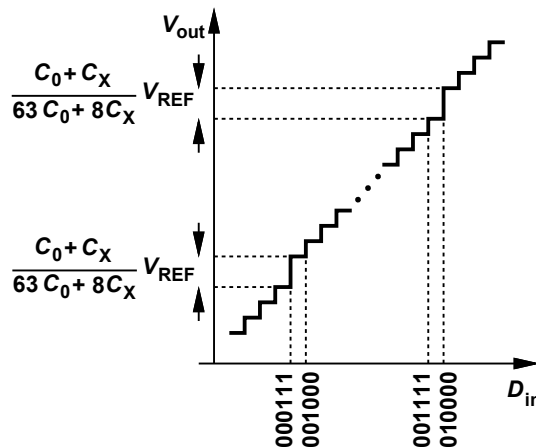


Figure 11.28 DAC characteristic illustrating the effect of top-plate parasitics.

Subtracting 1 LSB from the difference, we obtain the DNL:

$$\text{DNL} = \frac{56C_X}{63^2C_0}V_{REF}. \quad (11.51)$$

That is, for every eight code transitions, there is a DNL of $[56C_X/(63^2C_0)]V_{REF}$. These jumps accumulate and translate to both gain error and INL, as studied in Problem 11.17.

In order to minimize the parasitics at X and Y in Fig. 11.27, we employ the structures described in Section 11.5. Also, since C_S inevitably suffers from a large bottom-plate parasitic, we tie this plate to node Y because our analysis indicates that C_Y is less critical than C_X .

11.8 Choice of Unit Capacitance Value

In the design of CDACs, we must first decide on the unit capacitance value, C_0 , and then select the dimensions of the switches. The choice of C_0 is dictated by two factors, namely, kT/C noise and matching.

Recall that the output node of CDACs is reset to a desirable common-mode level by a switch in every cycle (Fig. 11.29). Upon turning off, this switch deposits kT/C noise on the entire capacitor array, corrupting the

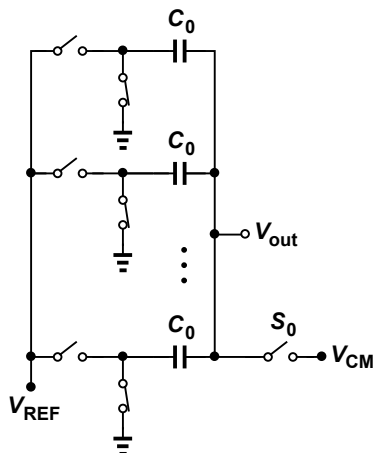


Figure 11.29 DAC with reset switch.

analog output generated afterwards. For a simple array comprising m units, the noise voltage is given by

$$V_{n,\text{rms}}^2 = \frac{kT}{mC_0}. \quad (11.52)$$

We must ensure that $V_{n,\text{rms}}$ is a small fraction of the output LSB size, e.g., $V_{n,\text{rms}} \approx 0.2 \text{ LSB}$, and hence

$$\sqrt{\frac{kT}{mC_0}} \approx 0.2 \frac{V_{REF}}{m}. \quad (11.53)$$

That is,

$$C_0 \approx \frac{25mkT}{V_{REF}^2}. \quad (11.54)$$

For example, if $m = 256$ and $V_{REF} = 0.5 \text{ V}$, we have $C_0 \geq 0.106 \text{ fF}$, a fairly small value.

After S_0 turns off, the DAC output voltage inherits the kT/C noise. Now, the DAC switches contribute thermal noise as well, as studied in the example below.

Example 11.10

Suppose the CDAC of Fig. 11.29 has entered the evaluation mode. Explain how the thermal noise of the bottom-plate switches manifests itself at the output.

Solution

Replacing V_{REF} with ac ground and simplifying the circuit as shown in Fig. 11.30(a), we observe that all

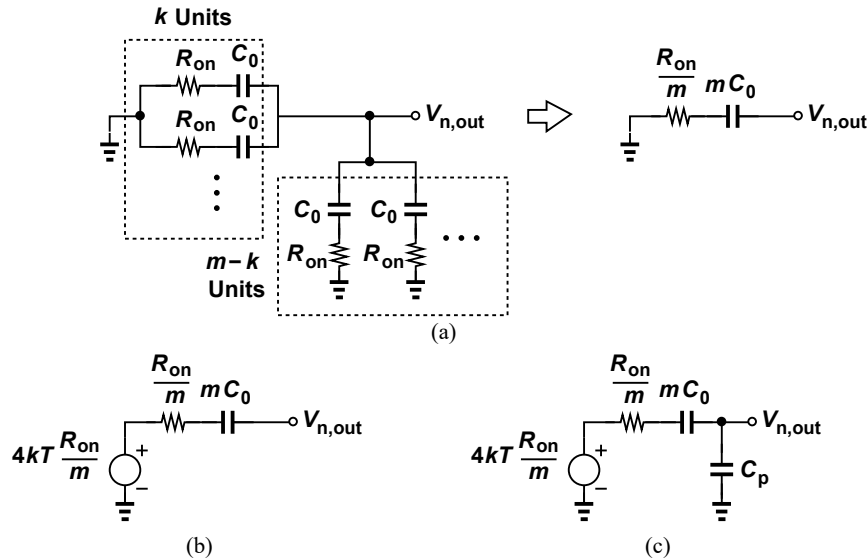


Figure 11.30 (a) CDAC equivalent circuit including switch resistances, (b) noise model, and (c) effect of parasitic capacitance.

of the branches appear in parallel and can be merged into one (why?). The noise equivalent circuit therefore consists of a voltage having a white spectrum given by $4kT R_{on}/m$ [Fig. 11.30(b)]. If the load driven by the CDAC is negligible, this noise simply appears at the output, implying an *infinite* integrated power because nothing appears to limit the bandwidth. With a (parasitic) load capacitance of C_p [Fig. 11.30(c)], on the other hand, the total integrated output noise is equal to kT/C_p if $mC_0 \gg C_p$.

Equation (11.54) yields a small value for C_0 in an 8-bit system, raising the possibility that the unit capacitor value is dictated by mismatch rather than by noise. Since the mismatch between capacitors is inversely proportional to their area, we must ponder the physical dimensions of $C_0 = 0.106$ fF. To maximize the area, we can select two plates with a large separation, e.g., metal 9 and metal 1 (Fig. 11.31). A typical value for the capacitance between these two metal layers is $10 \text{ aF}/\mu\text{m}^2$ for the parallel-plate component and $30 \text{ aF}/\mu\text{m}$

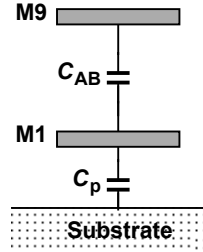


Figure 11.31 Realization of a small capacitor.

for the fringe component. Thus, two $0.8 \mu\text{m} \times 0.8 \mu\text{m}$ plates provide the desired capacitance. These dimensions typically yield a mismatch of less than a few percent, adequate for an 8-bit DAC because Eq. (11.20) stipulates a tolerable mismatch of 4.4% for $m = 256$.

The structure of Fig. 11.31 entails two issues. First, the bottom-plate parasitic is *greater* than C_{AB} (why?), a minor issue in the 8-bit example. Second, when placed next to the other units, the equivalent capacitance incurs significant errors in a manner similar to that in Fig. 11.14(d).

It is instructive to compare the kT/C noise requirement in (11.54) and the mismatch bound in (11.20):

$$C_0 \approx \frac{25mkT}{V_{REF}^2} \quad (11.55)$$

$$\frac{\Delta C}{C_0} < \frac{2}{\sqrt{m}}. \quad (11.56)$$

The key point here is that the former grows with m while the latter falls with $\sqrt{m}$. That is, at higher resolutions, the former dominates. For example, if m goes from 256 to 4096 and $V_{REF} = 0.5 \text{ V}$, then $C_0 = 1.7 \text{ fF}$ while $\Delta C/C_0 < 1.1\%$. Compared to the 8-bit case, C_0 rises by a factor of 16 whereas the tolerable mismatch becomes tighter by a factor of 4. We conclude that kT/C noise becomes the bottleneck, especially because capacitor mismatches can be removed by calibration.

11.9 Matrix Architecture for CDACs

Recall from Section 10.8 that the large number of current sources in high-resolution IDACs has led to the matrix architecture. The same idea can be applied to CDACs as well. As we will see, however, the distribution of signals and other voltages across CDAC arrays proves challenging. The reader is encouraged to review Section 10.8 first.

Let us consider a 4-bit thermometric CDAC, realized as a 4×4 array (Fig. 11.32). Each cell contains a unit capacitor, at least two switches, and local logic. Moreover, the following five voltages must reach the cells: the row thermometer code, the column thermometer code, the reference, the ground, and V_{DD} . We must decide how these voltages are distributed without degrading the DAC's performance.

We can identify two precision issues here. First, if a unit capacitor in one part of the array does not see *exactly* the same “environment” as another, then some mismatch arises between the two. We can consider this effect a “static” error. In Fig. 11.32, for example, the top capacitors see the column thermometer logic whereas the bottom ones do not. Second, if signal transitions couple to the capacitors, they can introduce “dynamic” errors in the output voltage. This issue proves serious and must be investigated carefully. The following example is our first step toward understanding this coupling effect.

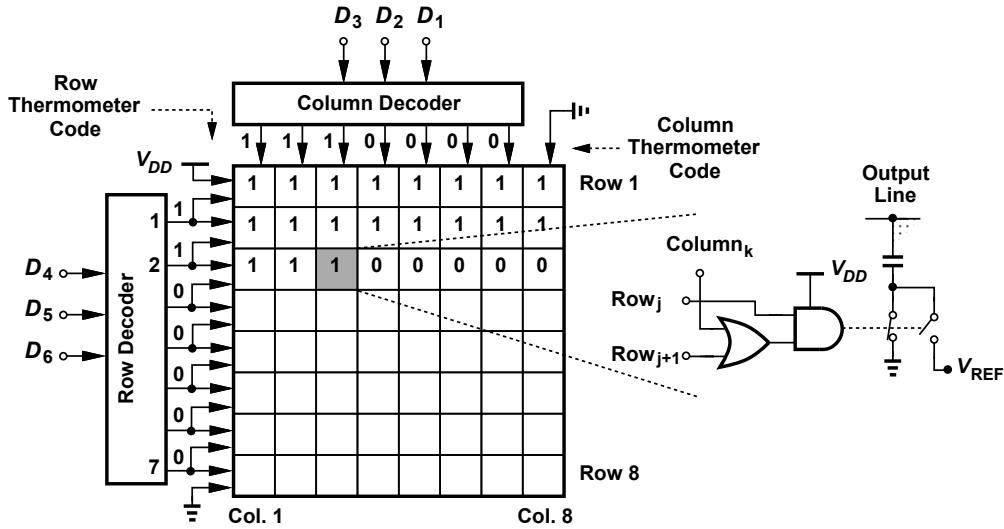


Figure 11.32 Matrix CDAC architecture.

Example 11.11

As shown in Fig. 11.33(a), a digital signal line travels in the vicinity of a unit capacitor in a CDAC, thus coupling to the capacitor through a fringe component, C_f . Explain whether this coupling corrupts the DAC output signal, assuming that the signal resides on metal 9 or metal 8.

Solution

With the aid of the side view depicted in Fig. 11.33(b), we observe that a transition on the digital line in metal 9 leads to an output corruption equal to

$$\Delta V_{out} = \frac{C_f}{C_f + mC_0} V_{DD} \quad (11.57)$$

$$\approx \frac{C_f}{mC_0} V_{DD}. \quad (11.58)$$

Even though $C_f \ll mC_0$, ΔV_{out} may be significant because of the digital signal's large swing.

Let us now assume that the digital line travels in metal 8 and mostly couples to the bottom plate of C_0 [Fig. 11.33(c)]. Does a jump on the left terminal of C_f corrupt V_{out} ? Suppose S_1 is on and modeled by an on-resistance R_{on1} . The equivalent circuit in Fig. 11.33(d) suggests that V_X initially jumps (because C_f acts as a short circuit at $t = 0^+$) but subsequently discharges to zero through R_{on1} . The output voltage follows V_X according to $V_{out} = [C_0/(mC_0)]V_X$, exhibiting a similar glitch. Whether or not we consider this effect to be corruption depends on the environment in which the DAC serves. Since the glitch decays with a very small time constant, $R_{on1}C_f$, we can ignore it if we sense the DAC output only slightly after $t = 0$. This is possible when the DAC appears within an ADC, the most common usage of CDACs. In summary, we prefer the scenario in Fig. 11.33(c) over that in Fig. 11.33(b).

We must now decide how the five voltages mentioned previously are distributed so as to produce negligible static and dynamic errors. We assume the CMOS process of interest provides nine metal layers. Figure 11.34 depicts an example of a unit cell design. The capacitor is formed by five layers and properly shielded (Section 11.5) so as to minimize leakage fringe electric field lines from its top plate. The row thermometer code, D_R , runs from left to right in metal 4, and the other lines travel into the page in metal 3. The control circuit and

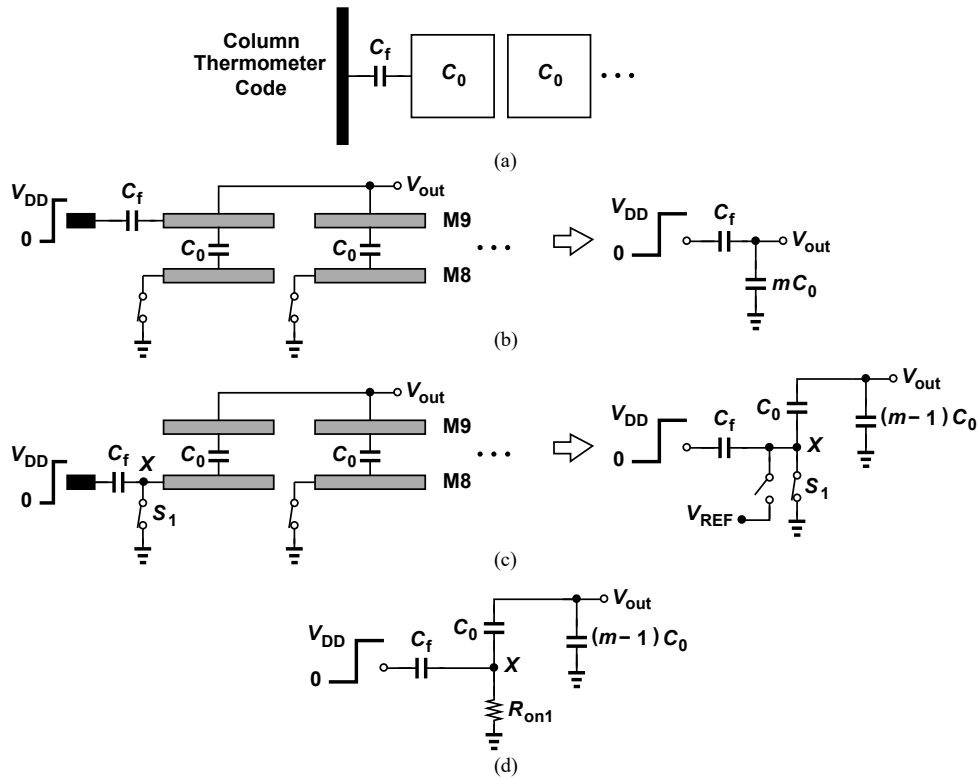


Figure 11.33 (a) Coupling of a digital line to a capacitor, (b) side view, (c) use of metal-8 for digital line, and (d) equivalent circuit.

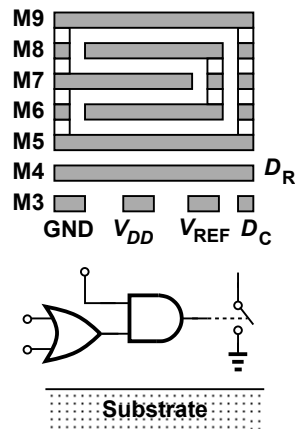


Figure 11.34 Side view of a unit-cell realization.

the bottom-plate switches are realized below these lines and use metal 1 and metal 2.⁶ We observe that (a) all of the unit cells have exactly the same structure, avoiding systematic mismatches, and (b) transitions in the row and column thermometer codes, D_R and D_C , respectively, couple to only the bottom plate of the unit capacitor, introducing minimal dynamic errors.

Our final point with respect to symmetry in Fig. 11.32 is the need for adding dummy cells at the edges of

⁶Each cell also needs the adjacent row thermometer code, which can be routed in metal 4.

the matrix, as we did in the case of IDACs (Section 10.8). The idea is to surround the matrix by a set of cells that are *identical* to the inner cells (e.g., similar to the structure in Fig. 11.34) so that each inner cell sees the same exact environment around it.

Problems

- 11.1. Suppose C_1 and C_2 in Fig. 11.1(b) are equal to C_u and $C_u + \Delta C$, respectively. Determine the output levels of the DAC. Calculate the offset, gain error, DNL, and INL of the circuit.
- 11.2. The DAC of Fig. 11.1(b) drives a small load capacitance equal to C_L . Determine the output levels and find the gain error, offset, DNL, and INL.
- 11.3. The CDAC of Fig. 11.3 is driven by thermometer code. Draw a binary version of it.
- 11.4. Capacitor C_2 is accidentally omitted from the CDAC of Fig. 11.2(a). Plot the input-output characteristic and compute the offset, gain error, DNL, and INL.
- 11.5. Repeat the previous problem if C_2 is present but S_{2b} is always on and S_{2a} is always off.
- 11.6. Suppose C_1 in Fig. 11.3 is equal to $C_u + \Delta C$ while the other capacitors are equal to C_u . Plot the input-output characteristic and determine the offset, gain error, DNL, and INL.
- 11.7. A CDAC must drive a load capacitance, C_L . Does the gain error depend on whether the circuit operates with a thermometer code or a binary code?
- 11.8. Compute the transfer function of the circuit shown in Fig. 11.9(b) from V_{in} to V_{out} and show that it contains a pole and a zero that cancel each other.
- 11.9. The circuit of Fig. 11.9(b) must drive a load capacitance of C_L . Determine the step response at the output. (Hint: Thevenin is your friend.)
- 11.10. The CDAC of Fig. 11.10(a) is designed for a resolution of 8 bits. (a) How much is the tolerable capacitor mismatch? (b) If we wish to increase the resolution to 9 bits, by what factor does the area of the DAC increase? Assume unit capacitor mismatch is inversely proportional to the area.
- 11.11. Suppose the CDAC of Fig. 11.10(a) is tied to the buffer shown in Fig. 11.22(a). We note that the DAC capacitance introduces a pole at the buffer output, degrading the phase margin and increasing the reference settling time. Estimate which DAC input code creates the worst-case scenario.
- 11.12. Suppose the bridge capacitor in Fig. 11.24(a) is equal to $C_0 + \Delta C$, where $\Delta C \ll C_0$. Calculate the outputs in Figs. 11.24(b) and (c) if $\alpha = \beta = 7$.
- 11.13. The circuit of Fig. 11.24(a) must drive a load capacitance of C_L at its output. Determine V_{out} in Figs. 11.24(b) and (c). Does the circuit still provide binary-weighted attenuation factors for the bottom-plate steps?
- 11.14. Repeat the analysis of Fig. 11.25(a) by allowing only the bottom plates of each $2C_0$ capacitor to swing from zero to V_{REF} .
- 11.15. In the CDAC of Fig. 11.25(a), we have assumed that the LSB and MSB sections have the same resolution. Is it possible to design the circuit for 6 bits but with 2 bits driving the MSB section and 4 bits driving the LSB section? Explain in detail.
- 11.16. Does the CDAC shown in Fig. 11.27 incur a DNL when D_{in} goes from 000011 to 000100? Explain.
- 11.17. Determine the gain error and the INL profile for the characteristic shown in Fig. 11.28.
- 11.18. Determine the transfer function of the circuit shown in Fig. 11.30(c) and compute the total integrated noise at the output.
- 11.19. We wish to design the CDAC of Fig. 11.29 for a resolution of 10 bits with $V_{REF} = 1024$ mV. If unit capacitor mismatch, $\Delta C/C$, is 1% and the DAC output kT/C noise must not exceed 0.2 LSB, determine whether the design is dominated by mismatches or by noise.
- 11.20. Consider the circuit of Fig. 11.24(a) and suppose a reset switch is tied from the output node to ground.
 - (a) Compute the kT/C noise at the output.
 - (b) Repeat this calculation for the split DAC of Fig. 11.26.

Introduction to Nyquist-Rate ADCs

As mentioned in Chapter 1, an analog-to-digital converter (ADC) receives a continuous-time, continuous-amplitude signal and transforms it to a discrete-time, discrete-amplitude representation. We also saw that this transformation requires sampling and quantization, and we studied the former in Chapter 3. We devote the remaining chapters of this book to ADC design.

The term “ADC” can be confusing as it refers to the sampler-quantizer cascade in some cases but to the quantizer itself in some other cases. Moreover, it is possible that the same circuit performs sampling and quantization (Chapter 13). Nevertheless, the exact meaning of the term should be clear from the context.

In this chapter, we first examine the concept of quantization more closely. Next, we introduce the performance parameters of ADCs and describe how they are affected by circuit imperfections. Finally, we describe what we mean by “Nyquist-rate ADCs.”

12.1 General Considerations

An ADC generates a digital output, D_{out} , proportional to its analog input, V_{in} :

$$D_{out} \propto V_{in}. \quad (12.1)$$

Since V_{in} has a dimension but D_{out} does not, a proportionality factor is necessary:

$$D_{out} = \alpha V_{in}. \quad (12.2)$$

Note that α has a dimension of 1/voltage. While the input can assume an infinite number of values, the output is selected from a finite number of codes, i.e., the ADC approximates each input level by one of these codes. As was practiced for DACs (Chapter 8), we can construct the input-output characteristic of ADCs. Depicted in Fig. 12.1 for a 3-bit system, the characteristic plots the equivalent digital output as the analog input varies. We should make two remarks. First, as V_{in} increases from zero, D_{out} remains at 000 up to $V_{in} = \Delta$. Then, D_{out} jumps to 001 and remains at this value up to $V_{in} = 2\Delta$, etc. We note that, for example, all of the analog values between $V_j = 3\Delta$ and $V_{j+1} = 4\Delta$ are approximated by $D_{out} = 011$. We say the ADC “quantizes” the input signal. Second, since the input and output of the ADC are not equal at all times, we expect that quantization introduces some type of error (Section 12.2).

We employ the following terminology with respect to Fig. 12.1: (1) each jump is called a “code transition”; (2) the input difference between the consecutive jumps, Δ , is called the analog least-significant bit (LSB);¹ (3) the input values at which the code transitions occur are called “decision thresholds”; (4) the unit output jumps are called the digital LSB; and (5) the maximum input level (the total input range) is called the “full scale,” V_{FS} .

¹This holds for an ideal quantizer and will be revisited in Section 12.3.

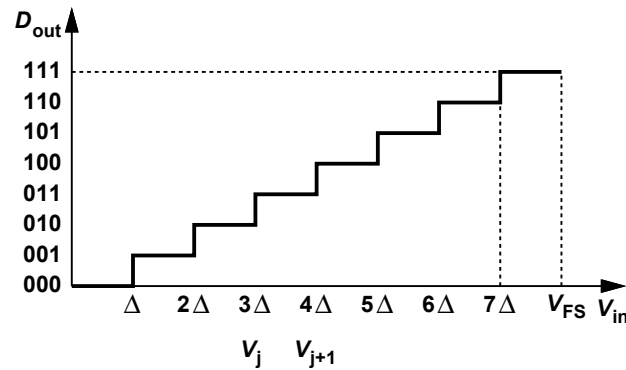


Figure 12.1 A 3-bit ADC characteristic.

12.2 A First Look at Quantization

12.2.1 Basic Concepts

As mentioned in the previous section, we expect that analog-to-digital conversion introduces an error because a quantizer's output is not identical to its input. The phenomenon of quantization can be studied from different perspectives so as to gain interesting insights. We pursue one here and another in Chapter 21.

Let us assume a quantizer characterized by the profile shown in Fig. 12.1. We apply a ramp to the input and examine the output [Fig. 12.2(a)], using the plots in Fig. 12.2(b) to see how V_{in} forms D_{out} . (For ease

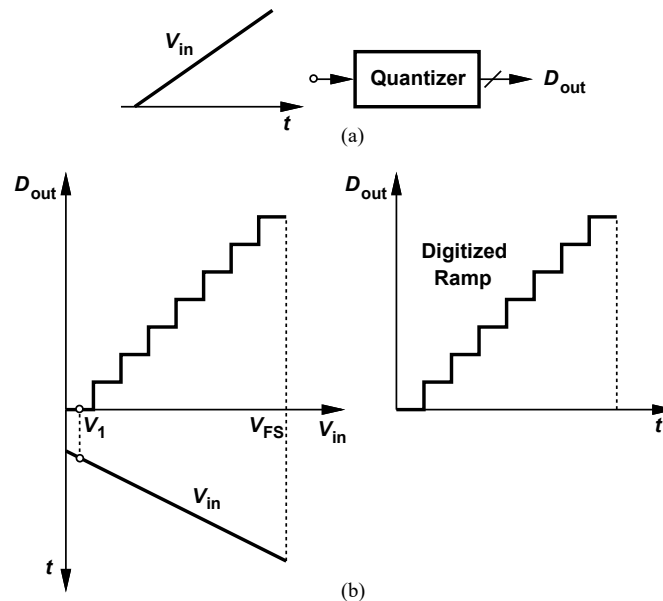


Figure 12.2 (a) A quantizer sensing a ramp, and (b) its output waveform.

of illustration, we plot D_{out} as a continuous-time waveform, as if the ADC were followed by a DAC and we plotted the DAC output.) This arrangement depicts V_{in} versus time, D_{out} versus V_{in} , and D_{out} versus time. As V_{in} reaches V_1 ($= \Delta = 1$ LSB), D_{out} jumps from 000 to 001. This behavior continues for the subsequent codes until V_{in} reaches the full-scale value, V_{FS} . The analog input therefore maps to a “digitized” ramp.

We now loosely define the “quantization error,” ϵ_q , as the difference between the input and output of the quantizer. Superimposing the two as shown in Fig. 12.3(a), and subtracting D_{out} from V_{in} , we obtain ϵ_q as a function of time [Fig. 12.3(b)]. Note that ϵ_q varies between zero and Δ (the analog LSB).

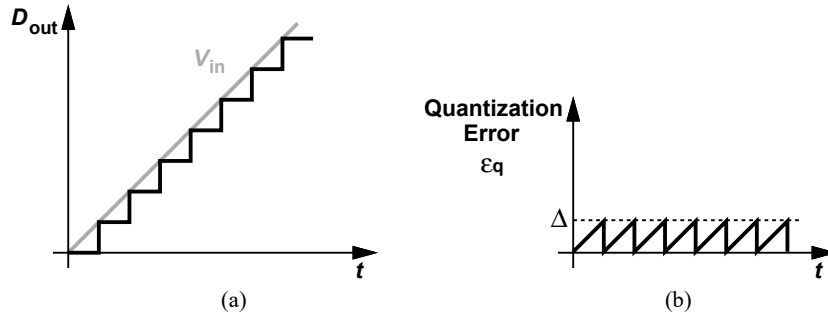


Figure 12.3 (a) Input and outputs waveforms of a quantizer, and (b) the difference between the two.

Example 12.1

For the situation illustrated in Fig. 12.2(a), plot ϵ_q as a function of V_{in} .

Solution

We first plot both D_{out} and V_{in} as a function of V_{in} and then construct $V_{in} - D_{out}$ (Fig. 12.4). The resulting profile suggests that ϵ_q begins at zero for $V_{in} = 0$ and rises linearly as V_{in} increases (because D_{out} remains at zero). At $V_{in} = \Delta$, D_{out} jumps by 1 LSB and $V_{in} = D_{out}$, causing $V_{in} - D_{out}$ to drop to zero, etc. It is important not to confuse the plots in Figs. 12.3(b) and 12.4(b).

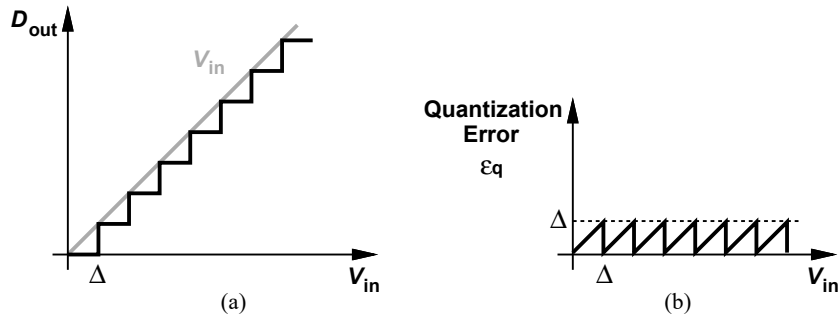


Figure 12.4 Input-output characteristic and quantization error of a quantizer.

The foregoing example teaches two important points. First, since the behavior of ϵ_q versus V_{in} is similar to that of ϵ_q versus time in a ramp test, we can construct the former from the latter. Second, the behavior of ϵ_q versus V_{in} depends on only the level of V_{in} and is independent of the shape of the input in the time domain—just as is D_{out} versus V_{in} .² For example, a sinusoidal V_{in} still generates ϵ_q according to Fig. 12.4, but, of course, a different ϵ_q waveform versus time (Problem 12.1).

12.2.2 Formulation of Quantization Error

We wish to analyze how quantization affects the performance of an ADC. Returning to Figs. 12.3(b) and 12.4(b), we recognize that $0 \leq \epsilon_q \leq \Delta$, i.e., the smaller the analog LSB is, the less ϵ_q is. But we also note that both ϵ_q plots exhibit an average value of $\Delta/2$; this dc component per se does not corrupt the analog signal and is benign in most cases. We then subtract $\Delta/2$ from ϵ_q (Fig. 12.5), observing that the result varies between $-\Delta/2$ and $+\Delta/2$ regardless of the shape of the input waveform. For example, if V_{in} is a random

²This holds only at low input frequencies.

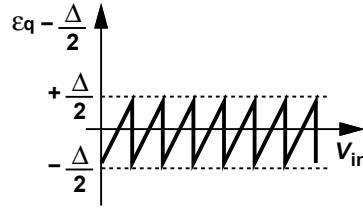


Figure 12.5 Quantization error minus dc offset.

signal, $\epsilon_q - \Delta/2$ fluctuates between those two levels randomly, thereby introducing noise in the digitized waveform. That is, quantization adds noise when the input is random.³ We call $\epsilon_q - \Delta/2$ the “quantization noise” and denote it by n_q .

How do we compute the power of n_q ? We can examine the spectrum of the noise (Chapter 21) or utilize our knowledge of random variables. We know that the “power” of a random variable, x , is given by

$$P_x = \int_{-\infty}^{+\infty} x^2 f(x) dx, \quad (12.3)$$

where $f(x)$ denotes the probability density function (PDF) of x .⁴ For example, a zero-mean Gaussian distribution with a standard deviation of σ carries a power of σ^2 ; hence the useful rule that the rms value of Gaussian noise is equal to its σ . We can apply Eq. (12.3) to quantization noise if $f(x)$ is known. Figure 12.5 suggests that $n_q = \epsilon_q - \Delta/2$ is distributed between $-\Delta/2$ and $+\Delta/2$, but the shape of the distribution for a random input is not known. A simplification that proves relatively accurate in practice is to assume that the quantization noise has a uniform distribution (Fig. 12.6). The height of the PDF is equal to $1/\Delta$ so that the

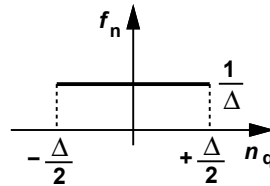


Figure 12.6 Uniform distribution assumed for quantization error.

total area amounts to unity. We can then write from (12.3)

$$P_{nq} = \int_{-\Delta/2}^{+\Delta/2} n_q^2 \frac{1}{\Delta} dn_q \quad (12.4)$$

$$= \frac{1}{\Delta} \int_{-\Delta/2}^{+\Delta/2} n_q^2 dn_q \quad (12.5)$$

$$= \frac{\Delta^2}{12}. \quad (12.6)$$

This important result signifies the corruption introduced by the quantizer. Note that the unit of P_{nq} is V^2 .

Signal-to-Noise Ratio Equation (12.6) implies that P_{nq} can be reduced by scaling Δ down. But what do we do with the full-scale voltage, V_{FS} ? If V_{FS} is maintained constant, then the *number* of quantization steps, m , (the “resolution”) must increase proportionally. Figure 12.7 depicts an example of doubling

³We defer the case of periodic inputs to Chapter 21.

⁴We can view this expression as the average value of x^2 .

the resolution. We expect that the output signal-to-noise ratio (SNR) rises with m . For an N -bit quantizer,

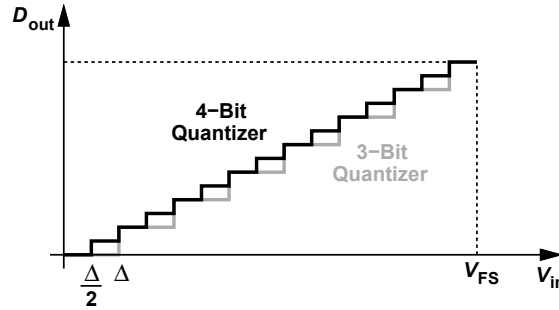


Figure 12.7 Effect of doubling an ADC's resolution.

we have $m = 2^N$. To verify, suppose the full-scale input applied to an N -bit quantizer is expressed as $V_{in} = V_0 \cos \omega_0 t + V_0$, where $V_0 = V_{FS}/2$ (Fig. 12.8). The signal power is given by $V_0^2/2$ and the analog LSB by $\Delta = V_{FS}/2^N = V_0/2^{N-1}$. It follows that

$$\text{SNR} = \frac{\frac{V_0^2}{2}}{\frac{\Delta^2}{12}} \quad (12.7)$$

$$= \frac{3}{2} 2^{2N}. \quad (12.8)$$

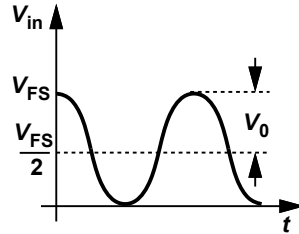


Figure 12.8 Full-scale sinusoid applied to a quantizer.

Taking 10 log of both sides, we have

$$\text{SNR(dB)} = 20N \log 2 + 1.76 \quad (12.9)$$

$$= 6.02N + 1.76. \quad (12.10)$$

Thus, doubling the resolution (i.e., increasing N by 1) raises the SNR by about 6 dB.

Example 12.2

The derivation leading to Eq. (12.10) has assumed that n_q is random and has a uniform distribution. Is this true for a sinusoidal input?

Solution

Strictly speaking, no, it is not. We surmise that a periodic input leads to a periodic output and hence a periodic quantization error. We return to this point in Chapter 21. Nevertheless, (12.6) and (12.10) provide reasonable approximations even for periodic inputs if N is greater than roughly 3.

Equation (12.10) suggests that the SNR is maximized by raising V_0 . Of course, if the input swing exceeds the full scale, the ADC simply generates an output of the form $11 \cdots 1$, i.e., it saturates, introducing high distortion.

12.3 ADC Performance Parameters

In our study of sampling circuits and DACs, we have seen that their performance is quantified by means of numerous parameters. In this section, we present similar parameters for ADCs. The reader is encouraged to first review those of DACs (Chapter 8).

12.3.1 Offset

Consider the simple ADC shown in Fig. 12.9(a). Due to switch charge injection and clock feedthrough, V_X

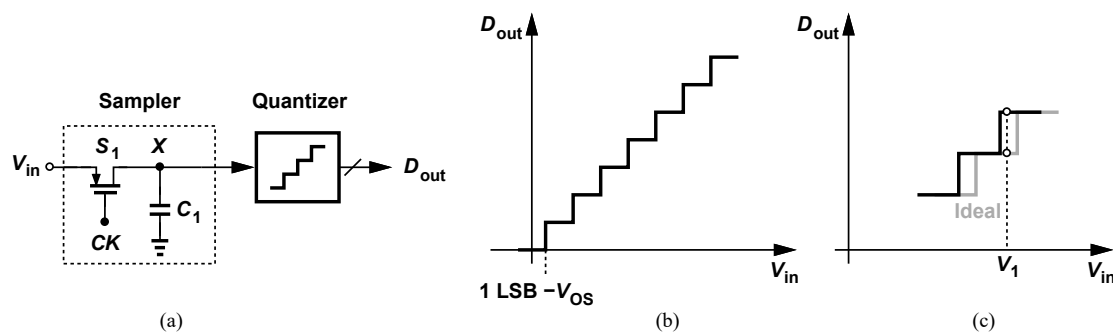


Figure 12.9 (a) A sampler followed by a quantizer, (b) effect of sampler offset on overall characteristic and (c) closeup view of offset.

experiences a positive pedestal equal to V_{OS} when S_1 turns off. That is, if $V_{in} = 0$, $V_X > 0$. More precisely, the first decision threshold of the quantizer is shifted to the left, as illustrated in Fig. 12.9(b). We say the ADC suffers from an offset. The nonidealities within the quantizer itself can introduce offset as well.

It is possible to view the effect of offset as a *vertical* shift. Turning to Fig. 12.9(c), we observe that the digital output corresponding to $V_{in} = V_1$ is *greater* than the ideal case, signifying an upward shift.

12.3.2 Gain Error

Gain error is defined as the departure of the characteristic's slope from its ideal value. Depicted in Fig. 12.10, this phenomenon arises because the analog LSB is greater or less than the nominal. In other words, if $\Delta_2 < \Delta_1$, the gain error is positive and vice versa. We can also see that the full-scale voltage changes. In fact, the *cause* of gain error can be a change in V_{FS} as we have $\Delta = V_{FS}/2^N$.

Example 12.3

A digital thermometer displays temperatures ranging from -20°C to $+60^\circ \text{C}$ with a resolution of 1° . If the ADC used therein exhibits offset and gain error, explain how the temperature readings are affected.

Solution

The ADC offset simply shifts all of the readings up or down by a constant amount. For example, the output values are 2°C greater than the actual temperature.

The ADC gain error multiplies each temperature by a constant coefficient. For example, a gain error of $+2\%$ yields an output equal to 1.02 times the actual temperature. We can see that the reading error *grows* as the temperature reaches the full scale.

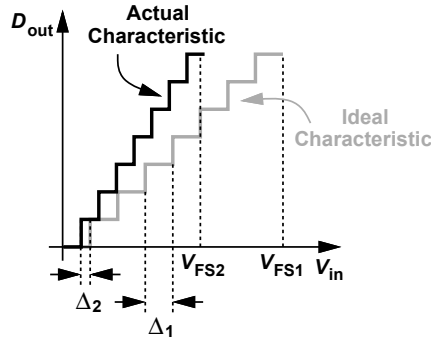


Figure 12.10 Illustration of gain error.

Given an ADC input-output characteristic, how do we compute the gain error? This is not possible if we do not know the ideal analog LSB or the ideal full scale. In contrast to offset, gain error is a *relative* quantity and can only be determined if both characteristics in Fig. 12.10 are known.

12.3.3 Differential Nonlinearity

Our treatment of quantization in the previous sections has assumed evenly-spaced code transitions. But is that true in practice? We must then ask, what type of circuit creates these transitions? As explained in the subsequent chapters, the decision thresholds are determined by one or more comparators, which inevitably suffer from offsets (Chapter 6). In other words, each transition may depart from its ideal position by a random value after an ADC has been fabricated (Fig. 12.11). That is, the input voltage difference between consecutive transitions may not be equal to Δ .

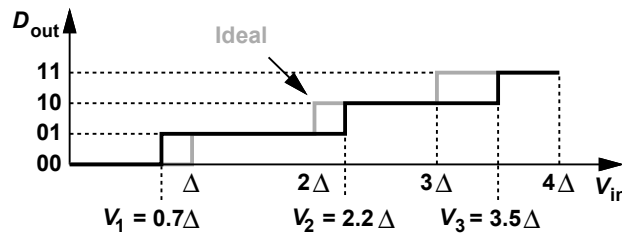


Figure 12.11 DNL in an ADC.

We define the “differential nonlinearity” (DNL) as the deviation in the difference between consecutive transitions from the ideal value of 1 LSB:

$$\text{DNL}_j = V_j - V_{j-1} - 1 \text{ LSB}, \quad (12.11)$$

where V_j and V_{j-1} denote the actual decision thresholds. In Fig. 12.11, for example, we have $\text{DNL}_1 = 0.7\Delta - 0 - \Delta = -0.3 \text{ LSB}$, $\text{DNL}_2 = +0.5 \text{ LSB}$, and $\text{DNL}_3 = +0.3 \text{ LSB}$. Note that there are three DNL values here while the quantizer has four levels.

The reader may wonder how we should plot the DNL: versus V_{in} or versus D_{out} ? In this book, we plot the DNL versus V_{in} as it is more intuitive. In practice, however, it is plotted versus the output “code.” For example, we say the profile shown in Fig. 12.11 exhibits a DNL equal to -0.3 LSB for $D_{out} = 00$, $+0.5 \text{ LSB}$ for $D_{out} = 01$, and $+0.3 \text{ LSB}$ for $D_{out} = 10$. This distinction is unimportant if we are only interested in the maximum DNL.

Example 12.4

An ADC exhibits a DNL of -1 LSB . Explain how its input-output characteristic is distorted. Repeat the exercise if $\text{DNL} = +1 \text{ LSB}$. Assume that only one code transition is offset.

Solution

We write $V_j - V_{j-1} - 1 \text{ LSB} = -1 \text{ LSB}$, obtaining $V_j = V_{j-1}$. Thus, the comparator offset is so large that the two transitions coincide. Illustrated in Fig. 12.12(a), the characteristic displays a jump of 2 LSB at $V_{in} = V_{j-1}$, revealing that code D_j cannot be created for any input value. We say the ADC suffers from a “missing” code.

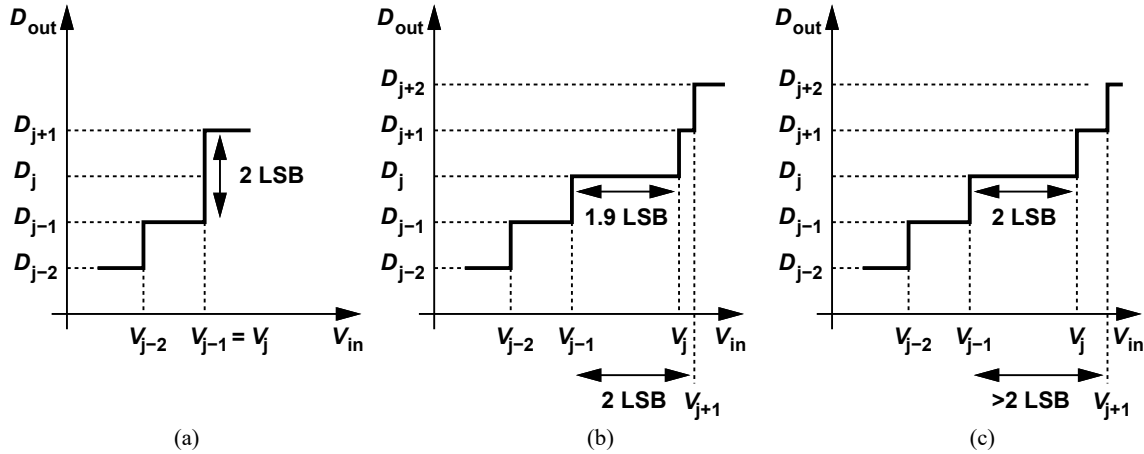


Figure 12.12 (a) Missing code in an ADC, (b) first case of $DNL = +1 \text{ LSB}$, and (c) second case of $DNL = +1 \text{ LSB}$.

To study the case of $DNL = +1 \text{ LSB}$, let us first assume $DNL = +0.9 \text{ LSB}$ [Fig. 12.12(b)]. The corresponding code transitions are 1.9 LSB apart. Now, suppose only the transition at V_j is offset and hence $V_{j+1} - V_{j-1} = 2 \text{ LSB}$; then as DNL_j approaches $+1 \text{ LSB}$, V_j and V_{j+1} coincide, causing code D_{j+1} to vanish.

On the other hand, if both V_j and V_{j+1} are offset and $V_{j+1} - V_{j-1} > 2 \text{ LSB}$, no missing code arises for $DNL_j = +1 \text{ LSB}$ [Fig. 12.12(c)]. In summary, $DNL = -1 \text{ LSB}$ always translates to a missing code but $DNL = +1 \text{ LSB}$ does not.

In ADC design, we target $|DNL| < 1 \text{ LSB}$. In general, differential nonlinearity arises from comparator offset and possibly other circuit imperfections as well. For example, in some ADC architectures, capacitor mismatch leads to DNL. Thus, all DNL contributions must be managed so as to ensure $|DNL| < 1 \text{ LSB}$.

It is important not to confuse the DNL definitions of DACs and ADCs. In the former, the *output* change is considered for DNL computation whereas in the latter, the *input* change is. In other words, the DNL is meaningful for only analog quantities.

Example 12.5

Explain what happens to an ADC's input-output characteristic if the DNL remains positive for part of the input range.

Solution

As shown in Fig. 12.13, the characteristic exhibits a *lower* slope than the ideal value in this range. Conversely, accumulation of negative DNL results in a higher slope.

12.3.4 Integral Nonlinearity

Recall from Chapter 8 that the INL of a DAC represents the curvature in the overall input-output characteristic. We apply the same notion to ADCs. As illustrated in Fig. 12.14(a), we pass a straight line through the

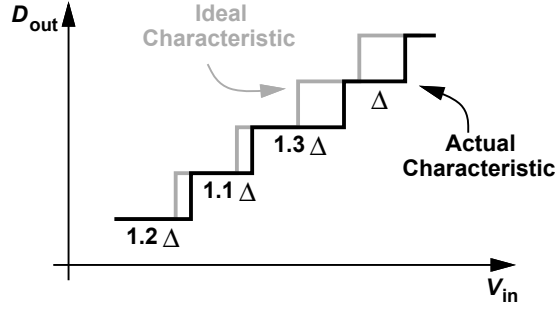


Figure 12.13 A characteristic having $\text{DNL} > 0$ for part of the range.

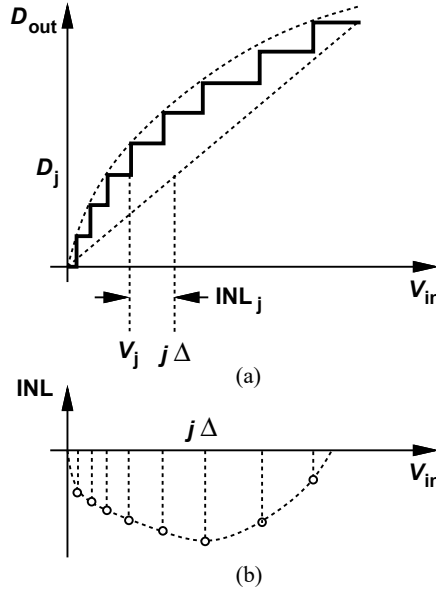


Figure 12.14 (a) Definition of INL in an ADC, and (b) INL profile.

end points and view the *horizontal* difference between the characteristic and this line as the INL profile [Fig. 12.14(b)]. Specifically, if a code transition occurs at V_j rather than at $j\Delta$, we define INL_j as

$$\text{INL}_j = V_j - j\Delta. \quad (12.12)$$

We thus observe that, in both DACs and ADCs, the DNL and the INL are measured along the analog axis. Our comment in the previous section about plotting the profile versus the analog input or versus the output code applies to the INL as well.

Relation Between INL and DNL We have seen in Chapter 8 that the INL of DACs is equal to the integral of their DNL. We then surmise that the same holds for ADCs. Let us write $\text{DNL}_j = V_j - V_{j-1} - \Delta$ and form the summation (integral) thereof as

$$\sum_{k=1}^j \text{DNL}_k = \sum_{k=1}^j (V_k - V_{k-1}) - j\Delta. \quad (12.13)$$

Here, $j\Delta$ represents the straight line in Fig. 12.14(a), i.e., the ideal decision thresholds.⁵ We also have

$$\sum_{k=1}^j (V_k - V_{k-1}) = (V_j - V_{j-1}) + (V_{j-1} - V_{j-2}) + \cdots + (V_1 - V_0), \quad (12.14)$$

where V_0 corresponds to $V_{in} = 0$ (why?). It follows that this summation is simply equal to V_j . Equation (12.13) thus reduces to

$$\sum_{k=1}^j \text{DNL}_k = V_j - j\Delta, \quad (12.15)$$

which is the same as Eq. (12.12).

Example 12.6

The nonlinear characteristic in Fig. 12.14(a) displays a higher slope for low values of V_{in} and a lower slope for high values. Explain how this behavior leads to DNL and INL.

Solution

The higher slope for V_{in} near zero can be viewed as a positive gain error, indicating analog LSB values *less* than the nominal. That is, the DNL is negative and accumulates to yield a negative INL. Beyond the valley INL, the slope of the characteristic falls, the DNL becomes positive, and its accumulation returns the characteristic to the straight line.

Distinction Between INL and Offset The INL profile example shown in Fig. 12.14(b) remains negative for the entire input range. We can argue that this profile, in fact, contains an *offset*, thus overestimating distortion. To appreciate this point, let us shift the profile up so as to create equal and opposite peaks [Fig. 12.15(a)]. This is equivalent to shifting the straight line in Fig. 12.14(a) to the left by the same amount [Fig. 12.15(b)] (why?). We observe that the ADC input-output characteristic introduces two artifacts in the digitized signal: an offset equal to α and a peak INL of $\pm\alpha$.⁶ For convenience, we still pass the straight line through the end points in this book.

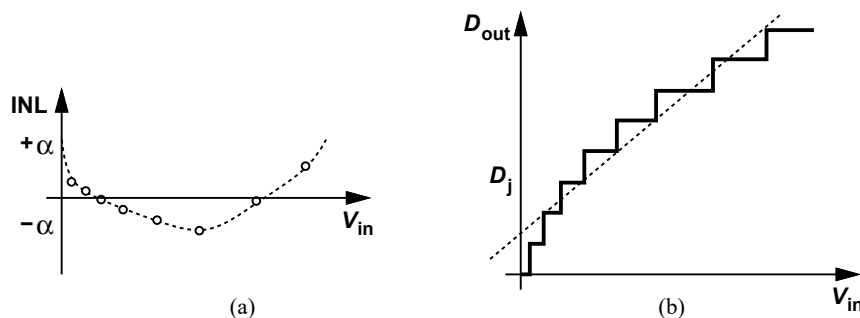


Figure 12.15 (a) INL profile with dc offset removed, and (b) corresponding straight line passing through the characteristic.

INL versus Input Swing We have thus far assumed that the analog input reaches the ADC full-scale voltage. But suppose we apply a smaller input swing, e.g., equal to $V_{FS}/2$ [Fig. 12.16(a)]. Does the signal experience less nonlinearity? Passing a straight line between A and C and finding the difference, we arrive at the two INL

⁵We ignore the effect of offset and gain error here.

⁶Of course, we also have DNL.

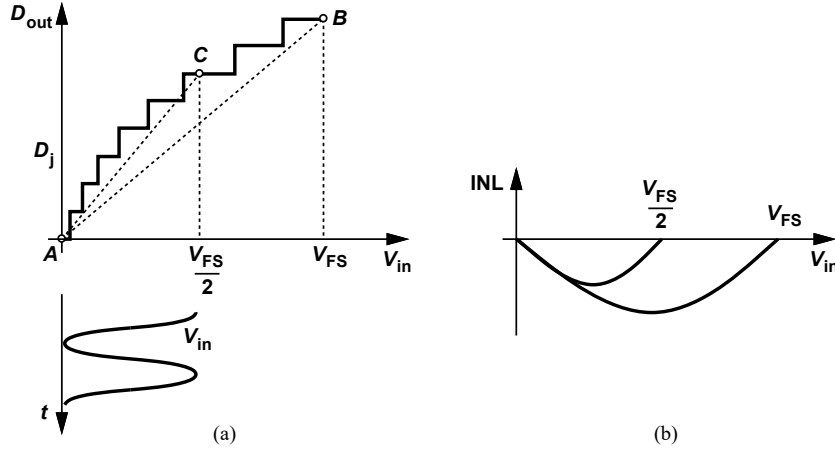


Figure 12.16 (a) An ADC sensing a half-scale input, and (b) reduction of INL.

profiles depicted in Fig. 12.16(b). Thus, the nonlinearity falls for a reduced swing. Unfortunately, however, the signal-to-noise ratio also drops, as predicted by Eq. (12.7). That is, the ADC faces a trade-off between linearity and SNR.

12.3.5 SNDR and ENOB

Recall from Section 12.2.2 that quantization produces noise, yielding an SNR equal to $(3/2)2^{2N}$. In the presence of nonlinearity, i.e., the curvature in Fig. 12.14(a), the ADC distorts the signal as well. We quantify this effect by means of the “signal-to-(noise+distortion)-ratio” (SNDR). For a sinusoidal input with a peak amplitude of V_0 , we have

$$\text{SNDR} = \frac{\frac{V_0^2}{2}}{P_{\text{noise}} + P_{\text{dist}}}, \quad (12.16)$$

where P_{noise} is the total noise power and P_{dist} the total distortion power. The former includes both quantization and electronic noise (e.g., kT/C noise). The latter is obtained by adding the powers of the harmonics that are present in the output. In this section, the term “harmonic” refers to those beyond the fundamental.

Example 12.7

Upon taking the fast Fourier transform (FFT) of an ADC’s output, a student obtains the spectrum shown in Fig. 12.17. Explain how the SNDR is computed from this plot.

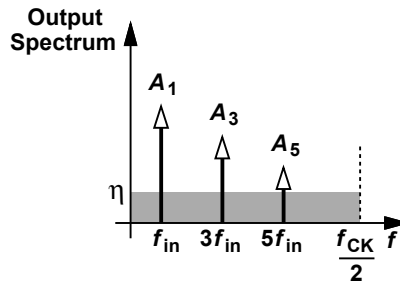


Figure 12.17 An ADC output spectrum.

Solution

We first exclude the impulses at integer multiples of f_{in} and integrate the noise power from $f = 0$ to

$f = f_{CK}/2$.⁷ This yields $P_{noise} = \eta(f_{CK}/2)$ in (12.16). Next, we add the harmonic powers as $P_{dist} = A_3^2/2 + A_5^2/2$, where A_j denotes the *peak* amplitude of harmonic number j . Finally, we substitute these values in (12.16) with $V_0^2/2 = A_1^2/2$.

Example 12.8

An ADC operates near the Nyquist rate, i.e., it senses an input $V_{in} = V_0 \cos \omega_{in} t$, where $\omega_{in} = 2\pi f_{in}$ and $f_{in} = f_{CK}/2 - \Delta f$. How do harmonics appear in the output spectrum?

Solution

Since the harmonics land beyond $f_{CK}/2$, the sampling operation in the ADC aliases them to the first Nyquist zone. Specifically, the second harmonic is convolved (mixed) with the clock frequency, yielding two components at $f_{CK} \pm 2f_{in}$ [Fig. 12.18(a)]. Of interest to us is $f_{CK} - 2f_{in} = 2\Delta f$. Figure 12.18(b) illustrates the overall effect.

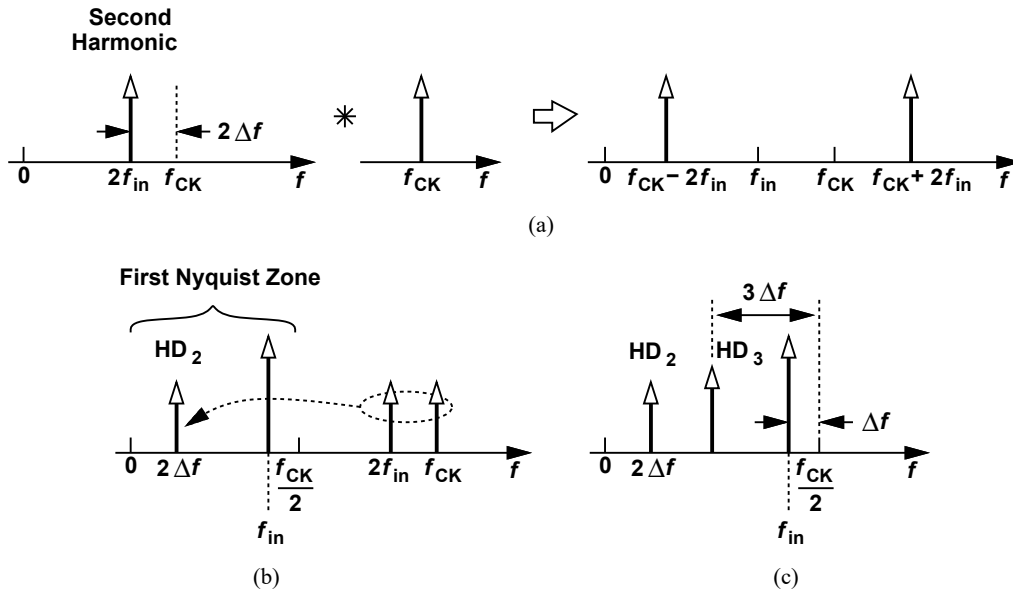


Figure 12.18 (a) Second harmonic generated in an ADC, (b) corresponding folded component, and (c) summary of folded harmonics.

The third input harmonic, given by $3f_{in} = 3f_{CK}/2 - 3\Delta f$, is sampled at a rate of f_{CK} , falling at $f_{CK} - (3f_{CK}/2 - 3\Delta f) = -f_{CK}/2 + 3\Delta f$. This negative value has a counterpart at $f_{CK}/2 - 3\Delta f$. Figure 12.18(c) summarizes the results.

In the presence of electronic noise and distortion, an N -bit ADC's SNDR falls short of the ideal SNR value of $(3/2)2^{2N}$. Equivalently, the ADC no longer provides a resolution of N bits. We can then define an "effective number of bits" (ENOB) so as to quantify the performance; we write (12.10) as

$$\text{SNDR} = 6.02\text{ENOB} + 1.76 \text{ dB} \quad (12.17)$$

⁷Called the "first Nyquist zone," this frequency range is the widest band in which the signal of interest resides.

and hence

$$\text{ENOB} = \frac{\text{SNDR} - 1.76}{6.02}. \quad (12.18)$$

For example, an ADC having a nominal resolution of 10 bits and exhibiting an SNDR of 58 dB provides $\text{ENOB} = 9.3$ bits.

Example 12.9

We evaluate the performance of an ADC by applying a sinusoidal input, gradually increasing its amplitude, and measuring the SNDR. Does the SNDR reach a maximum for an input swing equal to the full scale?

Solution

In typical designs, yes, it does. Equation (12.16) reveals two opposing mechanisms as V_0 increases: both the numerator and P_{dist} rise. Nonetheless, the numerator climbs more steeply up to a certain point. As P_{dist} exceeds P_{noise} , the growth of SNDR slows down.

12.3.6 SFDR

The “spurious-free dynamic range” (SFDR) of an ADC is defined as the ratio of the desired signal power to the highest unwanted component power (Fig. 12.19). Note that the SFDR does not account for noise.

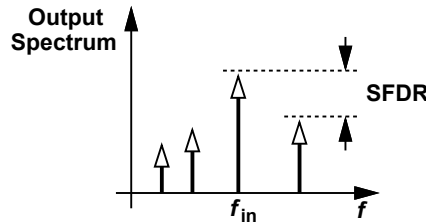


Figure 12.19 Definition of SFDR.

As a measure of ADC’s performance, the SFDR is less rigorous than the SNDR. The former can generally be improved by selecting a moderate or small input swing whereas the latter cannot. To understand this point, recall from Chapter 2 that the amplitude of the n -th harmonic grows in proportion to the n -th power of the input amplitude, V_0 . For example, suppose the third harmonic is the highest unwanted component in Fig. 12.19. If V_0 is halved, this harmonic falls by a factor of 8, raising the SFDR. One must then specify the input swing for which the SFDR is measured.

12.3.7 Static and Dynamic Performance

An N -bit ADC operating at a conversion rate of f_{CK} must ideally provide an SNR and SNDR of $(3/2)2^{2N}$ at all input frequencies below the Nyquist rate. In practice, however, the performance degrades as f_{in} increases. In fact, the SNDR and ENOB may fall considerably as f_{in} approaches $f_{CK}/2$. For this reason, we distinguish between “static” and “dynamic” performance. The former refers to the characteristics measured with a low input frequency, often simply a slow ramp. The latter signifies the ADC behavior with a full-scale sinusoidal input whose frequency is varied from low values to near $f_{CK}/2$.

12.3.8 Figure of Merit

The performance of ADCs is characterized by three metrics, namely, the resolution, the conversion rate, and the power consumption. We can then define a figure of merit (FOM) that encapsulates the trade-offs

among these parameters. Since we expect that doubling the resolution or speed requires doubling the power consumption, we write

$$\text{FOM}_1 = \frac{P}{f_{CK} \cdot 2^N}. \quad (12.19)$$

Note that P/f_{CK} in fact signifies the energy consumed by the ADC in one clock cycle. For this reason, we express FOM_1 in joules per conversion cycle. Of course, we wish to *minimize* FOM_1 .

FOM_1 , however, does not represent the *dynamic* performance of ADCs, failing to show how the ENOB falls as the analog input frequency approaches $f_{CK}/2$. We therefore replace 2^N with 2^{ENOB} and f_{CK} with twice the input frequency at which the ENOB is measured:

$$\text{FOM}_2 = \frac{P}{2f_{in} \cdot 2^{\text{ENOB}}}. \quad (12.20)$$

FOM_2 is expressed in joules per conversion cycle as well. Ideally, $f_{in} \approx f_{CK}/2$ and $\text{ENOB} = N$. But in practice, the minimum FOM_2 may occur below the Nyquist rate, depending on how steeply the ENOB falls versus f_{in} . Today's ADCs achieve FOM_2 values as low as a few tens of femtojoules per conversion cycle.

12.4 Nyquist-Rate ADC Architectures

Extensive work on ADCs has resulted in numerous architectures for Nyquist-rate operation. Depending on the resolution and the speed, we prefer some over others. Illustrated in Fig. 12.20 are some examples. Flash

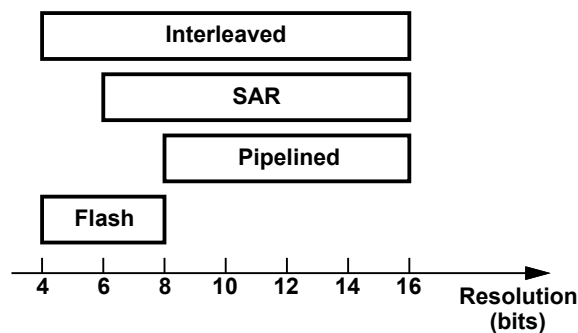


Figure 12.20 Most popular Nyquist-rate ADC architectures.

topologies (Chapter 13) are suited for roughly $4 \leq N \leq 8$. A flash variant employing “folding” and “interpolation” proves useful for $6 \leq N \leq 10$ but is less common. We describe a simple, efficient interpolation scheme in the next chapter. Pipelined ADCs (Chapter 17) find application for $8 \leq N \leq 16$, and successive-approximation (SAR⁸) architectures (Chapter 14) for $6 \leq N \leq 16$. Among these, SAR ADCs are the most popular today for their potentially low power consumption.

The concept of “interleaving” can be applied to any ADC architecture to achieve a higher conversion rate and/or ease the speed-power trade-off (Chapter 20).

⁸SAR traditionally stands for “successive-approximation register,” but we drop the third word here.

Problems

- 12.1. If $V_{in} = V_0 \cos \omega_0 t + V_0$ in Fig. 12.2 and $2V_0 \approx V_{FS}$, sketch the analog equivalent of the quantizer's output as a function of time.
- 12.2. We double the input voltage range of a 10-bit quantizer. Explain what happens to the Q-noise power and the maximum SNR.
- 12.3. A 12-bit quantizer operates with a full-scale voltage of 1 V.
 - (a) Determine the Q-noise power.
 - (b) Determine the output SNR for a full-scale sinusoidal input.
 - (c) How much does the SNR degrade if the sinusoidal amplitude is halved?
- 12.4. If we double the capacitor value in Fig. 12.9(a), what happens to the offset and the gain error?
- 12.5. Consider the 3-bit quantizer in Fig. 12.2(b). Suppose $DNL = 0.2$ LSB for all of the input codes. Sketch the resulting input-output characteristic.
- 12.6. Sketch the INL profile for the previous problem.
- 12.7. In a hypothetical 3-bit ADC, we have $DNL_j = 0.1j$ for $j = 1, \dots, 8$. Compute the gain error and the INL profile.
- 12.8. An ADC sensing an input sinusoid exhibits second and third harmonics that are respectively 40 dB and 50 dB below the fundamental. Determine the SNDR. [Hint: The power of the fundamental is expressed as $A^2/2$, where A is the peak input amplitude, and that of the second harmonic as $(A^2/2)/(10^{-40/10})$.]
- 12.9. An ADC operates near the Nyquist rate with an input of the form $V_{in} = V_0 \cos \omega_{in} t$. The circuit generates the fifth harmonic of the input. Plot the output spectrum.
- 12.10. A 3-bit ADC exhibits an INL profile given by $INL_j = j^2 - 8j$. Sketch the DNL profile.
- 12.11. An ADC exhibits an input-output characteristic given by $y(t) = \alpha_1 x(t) + \alpha_3 x^3(t)$. The input range is from zero to x_{max} . (a) Determine the INL profile and its maximum value. (b) If $x(t) = A \cos \omega_{in} t$, calculate the peak amplitude of the third harmonic and express it in terms of the maximum INL.
- 12.12. In the previous problem, we halve the input amplitude while the input range is still equal to x_{max} . What happens to the peak amplitude of the third harmonic and the Q-noise power?

Flash ADCs

As explained in Chapter 1, analog-to-digital conversion involves two operations, namely, sampling and quantization. The simplest and fastest ADC architecture is the “flash” topology. It can serve as a stand-alone converter or as part of a more complex ADC.

In this chapter, we study the flash structure in detail, delving into its nonidealities and performance trade-offs. We also examine some variants such as interpolating architectures. We should remark that, even if a design does not opt for a simple flash ADC, the knowledge gained here proves useful in the study of other architectures as well. The reader is encouraged to review Chapters 5 and 9 before proceeding here.

13.1 Basic Flash ADC

How do we design a circuit that performs quantization? We require two utilities: a means of generating the decision thresholds, and a means of comparing the input to these thresholds. We know from Chapter 9 that a resistor ladder can act as the former and from Chapter 5 that comparators can serve as the latter. From these observations emerges the flash ADC in Fig. 13.1(a). For a resolution of N bits, the converter employs $m = 2^N$ equal resistors and $m - 1$ identical comparators. We denote the thresholds (also called the references) by

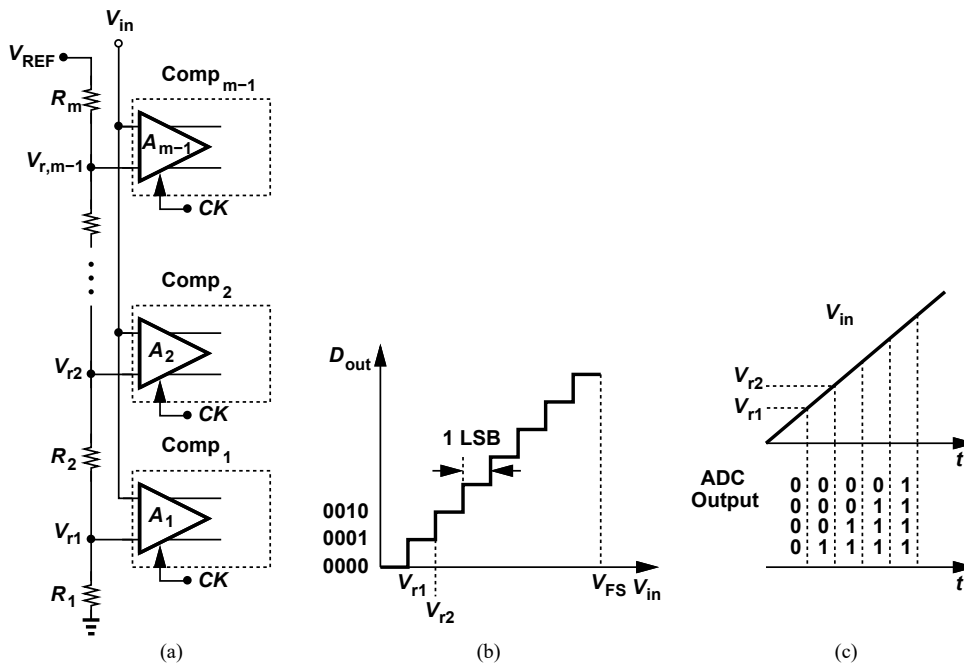


Figure 13.1 (a) Simple flash ADC, (b) its input-output characteristic, and (c) its response to an input ramp.

V_{r1} , V_{r2} , etc., and the full scale by V_{REF} . When CK rises (or falls), all of the comparators are activated simultaneously, sensing the *difference* between V_{in} and their corresponding decision thresholds.

Let us analyze the circuit by studying the comparator outputs as a function of V_{in} . For $V_{in} < V_{r1} = V_{REF}/m$, all of the comparators sense a negative input difference. Thus, their outputs are equal to a logical ZERO. If $V_{r1} < V_{in} < V_{r2}$, only comparator Comp_1 senses a positive difference and generates a logical ONE. For $V_{r,j} < V_{in} < V_{r,j+1}$, the bottom j comparators deliver ONES. That is, the comparator outputs form a thermometer code (Chapter 8). If these outputs are converted to a binary code, D_{out} , we can construct the characteristic shown in Fig. 13.1(b).

The flash ADC operation can also be observed in the time domain by selecting a slow ramp for V_{in} in Fig. 13.1(a) and monitoring the output. As depicted in Fig. 13.1(c), the thermometer code rises uniformly with time. When V_{in} is close to V_{rj} , we say Comp_j makes a “critical” decision.

The principal advantage of the flash ADC is its speed: all of the comparators are activated on the same edge of the clock. That is, the converter performs quantization in one clock cycle. As seen in the following chapters, other ADC architectures take longer to digitize their input.

The flash ADC nonetheless suffers from a number of drawbacks that are studied extensively in this chapter. Most notable is the exponential growth of its “cost” as a function of the resolution. The cost includes power consumption, input capacitance, comparator kickback noise (Chapter 6), comparator offset cancellation circuits, chip area, and difficulties in distributing signals. For this reason, the flash architecture is typically suited to resolutions no more than approximately 8 bits.

Example 13.1

Determine the full-scale value of the input in Fig. 13.1(b).

Solution

We note that V_{FS} denotes the greatest input level that the quantizer can sense properly. We define V_{FS} as 1 analog LSB above the last code transition, i.e., $V_{FS} = V_{REF}$. The flash ADC therefore provides a full scale equal to V_{REF} . If V_{in} exceeds this value, the ADC output remains at $11 \cdots 1$. In such a situation, we say the input experiences “overrange” and the output “saturates.”

The flash ADC of Fig. 13.1(a) merits several remarks. First, depending on the environment in which the converter is used, its thermometer-code output may need further processing. Specifically, if the circuit serves as a stand-alone ADC, the output must be converted to a binary or Gray code (Section 13.3). But if the flash stage operates as part of a more complex ADC, its thermometer-code output can be directly utilized (Chapter 19).

Second, it is possible that the comparators in Fig. 13.1(a) cannot accommodate an input common-mode level near zero. For example, A_1 and A_2 simply fail for $V_{in} \approx 0$ if they incorporate NMOS differential pairs at their inputs. For this reason, the bottom end of the ladder must be tied to a suitable voltage, e.g., $V_B = 0.5$ V, rather than to zero. The ADC input range is then equal to $V_{REF} - V_B$. This CM limitation can be lifted through the use of capacitive coupling (Section 13.6).

Example 13.2

Figure 13.2(a) depicts the simplified front end of a flash ADC, where each comparator incorporates a differential pair. (For simplicity, the pairs are not clocked for now.) Sketch the differential pair output waveforms if V_{in} is a ramp.

Solution

If V_{in} is well below V_{r1} , M_1 is off and M_2 is on, producing $V_{X1} = V_{DD}$ and $V_{Y1} = V_{DD} - R_D I_{SS}$. The same holds for V_{X2} , V_{Y2} , V_{X3} , and V_{Y3} . As V_{in} rises and approaches V_{r1} , V_{X1} decreases, and V_{Y1} increases. When $V_{in} = V_{r1}$, V_{X1} and V_{Y1} cross each other. Similarly, V_{X2} and V_{Y2} cross when $V_{in} = V_{r2}$, etc. [Fig. 13.2(b)].

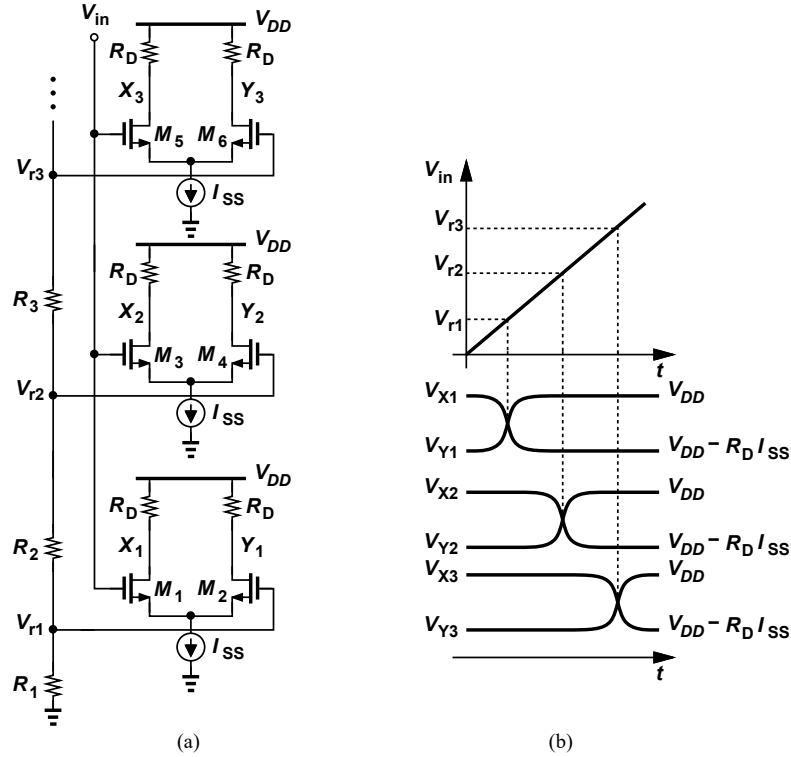


Figure 13.2 (a) Front end of a flash converter, and (b) its input and output waveforms.

13.2 Sampling in Flash ADCs

The architecture of Fig. 13.1(a) does not contain an explicit front-end sampling circuit. Can the comparators perform sampling in addition to quantization? Yes, they can—at least in principle. We elaborate on this point in this section.

Let us suppose, for simplicity, that the comparators in Fig. 13.1(a) sense their input difference for half of the clock cycle and regenerate for the other half. The current-steering topology in Chapter 5 exemplifies such an arrangement. We visualize the comparator as shown in Fig. 13.3(a), where A_0 represents the input differential pair and is enabled when CK is high. Turning to Fig. 13.3(b), we assume a full-scale sinusoid is applied to the ADC input and examine the outputs of the differential pairs while they are enabled. From Example 13.2, we recognize that each pair produces $V_X = V_Y$ when its input difference is zero. Moreover, $V_{X,j}$ falls and $V_{Y,j}$ rises before V_{in} reaches $V_{r,j}$, continuing their trajectories after V_{in} crosses $V_{r,j}$ and eventually assuming a low level and a high level, respectively. We say the output of A_j “tracks” the input when V_{in} is in the vicinity of $V_{r,j}$.

At a given point in time, most of the differential pairs in Fig. 13.3(b) generate a large $V_X - V_Y$, except for at least one whose reference is close to V_{in} . For example, at $t = t_a$, A_3 operates as a linear amplifier, delivering $V_{X3} - V_{Y3} \propto V_{in} - V_{r3}$.

Now we ask, what happens if a clock transition occurs at $t = t_0$? The differential pairs are disabled and the latches are enabled. Thus, $V_{X,j} - V_{Y,j}$ is “sampled” (and regeneratively amplified) by the latch following A_j . In other words, the comparators collectively act as a sampling circuit, with each comparator responsible for only a narrow range of the input swing. For example, if the converter is clocked at $t = t_0$, then the two bottommost comparators deliver ONES and the rest, ZEROS. We also say the comparators serve as a

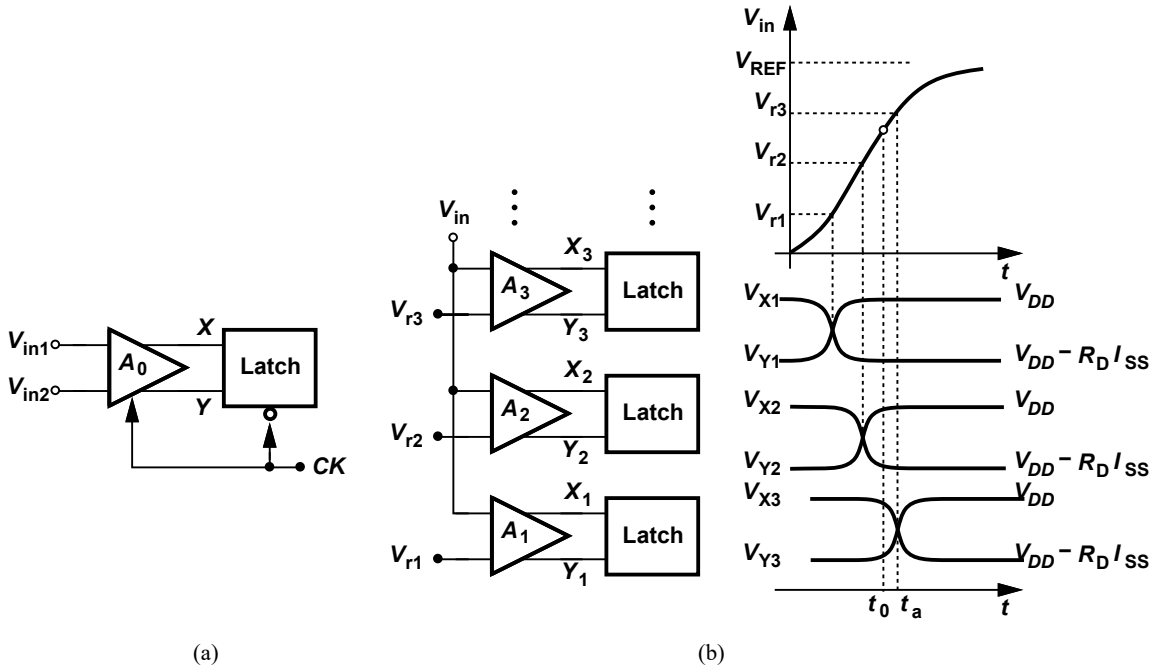


Figure 13.3 (a) A comparator consisting of a preamplifier and a latch, and (b) a flash ADC using this comparator and its waveforms.

“distributed” sampler because each one tracks only a small part of the input voltage linearly.

This analysis suggests that the flash ADC can operate without a front-end sample-and-hold amplifier (SHA), a great advantage in view of the difficulties in designing sampling circuits that must drive a large capacitance. However, as explained in Section 13.4.5, “SHA-less” flash converters suffer from certain issues at high frequencies.

13.3 Thermometer-to-Binary Conversion

The thermometric output of the flash ADC shown in Fig. 13.3(b) must be converted to a more compact form—typically, binary data—before it undergoes digital processing. To avoid confusion, we refer to the individual values in a thermometer code as “entries” rather than as bits.¹ In this section, we study several decoder examples that generate binary outputs. In some cases, we prefer conversion to the Gray code, which we defer to Section 13.4.3. Our principal challenge relates to the physical distance between the comparator outputs and hence the long interconnects that the decoder would require. In our studies, we use the terms “encoder” and “decoder” interchangeably.

13.3.1 ROM Decoder

Thermometer-to-binary conversion can be realized in two steps: we first transform the comparator outputs to a 1-of- n code (Chapter 8) and then convert the result to binary form. Illustrated in Fig. 13.4(a) is the first step: the AND gates sense the outputs of adjacent comparators, thus detecting the transition of the thermometer code from 1 to 0. Consequently, for the j -th entry in the 1-of- n code to be a logical ONE, the “height” of the thermometer code is equal to j . The key advantage of this scheme is that it maintains short interconnects between the comparator and the AND gates.

¹The term “bit” stands for “binary digit.”

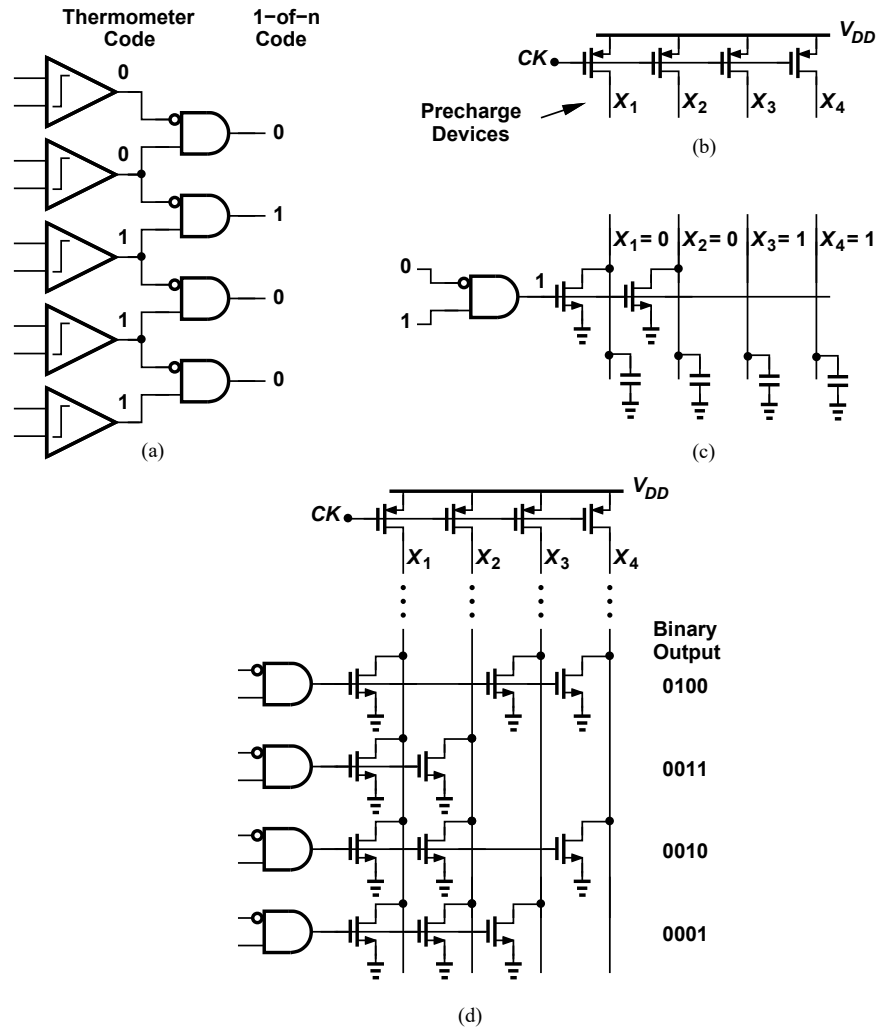


Figure 13.4 (a) Thermometer to 1-of- n decoder, (b) clocked precharge devices, (c) one row of a ROM sensing a 1-of- n code, and (d) ROM decoder.

In the second step, we must generate a binary word that represents the 1-of- n code value. Since only one entry in this code is high, it must create a unique binary output by itself. Suppose, for example, that an output equal to 0011 is required. As shown in Fig. 13.4(b), we create four lines and first precharge them to V_{DD} so that they begin with a logical value of ONE. We then turn off the precharge devices and allow the 1-of- n code to discharge the lines that must have a ZERO logical value [Fig. 13.4(c)].

Let us extend the foregoing development by creating more rows that are controlled by the 1-of- n code. Depicted in Fig. 13.4(d) is an array of NMOS transistors that delivers the necessary binary output when the 1-of- n code activates one row. This array exemplifies a read-only memory (ROM).

In summary, this type of decoder first precharges the output lines to a high value and then produces the proper binary word as the thermometer code travels through the AND gates and activates one row of the ROM. But how do we ensure that the PMOS devices in Fig. 13.4(d) do not “fight” the NMOS pull-down transistors? This point demands that all of the 1-of- n code entries be at zero when the PMOS devices precharge the output lines. We can investigate whether the StrongArm comparator (Chapter 5) satisfies this condition. As shown in Fig. 13.5, when CK is low, all comparator outputs are high, indeed maintaining all of the 1-of- n code entries at zero. The ROM thus precharges properly. The above observation reveals that we must omit the other stages prescribed to follow the StrongArm comparator in Chapter 5, namely, the inverters and the RS

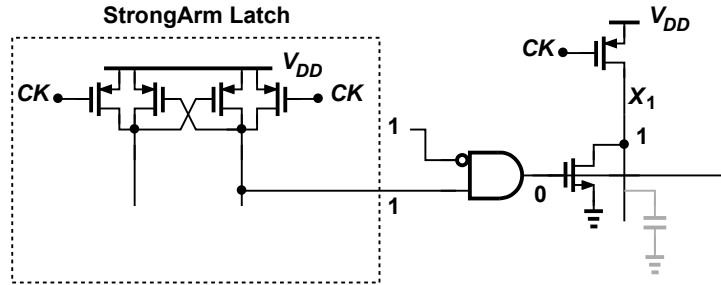


Figure 13.5 Interface between the StrongArm latch and a ROM decoder.

latch. This is, in fact, an advantage of the ROM decoder. Note, however, that the ROM outputs are invalid during the precharge mode, necessitating that the ROM be followed by RS latches.

The principal issue facing this decoder topology is the slow response of the output lines. We recognize that each vertical line in Fig. 13.4(d) must travel across the entire flash ADC, incurring substantial capacitance. Moreover, the transistors tied to these lines contribute their own capacitances. The fall time is particularly long because, in the worst case, a single NMOS transistor must discharge this large output capacitance.

Can we employ a ROM decoder if a flash ADC incorporates current-steering comparators? Recall from Chapter 5 that such topologies do not have a reset or precharge phase, thus providing no “predictable” output. If preceding the decoder in Fig. 13.4(d), current-steering comparators always generate a thermometer code and activate one row of the ROM. That is, the PMOS precharge devices must fight some NMOS pull-down transistors. For this reason, ROM decoders are rarely used with current-steering comparators.

13.3.2 MUX Decoders

In this section, we employ multiplexers (MUXes) to realize a decoder. A 2-to-1 MUX passes either of its inputs to the output based on the logical value of a select command [Fig. 13.6(a)]. A simple implementation using MOS switches is shown in Fig. 13.6(b).

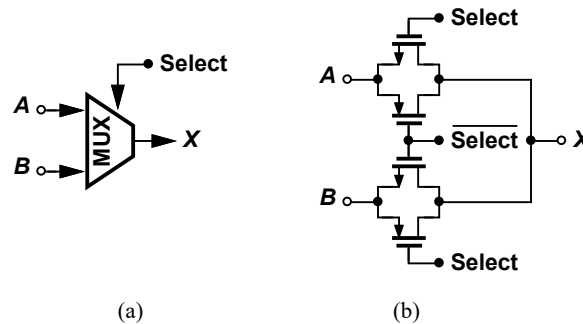


Figure 13.6 (a) Simple MUX, and (b) its implementation.

We can view the decoding task as identifying the topmost ONE entry in the thermometer code. We then surmise that a “binary search” method can determine this entry. Shown in Fig. 13.7(a) is an example. We first examine the fourth entry from the bottom for a 3-bit flash ADC, which represents the MSB. Depending on whether the MSB is low or high, we must examine the bottom or top half of the thermometer code, respectively, for computing the next MSB (denoted by MSB_{-1}). This procedure exemplifies binary search. For example, if $MSB = 0$, we must next sense the second entry from the bottom (T_2) as MSB_{-1} ; if $MSB = 1$, then the sixth entry gives MSB_{-1} .

These thoughts lead to the arrangement depicted in Fig. 13.7(b) [1]. Here, the fourth entry of thermometer code is sensed as the MSB and also utilized to control a 2-to-1 multiplexer. Based on the value of the MSB, the MUX passes the second or the sixth entry as MSB_{-1} .

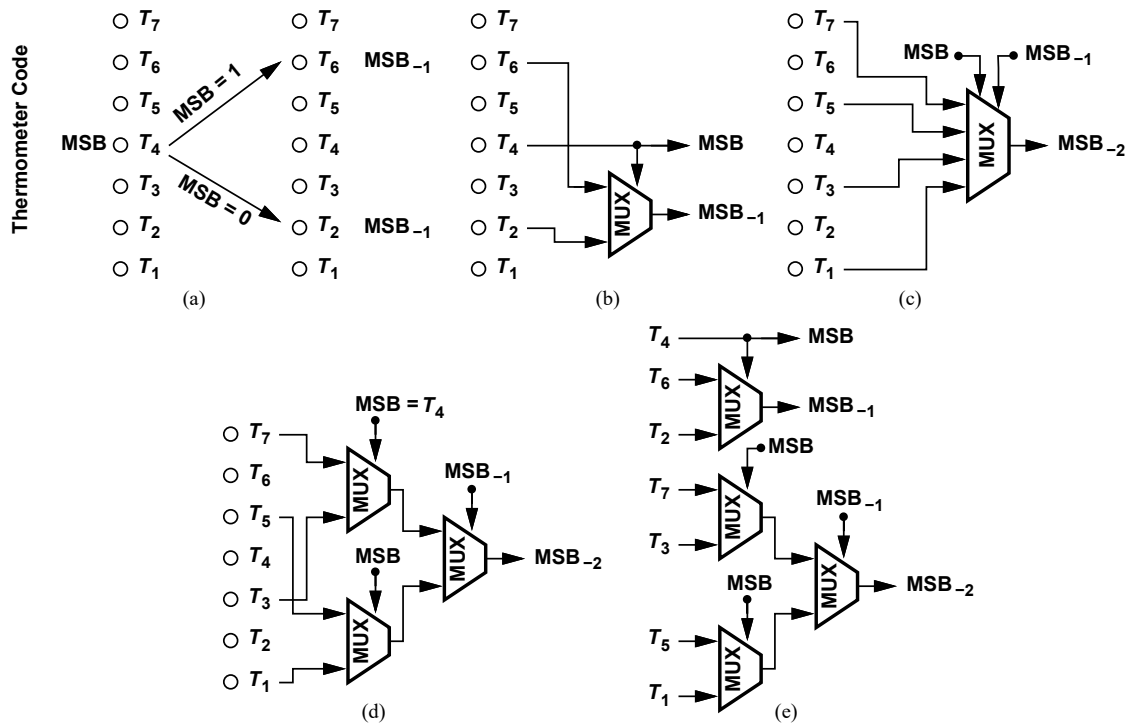


Figure 13.7 (a) Simple binary search in a thermometer code, (b) realization using a MUX, (c) generation of next MSB, (d) alternative implementation, and (e) complete design.

How do we compute MSB_{-2} ? Following our binary search method, we recognize that the values of MSB and MSB_{-1} can direct us to the proper subsection of the thermometer code. Specifically, Fig. 13.7(a) reveals that, if $MSB = 0$ and $MSB_{-1} = 1$, then we must sense the third entry in the thermometer code as MSB_{-2} . Similarly, if $MSB = 1$ and $MSB_{-1} = 1$, we must select the seventh entry. From these observations emerges the topology in Fig. 13.7(c), where a 4-to-1 MUX generates MSB_{-2} from the first, third, fifth, and seventh entries of the thermometer code. In this case, MSB_{-2} is, in fact, the LSB.

The 4-to-1 MUX in Fig. 13.7(c) can be realized as a tree of 2-to-1 MUXes [Fig. 13.7(d)], leading to the overall 3-bit decoder shown in Fig. 13.7(e) [1]. This approach entails two issues. First, the MSB line must travel across the entire flash ADC and control the first rank of multiplexers. In a 6-bit converter, for example, the MSB drives 32 MUXes, experiencing a significant load capacitance. Second, the thermometer-code entries also face long interconnects as they reach the 2-to-1 MUX stages; in our 6-bit example, the first and 33rd entries are multiplexed, and so are the second and 34th, etc. The latter issue is resolved in the U-shaped flash architecture (Section 13.6.2), but the former persists.

13.3.3 ONEs-Counter Decoders

Another approach to converting a thermometer code to binary form is to simply count the number of ONEs therein. The “counting” action can be performed by combinational logic. For this purpose, we must briefly review digital adder design.

The Half Adder Suppose two entries, A and B , can assume high or low levels. Whether A and B form a binary or a thermometer code is unimportant at this point. We wish to count the number of ONEs in AB and represent the result by another code, which must itself contain two bits as there are three different possibilities. We then arrive at the “half adder” (HA) illustrated in Fig. 13.8(a): from the truth table, we observe that the

binary word CS represents the number of ONEs in AB . We recognize that the sum $S = A \oplus B$ and the carry $C = AB$ [Fig. 13.8(b)].

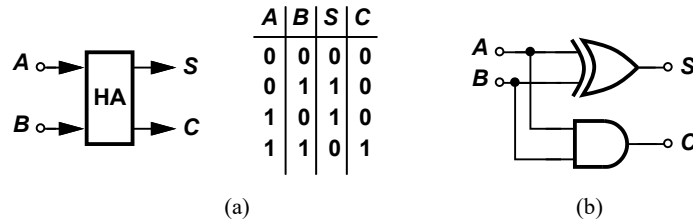


Figure 13.8 (a) Half adder, and (b) its implementation.

Example 13.3

How does the approach in Fig. 13.8(a) change if AB is a thermometer code?

Solution

In that case, AB can assume only three values, namely, 00, 10, and 11.² That is, S reduces to $A\overline{B}$ (why?).

The Full Adder Let us now extend the foregoing thoughts to three entries, A , B , and D , and decide how their ONEs can be counted. Shown in Fig. 13.9(a), the “full adder” (FA) assigns a high value to S when there is a single ONE or when there are three ONEs. The carry is high when there are two or three ONEs. The

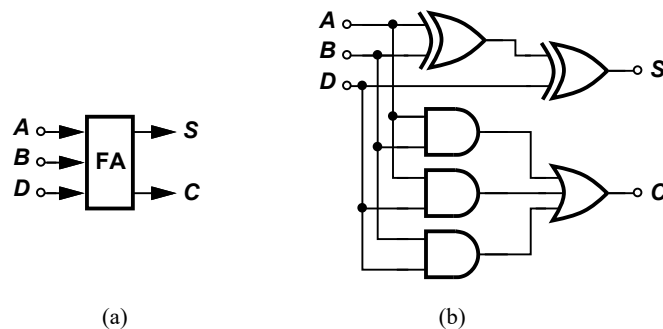


Figure 13.9 (a) Full adder, and (b) its implementation.

reader can show that

$$S = A \oplus B \oplus D, \quad (13.1)$$

$$C = AB + BD + AD. \quad (13.2)$$

The binary word CS represents the number of ONEs in ABD . An implementation is depicted in Fig. 13.9(b). In a manner similar to that in Example 13.3, we can simplify these expressions if ABD is thermometric and fills up according to 000, 100, 110, and 111:

$$S = A\overline{(B \oplus D)}, \quad (13.3)$$

$$C = B. \quad (13.4)$$

We next consider Fig. 13.10, where six thermometer-code entries are sensed by two full adders. We

²We assume A is the “lower” entry of the thermometer code.

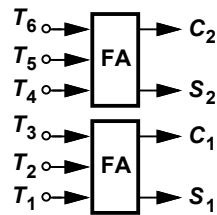


Figure 13.10 Use of adders to convert thermometer code.

observe that C_1S_1 and C_2S_2 yield binary values for the number of ONEs in $T_1T_2T_3$ and $T_4T_5T_6$, respectively. Thus, C_1S_1 and C_2S_2 must be simply summed so as to deliver a 3-bit binary value for the number of ONEs.

Addition of Binary Data As their names imply, half and full adders can also perform addition of binary words. Let us write

$$C_1S_1 \quad (13.5)$$

$$+ C_2S_2 \quad (13.6)$$

and note that S_1 and S_2 can be summed using a half adder. The result is of the form C_3S_3 . Next, the carry, C_3 , must be summed with C_1 and C_2 , requiring a full adder. We thus reach the arrangement shown in Fig. 13.11, where $D_1D_2D_3$ serves as the binary equivalent of the input thermometer code, $T_1 \cdots T_6$ [2].

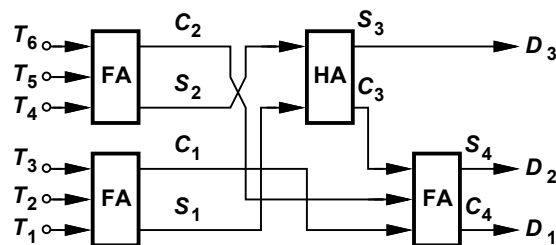


Figure 13.11 Thermometer-to-binary conversion by means of adders.

Example 13.4

How do we modify the decoder of Fig. 13.11 for a 3-bit flash ADC?

Solution

With an input of the form $T_1 \cdots T_7$, we must count the ONE in T_7 as well. This is readily accomplished by adding T_7 to S_1 and S_2 (Fig. 13.12). The circuit remains otherwise the same. The reader is encouraged to design the encoder for a 4-bit flash ADC.

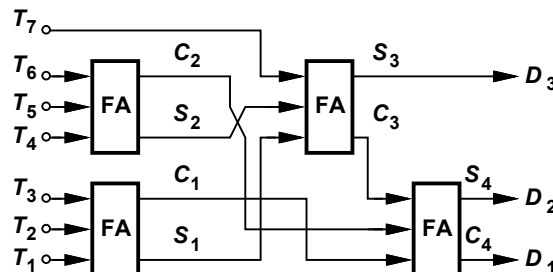


Figure 13.12 Thermometer-to-binary conversion by means of adders for a 3-bit flash ADC.

The principal advantage of the ONES-counter decoder is that it avoids long interconnects for the thermometer code lines. Also, as explained later in this chapter, this approach proves more robust in the presence of “bubbles.” However, as evident from Fig. 13.12, the first rank of full adders must drive long wires, suffering from a low speed. Also, the full adders require higher digital complexity than the other decoding schemes studied in the previous sections.

13.4 Nonidealities

The performance of flash ADCs is affected by various circuit nonidealities. In this section, we consider the offset, kickback noise, and metastability of the comparators, the problem of signal distribution, and the supply noise produced by the ADC. The reader is encouraged to review Chapter 12 first.

13.4.1 Comparator Offset

Recall from Chapter 5 that random asymmetries translate to an input-referred offset in comparators. That is, for a given comparator to make a correct decision, its input voltage difference must be sufficiently positive or negative. We wish to determine how comparator offsets affect the flash ADC characteristic.

Let us assume that only one comparator exhibits an offset equal to V_{OS} [Fig. 13.13(a)], yielding a decision threshold at $V_{in} = V_{r,j} + V_{OS}$ rather than at $V_{in} = V_{r,j}$. We denote this decision threshold by V_j and construct

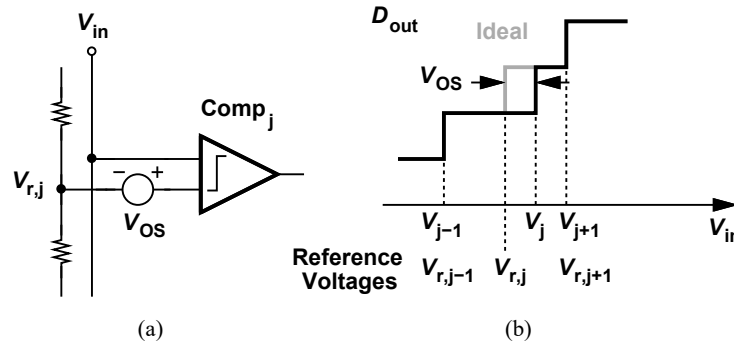


Figure 13.13 (a) A comparator with offset in a flash environment, and (b) resulting characteristic.

the input-output characteristic as shown in Fig. 13.13(b),³ where D_{out} represents the equivalent value of the overall output thermometer code. In the ideal case, the code transitions occur at $V_{in} = V_{r,j-1}$, $V_{r,j}$, $V_{r,j+1}$, etc. With V_{OS} present, the transition at $V_{r,j}$ is shifted to $V_{r,j} + V_{OS}$, introducing both DNL and INL. From our studies in Chapter 12, we note that $DNL = V_j - V_{j-1} - 1 \text{ LSB} = V_{OS}$ and $INL = V_{OS}$ in this particular case.⁴

Example 13.5

Does a missing code occur in Fig. 13.13(a) if V_{OS} approaches +1 LSB or −1 LSB?

Solution

If only one comparator exhibits offset, a missing code arises in both cases. Illustrated in Fig. 13.14 are the two situations.

³This notation should not be confusing: V_j indicates the input voltage at which a transition occurs whereas $V_{r,j}$ is the reference voltage generated by the ladder. In the absence of offsets, the two are equal.

⁴Recall that the INL is given by the horizontal difference between the actual characteristic and the straight line passing through the end points.

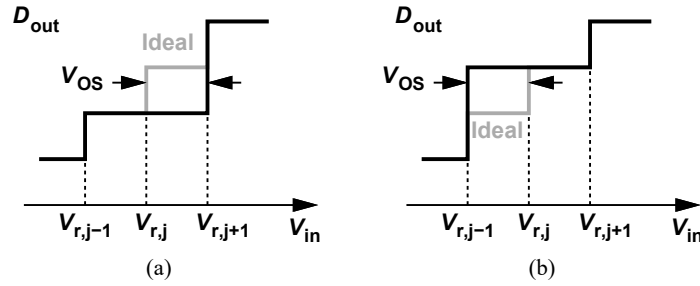


Figure 13.14 Missing code in two cases.

In practice, all of the comparators in a flash ADC suffer from offset. Thus, every two adjacent ones contribute DNL and INL. Depicted in Fig. 13.15 is an example where

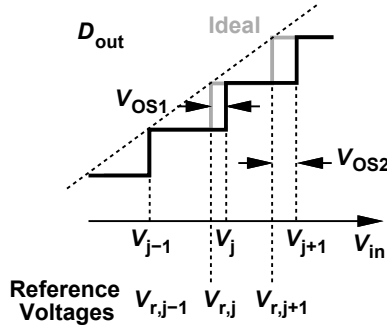


Figure 13.15 Effect of offset in two comparators.

$$DNL_j = V_j - V_{j-1} - 1 \text{ LSB} = V_{OS1} \quad (13.7)$$

$$DNL_{j+1} = V_{j+1} - V_j - 1 \text{ LSB} = V_{OS2} - V_{OS1}. \quad (13.8)$$

Since the offsets are random quantities, we express the standard deviation of the DNL as

$$\sigma_{DNL} = \sqrt{V_{OS1}^2 + V_{OS2}^2}. \quad (13.9)$$

The DNL is therefore equal to $\sqrt{2}$ times the offset of one comparator. We also observe that the INL is equal to V_{OS1} for the first transition and equal to V_{OS2} for the second, concluding that the tolerable offset is dictated by the DNL.

Bubbles in Thermometer Code We typically design the comparators for an offset less than 0.2 LSB, but, as a random quantity, it can exceed 1 LSB with a finite probability. While our foregoing analysis suggests that an offset slightly greater than 1 LSB yields a missing code, the actual consequence can be much more serious. Specifically, consider the flash stage depicted in Fig. 13.16, assuming that only comparator Comp_j exhibits $V_{OS} = 1.2 \text{ LSB}$. As a result, this comparator generates a low output if, for example, $V_{in} = V_{r,j} + 1.1 \text{ LSB}$, whereas Comp_{j+1} produces a high output. The thermometer code therefore suf-

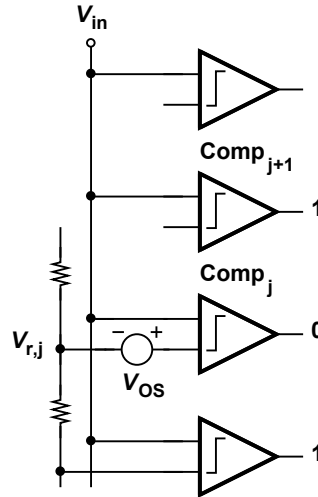


Figure 13.16 A bubble in a thermometer code.

fers from a “bubble,” i.e., a logical ZERO between two ONES.⁵ How much error does a bubble create? This depends on the type of decoder that processes the thermometer code and is studied in Section 13.5.

In the simple case studied above, one may say that the “correct” value of the code is obtained if the bubble is interpreted as a ONE.

13.4.2 Input Capacitance

The flash ADC input capacitance grows exponentially with the resolution, but in proportion to 2^{3N} rather than 2^N . To understand this point, suppose we wish to increase the resolution from N to $N + 1$ while the input voltage range is limited by the supply voltage and must remain unchanged. The analog LSB is therefore halved. In order to halve the comparator offset, V_{OS} , we note from Chapter 6 that it contains $\Delta\beta/\beta$ and ΔV_{TH} [Eq. (6.18)] and each of these is inversely proportional to $\sqrt{WL}$ [Eqs. (6.27) and (6.28), respectively]. We then conclude that WL and hence the comparator input capacitance must be *quadrupled*. Consequently, the overall ADC input capacitance rises by a factor of 8.

The large input capacitance demands that a buffer precede the ADC, but such a buffer consumes high power and introduces its own nonlinearity. For this reason, we typically employ comparator offset cancelation (Section 6.5) for resolutions of 4 bits and above so as to maintain a low input capacitance.

13.4.3 Comparator Metastability

We have seen in Chapter 6 that comparators experience metastability if they sense a small input difference, thus failing to generate a well-defined logical level at the end of the clock cycle. We quantify metastability by the error rate that it causes, which in turn depends on the comparator’s regeneration time constant, τ_{reg} , and the clock period, T_{CK} . The probability of metastable errors was found to be

$$P_{error} = \exp \frac{-T_{CK}}{2\tau_{reg}} \quad (13.10)$$

for a uniformly-distributed input signal if we allow $T_{CK}/2$ seconds for regeneration.

Shown in Fig. 13.17(a) is a flash ADC sensing $V_{in} \approx V_{r,j}$ and hence creating a metastable entry in the thermometer code. Since V_{in} is close to $V_{r,j}$, whether this state is considered a ONE or a ZERO is unimportant, but the difficulty is that the indeterminate logical value can lead to gross errors in the final digital output. As an example, suppose the ambiguous thermometer code in Fig. 13.17(a) is applied to the

⁵ As if a bubble appeared within the alcohol in a thermometer.

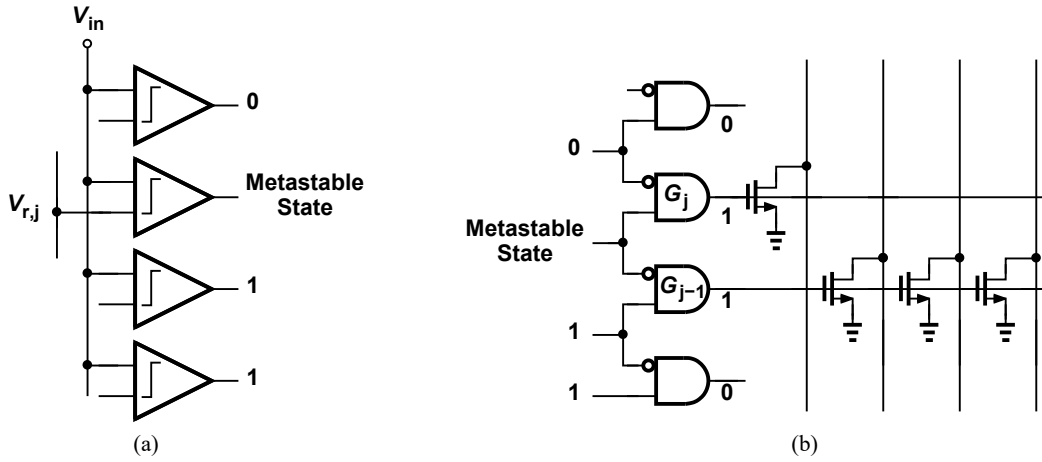


Figure 13.17 (a) A metastable output in a flash ADC, and (b) its effect on the ROM output.

ROM decoder studied in Section 13.3.1 [Fig. 13.17(b)]. Owing to their random device mismatches, gates G_{j-1} and G_j may interpret the metastable state *differently*. If G_{j-1} senses this state as a ZERO and G_j as a ONE, then *two* rows of the ROM are activated simultaneously, delivering, for example, an output equal to 0000 rather than 1000. We observe that analog input levels near mid-scale can incur a large error.

In order to reduce the probability of metastable errors in flash ADCs, we can pursue two techniques, namely, “pipelining” and proper decoder design. Let us delve into these concepts.

Pipelining Equation (13.10) suggests that P_{error} falls if the time available for regeneration increases, a possibility afforded by pipelining. Returning to the conceptual comparator structure of Fig. 13.3(a), we recognize that the latch’s output is metastable for a small $|V_{in1} - V_{in2}|$. We now add another latch, as shown in Fig. 13.18(a), and drive it by the opposite phase of the clock. When CK is low (before $t = t_1$), A_0 amplifies

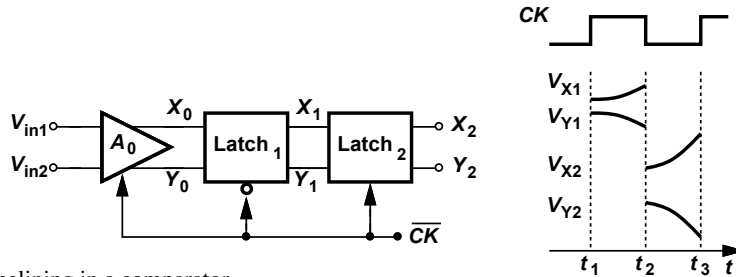


Figure 13.18 Pipelining in a comparator.

$V_{in1} - V_{in2}$, generating $V_{X0} - V_{Y0} = V_0$, Latch₁ holds the previous value, and Latch₂ senses $V_{X1} - V_{Y1}$. At $t = t_1$, CK rises, and Latch₁ begins to amplify regeneratively while Latch₂ stores the previous value. By $t = t_2$,

$$V_{X1} - V_{Y1} = V_0 \exp \frac{T_{CK}}{2\tau_{reg}}. \quad (13.11)$$

Then, CK goes low, and Latch₂ begins to amplify this voltage for another $T_{CK}/2$ seconds, delivering

$$V_{X2} - V_{Y2} = V_0 \exp \frac{T_{CK}}{2\tau_{reg}} \exp \frac{T_{CK}}{2\tau_{reg}} \quad (13.12)$$

$$= V_0 \exp \frac{T_{CK}}{\tau_{reg}}, \quad (13.13)$$

where the two latches are assumed to have the same τ_{reg} . We say the two latches pipeline the digital signal, providing twice the time available for regeneration. As a result, the error rate falls considerably.

Example 13.6

Suppose P_{error} in (13.10) is equal to 10^{-7} . By what factor does it drop if pipelining doubles the regeneration time?

Solution

We write (13.10) as

$$\exp \frac{T_{CK}}{2\tau_{reg}} = 10^{-7}, \quad (13.14)$$

where β is a proportionality factor, typically in the range of 0.2 to 2. It follows that

$$\frac{T_{CK}}{2} = \tau_{reg} \ln 10^{-7}. \quad (13.15)$$

We now double this result and compute the corresponding error rate:

$$P_{error} = \beta \exp \frac{2\tau_{reg} \ln(10^{-7})}{\tau_{reg}} \quad (13.16)$$

$$= 10^{-14}. \quad (13.17)$$

We observe that P_{error} decreases dramatically.

It is important to note that these results are valid only if the secondary latch in Fig. 13.18 employs *regenerative* amplification, as practiced in current-steering topologies, for example. Latches without positive feedback, on the other hand, simply provide a “linear” gain, reducing P_{error} proportionally rather than exponentially. Figure 13.19 shows such a latch topology.

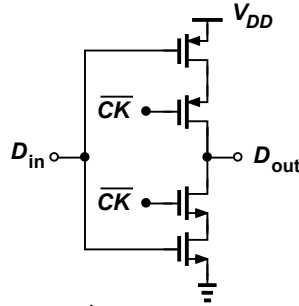


Figure 13.19 Example of a latch without regeneration.

Pipelining entails two issues: the secondary latches consume power and introduce additional capacitance in the clock path. Can we employ pipelining *within* the ROM decoder of Fig. 13.4(d)? If the goal is to lower metastability errors, then no, we cannot. For example, interposing latches between the AND gates and the ROM does not correct the 1-of- n code (why?). Similarly, pipelining after the ROM has no effect on this error.

Gray Coding We ask whether the decoder can be so designed as to incur only a small error in the presence of metastability. Fortunately, this is possible through the use of Gray coding—rather than binary coding. Recall from basic logic design that, in the Gray code, only one bit changes as the equivalent decimal value increments by 1. Depicted in Fig. 13.20(a) are the thermometer and Gray codes for a 3-bit system, yielding

$$G_1 = T_1 \overline{T_3} + T_5 \overline{T_7}, \quad (13.18)$$

$$G_2 = T_2 \overline{T_6}, \quad (13.19)$$

$$G_3 = T_4. \quad (13.20)$$

The logical implementation in Fig. 13.20(b) reveals an important property of the Gray code: the signals

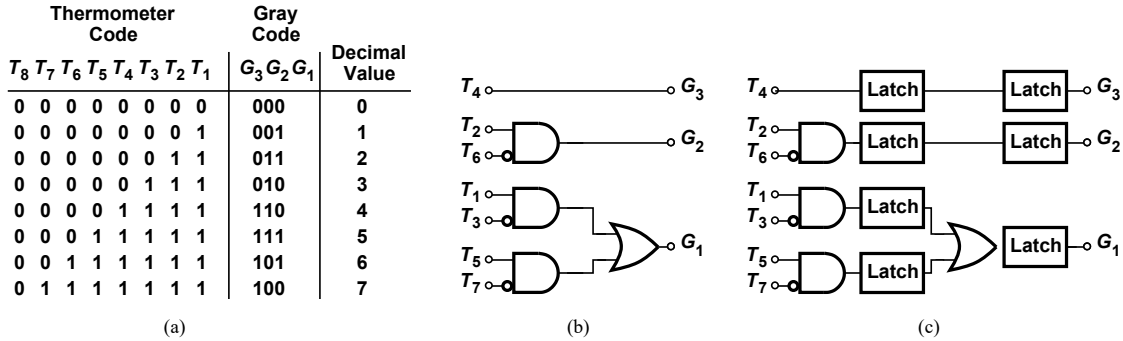


Figure 13.20 (a) Thermometer to Gray code conversion, (b) its implementation, and (c) addition of pipelining.

propagating through the circuit do not “split,” i.e., each drives a single gate. This means that a metastable state is not interpreted differently by two gates, a key difference with respect to the situation in Fig. 13.17(b).

Example 13.7

How does the logic of Fig. 13.20(b) respond to the metastable thermometer entry?

Solution

Suppose $T_1 = \dots = T_j = 1$ and T_{j+1} is metastable. If the gate sensing this state interprets it as a ONE, then $G_3 G_2 G_1$ represents $T_1 = \dots = T_{j+1} = 1$ and $T_{j+2} = \dots = 0$. Similarly, if the state is interpreted as a ZERO, $G_3 G_2 G_1$ is equivalent to $T_1 = \dots = T_j = 1$ and $T_{j+1} = \dots = 0$. The two results differ by 1 LSB, demonstrating the robustness of Gray coding.

Another benefit of Gray coding is its ability to reduce metastability by pipelining [Fig. 13.20(c)], with the number of latches dropping by a factor of 2 from one stage to the next. This is again possible because the signals do not split.

Other decoders, e.g., the MUX and ONES-counter topologies, exhibit different responses to metastability. Detailed transistor-level simulations are necessary to quantify their behavior.

13.4.4 Kickback Noise

As explained in Chapter 5, comparators draw transient currents from their inputs, disturbing the tap voltages generated by the resistor ladder [Fig. 13.21(a)]. This phenomenon can corrupt the present sample and/or persist long enough to affect the next sample. We formulate the disturbance for a ladder with m segments. In a well-designed flash ADC, the total kickback injection onto the ladder should be so small that the resulting corruption remains well below 1 LSB. We approximate the injected components by equal current sources [Fig. 13.21(b)] because, in a good design, the impedance of the injecting capacitors is much higher than the ladder’s Thevenin resistance. Let us compute the voltage perturbation at tap number j by finding the Norton equivalents of the networks within the dashed boxes, assuming $I_1 = \dots = I_{m-1} = I_u$. For that on the left, the Norton resistance is equal to jR_u . The Norton (short-circuit) current is obtained by summing the contributions of $I_1, \dots, I_{j-1}$ as

$$I_{Nort} = I_1 \frac{R_u}{jR_u} + I_2 \frac{2R_u}{jR_u} + \dots + I_{j-1} \frac{(j-1)R_u}{jR_u} = \frac{j-1}{2} I_u. \quad (13.21)$$

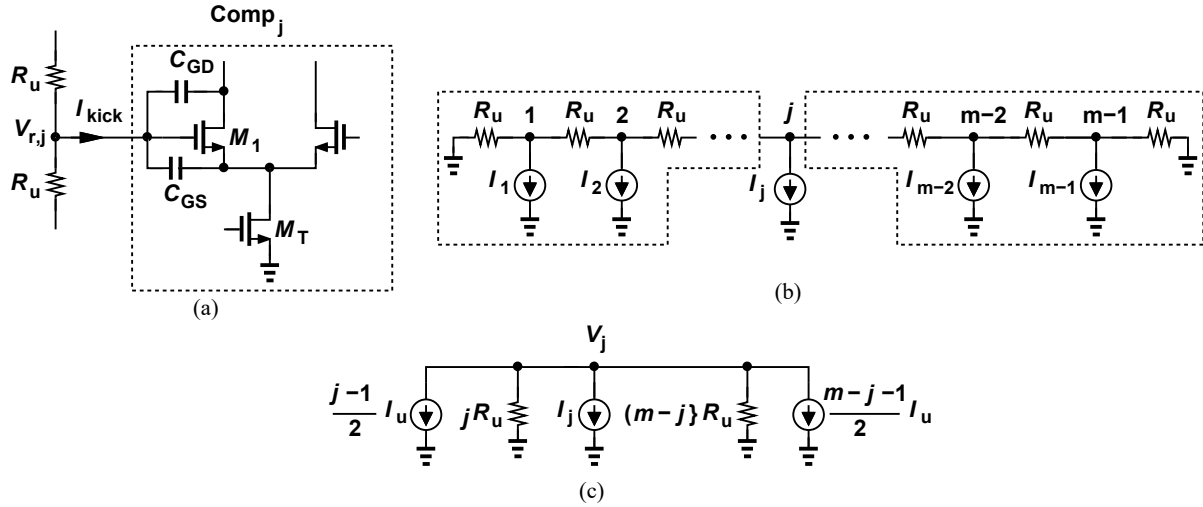


Figure 13.21 (a) Kickback noise in a flash ADC, (b) equivalent circuit, and (c) simplified structure.

Similarly, for the dashed box on the right, we have a Norton resistance of $(m-j)R_u$ and a Norton current of $[(m-j-1)/2]I_u$, arriving at Fig. 13.21(c). It follows that

$$V_j = \frac{(m-j)j}{2} I_u R_u, \quad (13.22)$$

reaching a maximum of

$$V_{\max} = \frac{m^2}{8} I_u R_u \quad (13.23)$$

at the ladder's midpoint. For a given comparator design, this error is minimized by reducing R_u and hence increasing the power consumed by the ladder.

Example 13.8

We wish to increase the speed of a flash ADC. Explain how the ladder's power consumption changes.

Solution

Consider the two cases depicted in Fig. 13.22. If the conversion rate is raised, the kickback noise has less time to decay, introducing a greater error in the next sample. Thus, R_u must be reduced, leading to higher power consumption.

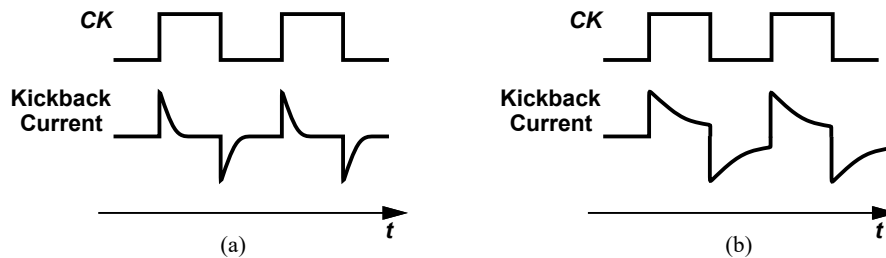


Figure 13.22 Effect of kickback noise with (a) fast, or (b) slow decay.

Kickback noise becomes particularly serious for resolutions above 4 bits as the error in Eq. (13.23) rises sharply with m . For example, a 6-bit ADC incorporating the StrongArm design of Chapter 7 and $R_u = 10\ \Omega$ incurs a peak disturbance of $(64^2/8)(200\ \mu\text{A})(10\ \Omega) = 1024\ \text{mV}$!

We describe two methods of alleviating this issue. First, the comparator can be preceded by a simple continuous-time (unlocked) differential pair. Illustrated in Fig. 13.23(a), the idea is to create some “reverse

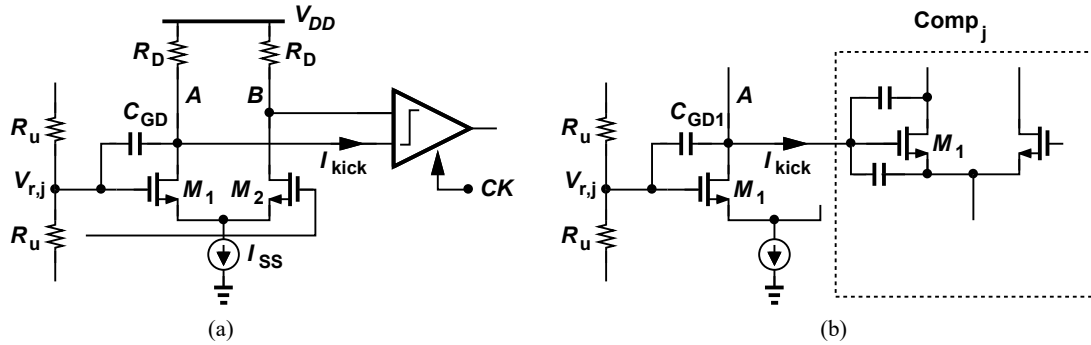


Figure 13.23 (a) Comparator preceded by a differential pair to reduce kickback noise, and (b) kickback noise path.

isolation” so that I_{kick} is heavily attenuated before reaching the ladder. We observe that I_{kick} must now flow through C_{GD1} , which, from the equivalent shown in Fig. 13.23(b), appears in series with the comparator’s capacitances. Of course, our assumption is that the perturbation in $V_A - V_B$ due to the kickback current does not affect the comparator’s present or next decision.

The principal drawback of using a differential pair is its power consumption, which can, in fact, be *greater* than the comparator’s. This is because nodes A and B in Fig. 13.23(a) must exhibit sufficient bandwidth, thereby requiring that R_D be small enough and I_{SS} be large enough (to provide a voltage gain of 2 or 3).

The second approach to lowering the effect of kickback noise is to use capacitors to bypass one or more taps of the ladder to a constant potential, e.g., the ground. As shown in Fig. 13.24(a), I_{kick} flows primarily through C_1 , disturbing $V_{r,j}$ to a lesser extent. If the ladder’s midpoint is successfully held at ac ground, then

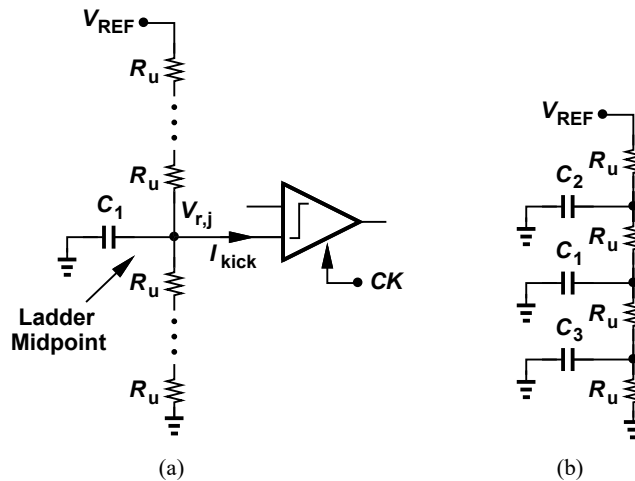


Figure 13.24 Addition of (a) one, or (b) multiple capacitors to the ladder to reduce the effect of kickback noise.

the largest voltage perturbation occurs at the midpoints of the lower and upper segments⁶ and is four times less (why?). We can then bypass these points as well, as exemplified by Fig. 13.24(b), further reducing the corruption.

⁶That is, halfway between the midpoint and ground and halfway between the midpoint and V_{REF} .

Let us estimate the necessary value of C_1 in Fig. 13.24(a) for the 6-bit ADC example mentioned above, assuming C_2 and C_3 are absent. To this end, we first recognize that the error in Eq. (13.23) can be rewritten as $(m/4)R_u(m/2)I_u$, as if half of the total kickback noise current flowed from the ladder's midpoint [Fig. 13.25(a)]. We view this arrangement as a lumped equivalent of Fig. 13.21(b). Next, we construct Fig. 13.25(b) and approximate $mI_u/2$ by two pulses [Fig. 13.25(c)].⁷ For the comparator design in Chapter 7, we select

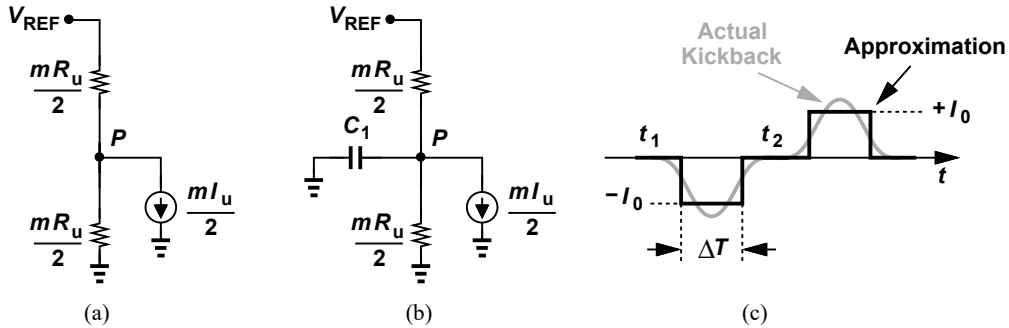


Figure 13.25 (a) Ladder model including kickback noise, (b) addition of a capacitor, and (c) approximation of kickback current.

$I_0 \approx (64/2)100 \mu\text{A}$ and $\Delta T \approx 50 \text{ ps}$. If C_1 is large enough to hold V_P relatively constant, then most of $-I_0$ flows through the capacitor from t_1 to t_2 , yielding a voltage change of

$$\Delta V_P = \frac{I_0}{C_1} \Delta T. \quad (13.24)$$

With an LSB of, say, 5 mV, ΔV_P must not exceed roughly 1 mV. It follows that $C_1 \geq 160 \text{ pF}$. This amount of capacitance takes substantial chip area.

We have thus far disregarded the effect of kickback on the *input* signal. Of the two approaches introduced here, the former may prove inevitable if the input corruption must be minimized. Bearing in mind that an ADC is typically preceded by a buffer, we ask what happens if the kickback noise current of all of the comparators flows through this buffer's output impedance (Fig. 13.26). In the foregoing 6-bit example and for the StrongArm design of Chapter 7, we have a peak current of $64 \times 200 \mu\text{A} = 12.8 \text{ mA}$. Thus, Z_{out} must absorb this current with a corruption less than 1 mV, a daunting challenge. The use of continuous-time differential pairs at the comparator inputs alleviates this issue, but the ladder bypass technique does not.

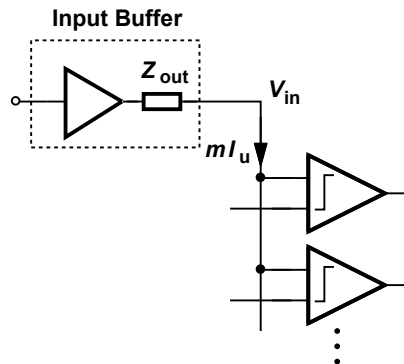


Figure 13.26 A buffer providing the analog input for a flash ADC.

⁷In the presence of C_1 , the lumped model becomes only a rough approximation of the actual ladder (why?).

Example 13.9

An engineer implements the buffer of Fig. 13.26 as a source follower with a bias current of 1 mA. How does this stage respond to the kickback noise of 64 StrongArm comparators?

Solution

As depicted in Fig. 13.27(a), the source follower output, V_{SF} , drops by a large amount when the compar-

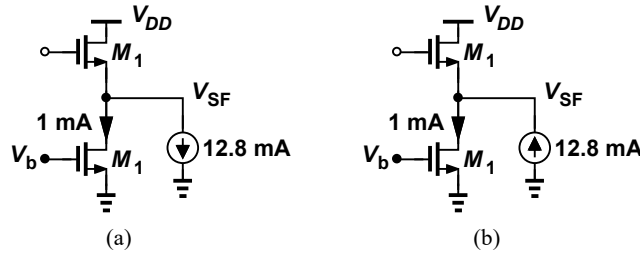


Figure 13.27 A source follower (a) sourcing, or (b) sinking a current.

tors pull 12.8 mA from M_1 . The flash ADC input therefore incurs a great error during this time. Similarly, V_{SF} rises—even above V_{DD} —when the comparators push 12.8 mA into the follower [Fig. 13.27(b)]. The actual situation is more complex for two reasons. First, the comparators present a large capacitance to the follower's output node. Second, the 200- μ A peak kickback current obtained in Chapter 7 has assumed that the comparator inputs are driven by ideal voltage sources. We thus expect a *lower* kickback noise current in the presence of the buffer's finite output impedance.

13.4.5 Input-Dependent Effects

Recall from Section 13.2 that flash ADCs can, in principle, perform sampling by means of their comparators. However, the absence of a front-end sampler exposes the converter to fast-varying inputs, introducing three issues.

The first relates to the coupling of the input to the ladder through the input capacitances of the comparators. Illustrated in Fig. 13.28(a), this phenomenon perturbs the ladder tap voltages and can be modeled in a manner

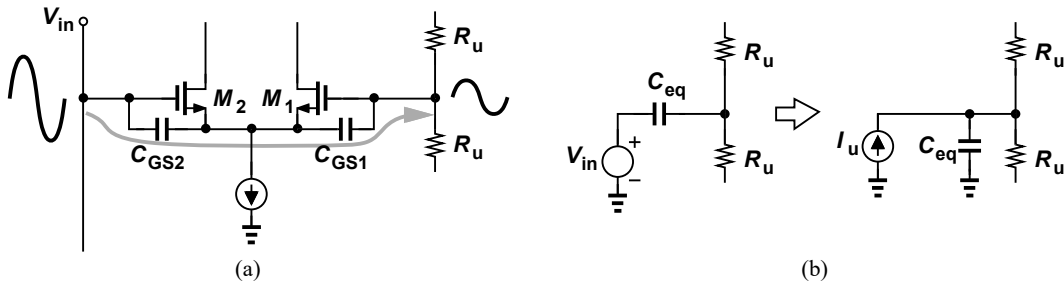


Figure 13.28 (a) Input feedthrough to the ladder, and (b) equivalent circuit.

similar to that in Fig. 13.21, as shown in Fig. 13.28(b). Here, $C_{eq} \approx C_{GS1}C_{GS2}/(C_{GS1} + C_{GS2})$ and the input has a frequency of ω_{in} . The current-source representation in Fig. 13.21 assumes that, in a well-designed ADC, the impedance associated with C_{eq} is much greater than the Thevenin resistance of the ladder, thereby delivering a current approximately equal to $V_{in}\omega_{in}C_{GS1}C_{GS2}/(C_{GS1} + C_{GS2})$. As Eq. (13.23) suggests, this disturbance can be minimized by reducing R_u . Alternatively, we can employ the bypass technique of Fig. 13.24.

The second effect manifests itself in the presence of timing mismatches between the comparators. Consider the situation shown in Fig. 13.29(a), where the clock paths of Comp_j and Comp_{j+1} suffer from a small delay mismatch, ΔT . As depicted in Fig. 13.29(b), Comp_j and Comp_{j+1} sample V_{in} at t_1 and $t_1 + \Delta T$, respectively. If V_{in} changes by more than 1 LSB in this interval, it is possible that Comp_j decides that

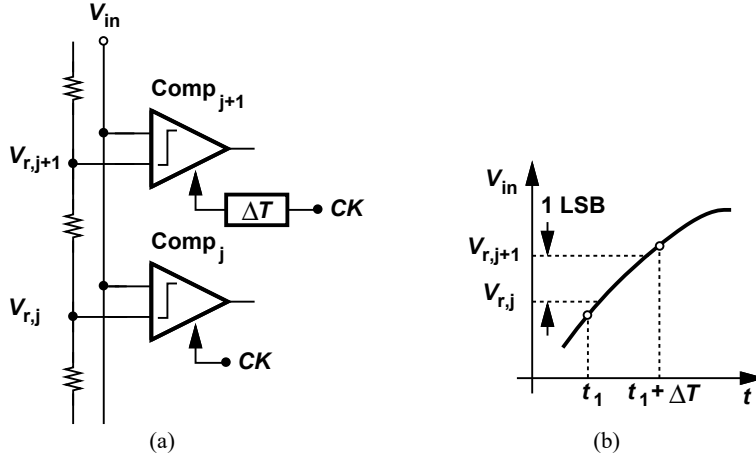


Figure 13.29 (a) A flash stage with timing mismatch, and (b) its effect on sampled values.

$V_{in} < V_{r,j}$, producing a low output, whereas Comp_{j+1} infers that $V_{in} > V_{r,j+1}$, yielding a high output. Thus, the thermometer code exhibits a bubble, potentially incurring a large error as it undergoes subsequent decoding (Section 13.5). With $V_{in} = V_0 \cos \omega_0 t$ and $1 \text{ LSB} = 2V_0/2^N$, we must ensure that the maximum input slope, $V_0\omega_0$, causes a change less than 1 LSB:

$$V_0\omega_0\Delta T < \frac{2V_0}{2^N}. \quad (13.25)$$

That is,

$$\Delta T < \frac{1}{2^{N-1}\omega_0}. \quad (13.26)$$

For example, a 6-bit, 10-GHz ADC sensing $\omega_0 = 2\pi(5 \text{ GHz})$ requires that $\Delta T < 1 \text{ ps}$. We study bubble correction in Section 13.5, but let us ask, what is the “correct” value of the thermometer code here? Since the ΔT mismatch can be assigned to either Comp_j or Comp_{j+1} , the correct code can include or exclude the topmost ONE. That is, the correct value can be (arbitrarily) considered to be equal to the number of ONES, or the number of ONES minus 1, or even the number of ONES plus 1 (if the bubble is replaced by a ONE). This point suggests the use of a ONES counter for the decoder.

The third input-dependent phenomenon arises from the nonlinearity and the finite bandwidth of the comparators’ input stage [3]. Consider the representative circuit shown in Fig. 13.30(a), with the bandwidth defined by R_D and C_D . If $V_{in} = V_0 \cos \omega_0 t$ and V_0 is small, the differential pair acts as a linear, low-pass filter, introducing a phase shift equal to

$$\theta = -\tan^{-1}(R_D C_D \omega_0). \quad (13.27)$$

Assuming $R_D C_D \omega_0 \ll 1 \text{ rad}$, we can find the equivalent delay, T_{D1} , if we divide the result by 2π and multiply it by the input period, T_0 :

$$T_{D1} = \frac{R_D C_D \omega_0}{2\pi} T_0 \quad (13.28)$$

$$= R_D C_D. \quad (13.29)$$

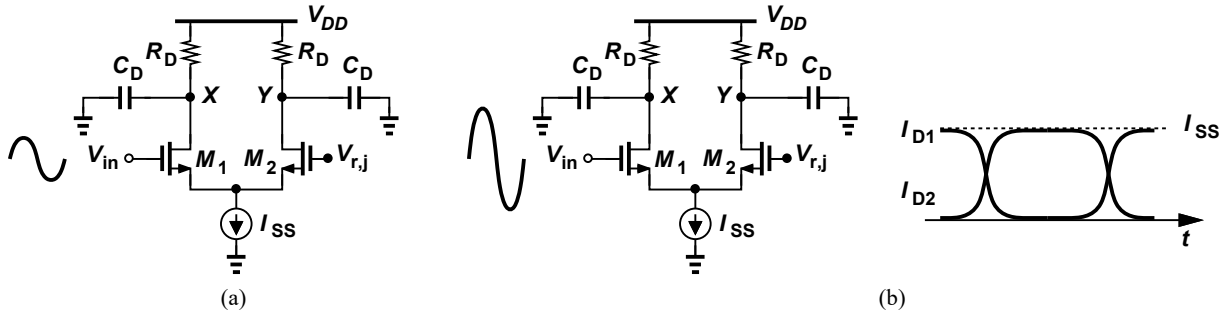


Figure 13.30 Differential pair with output bandwidth limitation sensing a (a) small, or (b) large sinusoid.

This delay is independent of the input amplitude or frequency.

On the other hand, if we assume V_0 is so large as to steer the tail current completely and we approximate the drain currents by square waves [Fig. 13.30(b)]. Upon flowing through R_D and C_D , each current transition causes a change of the form $\exp[-t/(R_D C_D)]$ in V_X or V_Y . The delay between the drain currents and the drain voltages is therefore equal to

$$T_{D2} = R_D C_D \ln 2. \quad (13.30)$$

Comparing (13.29) and (13.30), we observe that a change in the input amplitude translates to a change in the delay, a property absent in linear, time-invariant systems.

Let us now determine how the foregoing effect degrades a flash ADC's performance, noting that for a given input voltage, only one comparator makes a critical decision.⁸ Turning to the 6-bit example illustrated in Fig. 13.31(a) and assuming a full-scale sinusoidal input, we recognize that comparators near the top or bottom

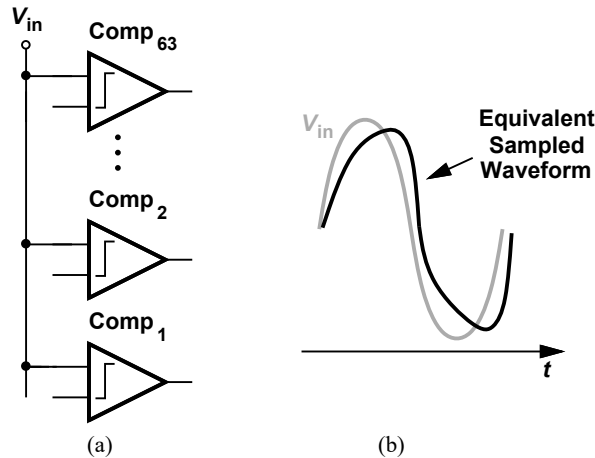


Figure 13.31 (a) A 6-bit flash ADC, and (b) its equivalent sampled waveform due to the finite bandwidth and nonlinearity of comparators.

of the array sense a lower slew rate than those in the middle. Thus, the peaks and valleys of the input are sampled with a *greater* delay [Fig. 13.31(b)]. If a comparator near the top makes a critical decision (because it senses a small input difference), then its longer delay displaces the sampling point by a greater amount. The distributed sampling in a flash ADC therefore distorts the signal [3]. This issue must be quantified by careful transistor-level simulations.

⁸We neglect other imperfections studied previously.

In Problem 13.12, we study how these three issues are alleviated if the flash ADC incorporates a single front-end sampler before the comparators.

13.5 Bubble Correction

We have observed that large comparator offsets or timing mismatches introduce bubbles in the thermometer code. We must now determine how different decoders respond to bubbles.

We begin with the ROM decoder studied in Section 13.3.1 and determine whether it is tolerant of bubbles. As illustrated in Fig. 13.32, both AND gates AND_{j-1} and AND_{j+1} detect a 1-to-0 transition in the

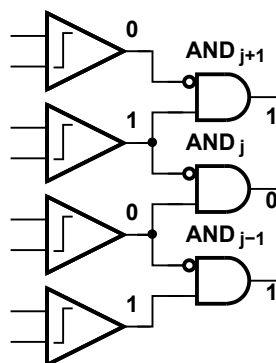


Figure 13.32 Effect of a bubble in the thermometer code.

thermometer code, thus generating high output levels and activating two rows of the ROM simultaneously. Similar to the case shown in Fig. 13.17(b), the resulting error can be quite large. In a 6-bit ADC, for example, the two rows can correspond to binary outputs of 011111 and 100001, yielding a result equal to 000001. The error is therefore about half of the full scale.

To avoid large errors, we can simply employ three-input AND gates to sense the thermometer code. As shown in Fig. 13.33, the gates generate a high output only if they sense a section of the thermometer code that has the form $T_j T_{j+1} T_{j+2} = 100$. Thus, only one row of the ROM is activated. This method equivalently replaces the bubble with a high level. That is, the “correct” code is equal to the number of ONEs plus 1.

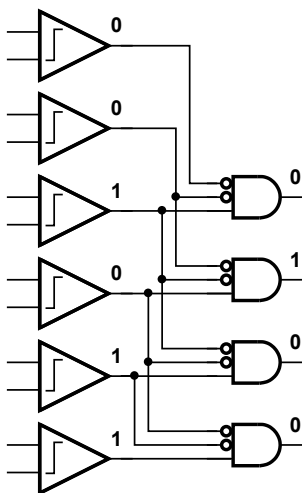


Figure 13.33 Use of three-input NANDs to tolerate a bubble.

It is possible that the thermometer code exhibits two bubbles, e.g., in the form of $\dots 11100100 \dots$. This can occur if, for example, the comparator offset exceeds 2 LSB (why?). The three-input AND gates in Fig.

13.33 fail to provide correction in this case; we must then resort to four-input gates but at the cost of more complex routing.

Recall from Section 13.4.3 that Gray coding can reduce the magnitude of errors resulting from metastability. We then surmise that it can also tackle bubbles. Let us revisit the table in Fig. 13.20(a) and determine the result in response to a bubble. From Table 13.1 and Eqs. (13.18)-(13.20), we observe that the output can incur an error of ± 2 LSB. The reader can extend this analysis to a 6-bit ADC.

Thermometer Code							Gray Code			Decimal Value	Error
T_7	T_6	T_5	T_4	T_3	T_2	T_1	G_3	G_2	G_1		
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	-2
0	0	0	1	0	1	1	1	1	1	5	+2
0	0	1	0	1	1	1	0	1	1	2	-2
0	1	0	1	1	1	1	1	0	0	7	+2
1	0	1	1	1	1	1	1	1	0	4	-2
0	1	1	1	1	1	1	1	0	0	7	0

Table 13.1 Effect of a bubble on the Gray code.

We next consider the MUX decoder of Section 13.3.2. Shown in Fig. 13.34 while receiving $T_1 \cdots T_7 = 1110100$, the circuit generates a binary output equal to 011, close to the thermometer value.

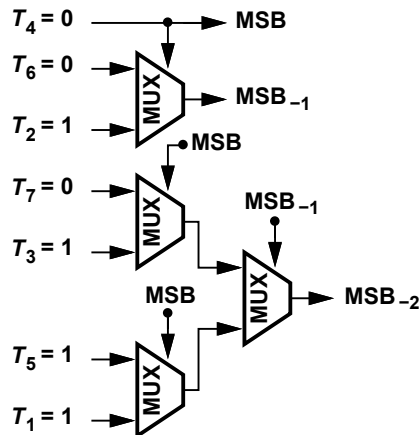


Figure 13.34 Effect of a bubble on a MUX decoder.

Similarly, for a thermometer code of the form 1111010, the output emerges as 110. Thus, this decoder tolerates bubbles to some extent. According to [1], the tolerance is similar to the three-input AND method depicted in Fig. 13.33.

The last decoder considered here is the ONES counter of Section 13.3.3. We expect this topology to be relatively insensitive to bubbles as it simply counts the number of high levels in the thermometer code regardless of their position [1]. The cost is greater complexity.

13.6 Fully-Differential Flash ADCs

We know from analog design principles that fully-differential operation is preferred for its tolerance of common-mode noise and its lower distortion. Another important advantage is the greater input voltage range afforded by differential signaling; if the ADC senses differential inputs of the form $V_{in}^+ = +V_0 \cos \omega_0 t + V_{CM}$ and $V_{in}^- = -V_0 \cos \omega_0 t + V_{CM}$, then the analog LSB is doubled. This means that the comparator offset can

be twice as large. The flash architectures studied thus far, however, can digitize only single-ended signals because their comparators have only two inputs. In this section, we deal with fully-differential designs.

It is instructive to first consider the input-output characteristic of a fully-differential ADC. Shown in Fig. 13.35 is a 4-bit example. As the differential input, $V_{in}^+ - V_{in}^-$, varies from $-V_{FS}$ to $+V_{FS}$, D_{out} begins at

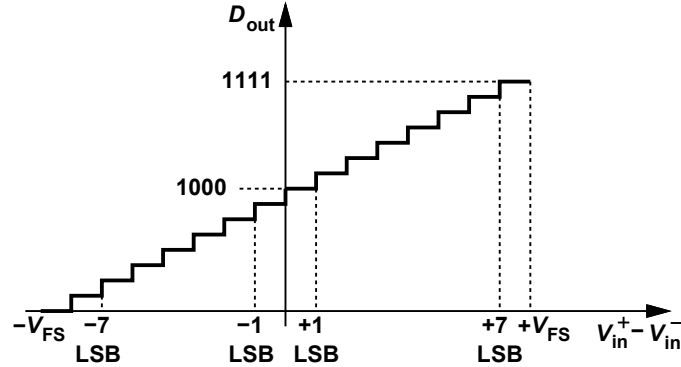


Figure 13.35 Characteristic of a fully-differential flash ADC.

0000 and reaches 1000 for $V_{in}^+ - V_{in}^- = 0$ and 1111 for $V_{in}^+ - V_{in}^- = +V_{FS} - 1 \text{ LSB}$. Thus, 15 comparators must compare $V_{in}^+ - V_{in}^-$ to reference values equal to -7 LSB , -6 LSB , $\dots$, 0 , $\dots$, $+6 \text{ LSB}$, and $+7 \text{ LSB}$.

13.6.1 Fully-Differential Comparators

The comparators studied previously can compare a single-ended voltage to a single-ended reference. For example, one such comparator determines whether $V_{in} > 100 \text{ mV}$ or $V_{in} < 100 \text{ mV}$ if the reference is equal to 100 mV . In our pursuit of differential flash ADCs, we must first construct topologies that sense a differential analog input and compare it to a differential reference. To understand this point, suppose we wish to determine whether the difference between the inputs $V_{in}^+ = +V_0 \cos \omega_0 t + V_{CM}$ and $V_{in}^- = -V_0 \cos \omega_0 t + V_{CM}$ is greater or less than some amount, e.g., 100 mV . We recognize that this is possible if we form $(V_{in}^+ - V_{in}^-) - (100 \text{ mV})$ and detect the sign of the result. While the 100-mV reference need not be a differential voltage, in practice, it is.

Subtraction in the Current Domain Since differential pairs can act as voltage-to-current converters, we surmise that the above subtraction operation can be realized in the current domain. Illustrated in Fig. 13.36, the idea is to generate two differential currents proportional to $V_{in}^+ - V_{in}^-$ and $V_{r1} - V_{r2}$ and subtract them.

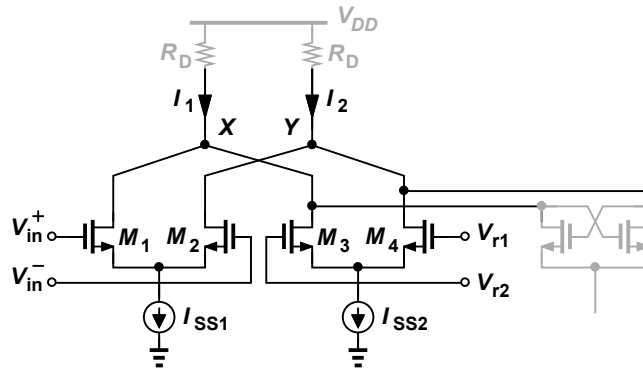


Figure 13.36 One method of comparing a differential input to a differential reference.

Note that $I_{SS1} = I_{SS2}$. We have $I_1 = I_{D1} + I_{D3}$, $I_2 = I_{D2} + I_{D4}$, and hence

$$I_1 - I_2 = (I_{D1} - I_{D2}) - (I_{D4} - I_{D3}) \propto (V_{in}^+ - V_{in}^-) - (V_{r1} - V_{r2}). \quad (13.31)$$

As shown in gray, I_1 and I_2 can flow through resistors to generate a proportional voltage, and a regenerative pair can be added to form a complete comparator.⁹

Example 13.10

The 4-bit flash ADC of Fig. 13.35 allows a single-ended peak-to-peak input swing of 0.5 V. Determine the necessary values of $V_{r1} - V_{r2}$ in Fig. 13.36 for such a design.

Solution

For a total differential input swing of 1 V, we have $1 \text{ LSB} = 1 \text{ V}/16 = 62.5 \text{ mV}$. Thus, $V_{r1} - V_{r2}$ ranges from $-7 \times 62.5 \text{ mV} = -437.5 \text{ mV}$ to $+7 \times 62.5 \text{ mV} = +437.5 \text{ mV}$ (Fig. 13.37).

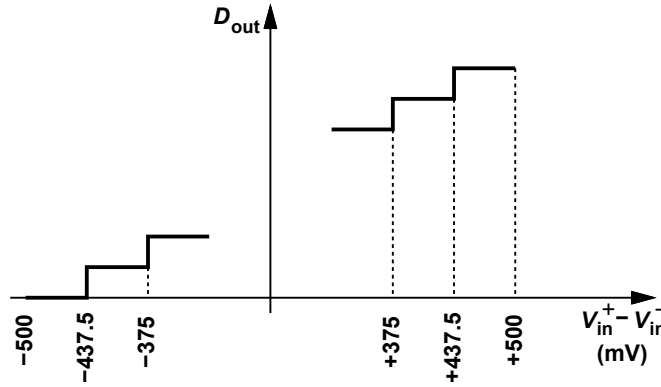


Figure 13.37 Reference values for a 4-bit flash ADC.

The circuit of Fig. 13.36, in principle, produces a differential output, $V_X - V_Y$, in proportion to $(V_{in}^+ - V_{in}^-) - (V_{r1} - V_{r2})$. But we know from analog design that the output of a differential pair follows the input voltage difference linearly only for small swings. For large inputs, the pair's transconductance, G_m , falls and eventually approaches zero. In the foregoing example, $V_{in}^+ - V_{in}^-$ can assume any value from -0.5 V to $+0.5 \text{ V}$ and $V_{r1} - V_{r2}$ from -437.5 mV to $+437.5 \text{ mV}$. If $V_{in}^+ - V_{in}^- = 450 \text{ mV}$ or 425 mV , then the comparator sensing $V_{r1} - V_{r2} = +437.5 \text{ mV}$ must make a critical decision. Unfortunately, this proves difficult as such large voltage differences can cause M_2 and M_3 to turn off, reducing the transconductance of each pair to zero, and making the decision meaningless. In other words, since these large differences yield $I_1 \approx I_{SS1}$ and $I_2 \approx I_{SS2}$, the output carries no knowledge of $V_{in}^+ - V_{in}^-$ or $V_{r1} - V_{r2}$.

The above issue can be resolved if we write

$$(V_{in}^+ - V_{in}^-) - (V_{r1} - V_{r2}) = (V_{in}^+ - V_{r1}) - (V_{in}^- - V_{r2}) \quad (13.32)$$

and form $V_{in}^+ - V_{r1}$ and $V_{in}^- - V_{r2}$ by means of differential pairs (Fig. 13.38). We still have $I_1 - I_2 \propto (V_{in}^+ - V_{in}^-) - (V_{r1} - V_{r2})$, but the comparator making a critical decision is not “saturated.” For example, suppose we wish to compare $V_{in}^+ - V_{in}^- = 450 \text{ mV}$ to $V_{r1} - V_{r2} = 437.5 \text{ mV}$. If $V_{in}^+ = 980 \text{ mV}$, $V_{r1} = 970 \text{ mV}$, $V_{in}^- = 530 \text{ mV}$, and $V_{r2} = 532.5 \text{ mV}$, then M_1 and M_2 sense a difference of 10 mV and M_3 and M_4 a difference of 2.5 mV , thus operating with nearly maximum transconductance. Figure 13.39 summarizes the two cases. We observe that the situation in Fig. 13.39(a) turns M_2 and M_3 nearly off, producing $I_{D1} \approx I_{SS1}$ and $I_{D4} \approx I_{SS2}$. In Fig. 13.39(b), on the other hand, $I_{D1} - I_{D2} \propto V_{in}^+ - V_{r1}$ and $I_{D4} - I_{D3} \propto V_{in}^- - V_{r2}$ because the voltage differences are fairly small.

⁹As in the current-steering structure studied in Chapter 5, the tail currents must be clocked.

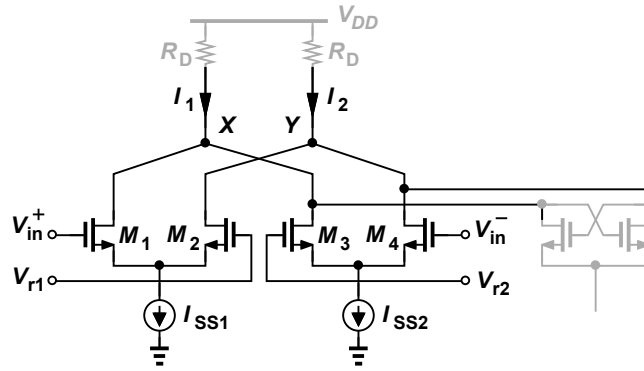


Figure 13.38 A better method of comparing a differential input to a differential reference.

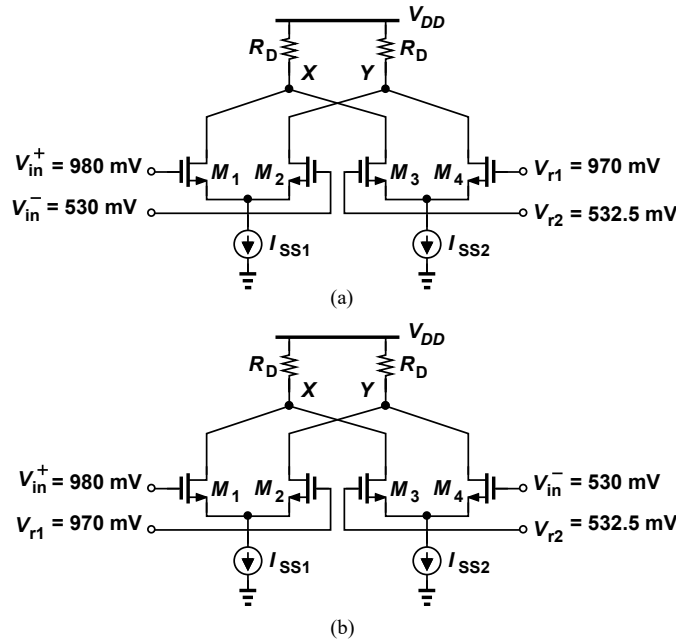


Figure 13.39 Behavior of two fully-differential subtractors for a large input difference.

Example 13.11

While one comparator in a flash ADC senses the above exemplary values, another can sense $V_{in}^+ - V_{in}^- = 450$ mV and $V_{r1} - V_{r2} = -7$ LSB = -437.5 mV. How does such a comparator behave?

Solution

If, in Fig. 13.39(b), $V_{in}^+ = 980$ mV, $V_{r1} = 532.5$ mV, $V_{in}^- = 530$ mV, and $V_{r2} = 970$ mV, then M_2 and M_4 turn off. This situation is acceptable because the comparator does *not* make a critical decision.

The use of two differential pairs in Fig. 13.38 entails three issues. First, the input offset voltage arises from both $M_{1,2}$ and $M_{3,4}$. We write the standard deviation of the offset as $\sigma^2 = \sigma_{12}^2 + \sigma_{34}^2$, where σ_{12} and σ_{34} denote those of $M_{1,2}$ and $M_{3,4}$, respectively. That is, $\sigma = \sqrt{2}\sigma_{12}$. (The mismatch between the drain resistors further contributes to this offset.) Compared to a single differential pair, the input offset rises by about 40%.

Second, the available voltage gain for $V_{in}^+ - V_{in}^-$ is less than that in a single differential pair. Let us neglect channel-length modulation and assume identical transistors operating in the small-signal regime. We have

$V_X - V_Y = -g_m(V_{in}^+ - V_{r1})R_D - g_m(-V_{in}^- + V_{r2})R_D$, where $g_m = g_{m1,2} = g_{m3,4}$ (why?). That is, $V_{in}^+ - V_{in}^-$ is amplified by a factor of $-g_m R_D$. Now, consider the pair in Fig. 13.40, which draws the same amount of power. In this case, the gain is equal to $-g_{ma,b} R_D$ but $g_{ma,b} > g_{m1,2}$ because M_a and M_b carry twice as much bias current as do M_1 - M_4 in Fig. 13.38.

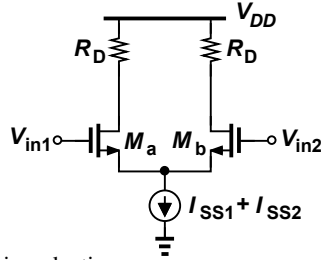


Figure 13.40 Differential pair showing gain reduction.

The third issue in Fig. 13.38 relates to the input voltage range that the differential pairs—and hence the flash ADC—can accommodate. We estimate the minimum and maximum allowable values of the input CM level, V_{CM} . It must be high enough to ensure that the tail currents operate in saturation. For example, if V_{GS1} , ..., V_{GS4} are around 400 mV and I_{SS1} and I_{SS2} require about 100 mV, then V_{CM} must be at least 500 mV. This means that V_{r1} and V_{r2} cannot be less than this value. How high can V_{CM} be? To obtain a reasonable voltage gain, say, around 3 or 4, R_D must be large enough, demanding a voltage drop, $R_D(I_{SS1} + I_{SS2})/2$, of roughly 0.5 V (while $M_{1,2}$ and $M_{3,4}$ operate with small input differences) [Fig. 13.41(a)]. For the transistors to remain in saturation, their gate voltages must be below $V_{DD} - R_D(I_{SS1} + I_{SS2})/2 + V_{TH}$, which amounts to 0.8 V if $V_{DD} = 1$ V and $V_{TH} = 0.3$ V. Thus, the single-ended input voltage range is about 0.3 V. Figure 13.41(b) depicts the input range and the reference ladder top and bottom voltages.

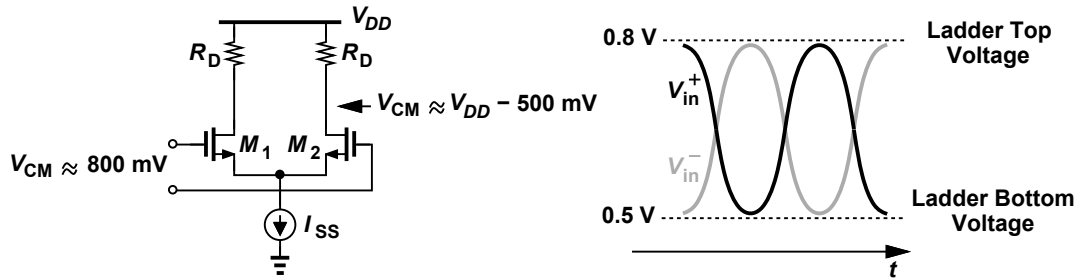


Figure 13.41 Differential pair and the ADC input voltage range.

Example 13.12

An engineer incorporates the circuit of Fig. 13.38 in the design of a flash ADC and generates the reference voltages by means of a resistor ladder. As shown in Fig. 13.42, the ladder top and bottom voltages have a proper difference but they are accidentally shifted down. Do the comparators operate correctly?

Solution

No, they do not. Suppose, for example, that $V_{in}^+ \approx 0.8$ V and $V_{in}^- \approx 0.5$ V. The comparator making a critical decision must sense $V_{r1} \approx 0.4$ V and $V_{r2} \approx 0.1$ V. But the large values of $V_{in}^+ - V_{r1}$ and $V_{in}^- - V_{r2}$ cause $M_{1,2}$ and $M_{3,4}$ in Fig. 13.38 to switch completely under these conditions, yielding little transconductance. In other words, the common-mode levels of the analog input and the resistor ladder must be sufficiently close to each other.

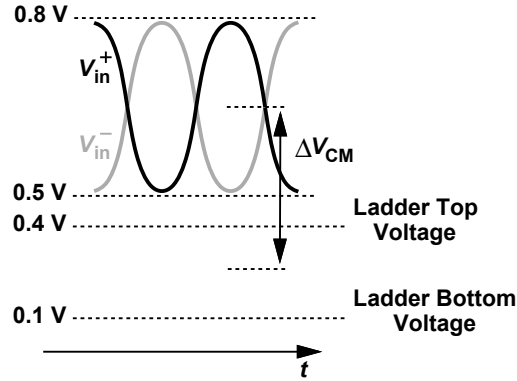


Figure 13.42 Input and reference voltages with a large CM difference.

Subtraction in the Voltage Domain It is possible to form $(V_{in}^+ - V_{in}^-) - (V_{r1} - V_{r2})$ in the voltage domain and benefit from certain properties. In this case, the comparator itself has only two inputs but is preceded by a switched-capacitor network that performs the subtraction.

Let us begin with the structure in Fig. 13.43(a), noting that C_1 and C_2 store V_{in}^+ and V_{in}^- , respectively, while S_1 and S_2 are on. Next, we turn off the switches and connect the left plates of C_1 and C_2 to V_{r1} and

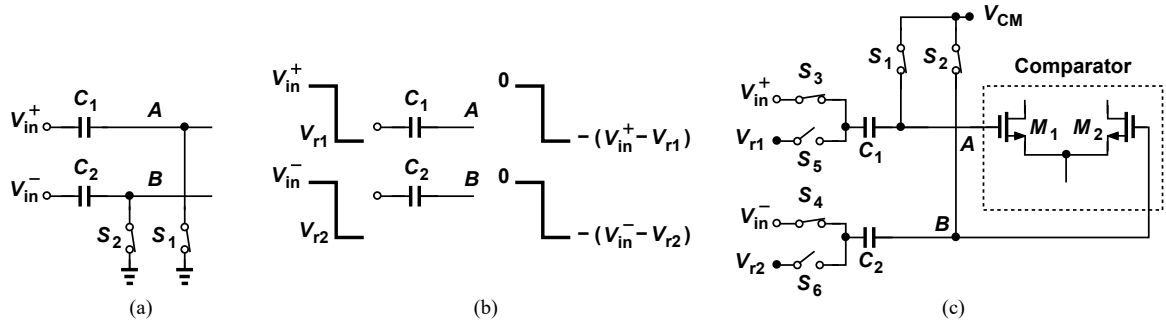


Figure 13.43 (a) Two capacitors sensing V_{in}^+ and V_{in}^- , (b) response of circuit to input changes, and (c) fully-differential sampling and subtraction.

V_{r2} , respectively [Fig. 13.43(b)]. How much are the new voltages at A and B ? Since nodes A and B are now floating, the charge on C_1 and C_2 cannot change, and hence the input voltage steps must travel to these nodes. That is, V_A changes from zero to $-(V_{in}^+ - V_{r1})$ (why?) and V_B from zero to $-(V_{in}^- - V_{r2})$. We then have

$$V_A - V_B = -(V_{in}^+ - V_{r1}) + (V_{in}^- - V_{r2}), \quad (13.33)$$

which is the difference that we have sought (except for a negative sign). But the common-mode level of V_A and V_B is around zero and ill-suited to drive an NMOS differential pair.

In the next step of this development, we add the necessary switches to the left plates of the capacitors and also use a proper CM level on their right plates [Fig. 13.43(c)]. Here, S_1 - S_4 are on in the sampling mode, and S_5 and S_6 are on during comparison. Note that V_A and V_B begin from V_{CM} and change by $V_{r1} - V_{in}^+$ and $V_{r2} - V_{in}^-$, respectively.¹⁰

The topology of Fig. 13.43(c) offers three advantages over the current-domain addition method of Fig. 13.38. First, it employs a single differential pair, exhibiting less offset (and higher transconductance for a given power dissipation). Second, by virtue of capacitive coupling, it can accommodate rail-to-rail input swings. Third, the circuit is fairly insensitive to the common-mode difference between the input and the

¹⁰We define the “change” as the final value minus the initial value.

ladder voltages. We also note that the sampling action obviates the need for a front-end sampler in the flash ADC.

Example 13.13

How do we select the value of C_1 and C_2 in Fig. 13.43(c)?

Solution

The value must be large enough to minimize the attenuation due to parasitics at A and B . We recognize that S_1 and M_1 contribute capacitances to A , and S_2 and M_2 to B . As shown in Fig. 13.44, these parasitics act as a capacitor, C_p , to ground, yielding

$$\Delta V = -(V_{in}^+ - V_{r1}) \frac{C_1}{C_1 + C_p}. \quad (13.34)$$

We must therefore select C_1 and C_2 to be about five to ten times C_p .

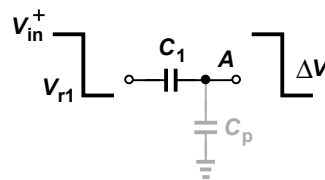


Figure 13.44 Effect of parasitics at node A .

Example 13.14

Explain how the input and ladder common-mode difference, ΔV_{CM} , illustrated in Fig. 13.42 affects the structure of Fig. 13.43(c).

Solution

From Fig. 13.45, we note that, if the input CM level jumps by ΔV_{CM} , so does the output CM level.¹¹ That is, the input stage of the comparator is driven by an input CM level of $V_{CM} - \Delta V_{CM}$ and must still operate properly.

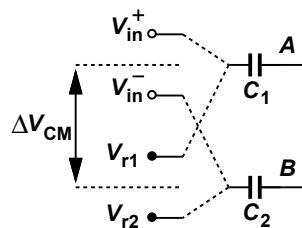


Figure 13.45 Behavior of subtractor for a large CM difference.

¹¹For CM change studies, the reader can instead assume a zero differential input and a zero differential reference.

The voltage subtraction network in Fig. 13.43(c) does lead to a large input capacitance for the flash ADC. In the sampling mode, each comparator presents C_1 plus the parasitics of S_3 and S_5 to V_{in}^+ . Since C_1 is more than five times the gate-source capacitance of M_1 , the overall input capacitance is at least five times that of the differential pairs in Fig. 13.38.¹² This is the principal drawback of the switched-capacitor approach. We also remark that S_1 and S_2 should preferably turn off before S_3 and S_4 , requiring additional clock phases.

It is possible to reduce the capacitance presented to the analog input by changing the switching sequence in Fig. 13.43(c). As depicted in Fig. 13.46, we first sample the reference voltages on the capacitors, and then

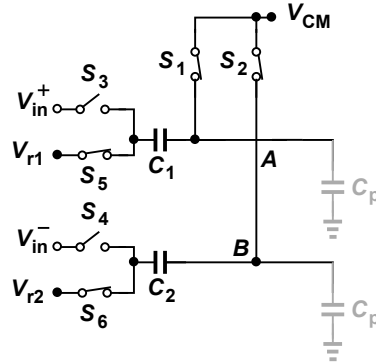


Figure 13.46 Alternative switching sequence.

switch the left plates of the capacitors to V_{in}^+ and V_{in}^- after S_1 and S_2 are turned off. Now, the analog inputs see the series combination of C_1 (or C_2) with C_p , a much smaller value. However, we face two issues. First, the analog input is no longer sampled by this network, i.e., a front-end sampler must precede this network. Second, the switching of C_1 and C_2 onto the ladder creates substantial disturbance in the tap voltages. Similar to other ladder disturbance mechanisms studied in the previous section, this effect can be alleviated through the use of large bypass capacitors.

13.6.2 U-Shaped Flash Architecture

The foregoing developments enable us to construct a fully-differential flash ADC. Beginning with the 4-bit characteristic example shown in Fig. 13.35, we note that the comparators require differential references ranging from -7 LSB to zero to $+7$ LSB. Depicted in Fig. 13.47(a) is an example illustrating how Comp_1 , Comp_2 , and Comp_8 sense differential references equal to -7 LSB, -6 LSB, and zero, respectively. (The other seven comparators are not shown.) Voltages V_{REF}^+ and V_{REF}^- are chosen such that $V_{REF}^+ - V_{REF}^- = 8$ LSB and their CM level, $(V_{REF}^+ + V_{REF}^-)/2$, is close to that of V_{in}^+ and V_{in}^- .

We observe the routing complexity in this arrangement, seeking a more modular structure. Let us then “fold” the ladder (as explained Chapter 9) and double the number of its resistors, arriving at the “U-shaped” flash ADC shown in Fig. 13.47(b). Noting that each resistor sustains a voltage drop equal to 0.5 LSB, we provide a difference of -7 LSB to Comp_1 by attaching its reference inputs to nodes N_{15} and N_1 . Similarly, since $V_{N14} - V_{N2} = -6$ LSB, we tie Comp_2 to these nodes. Thus, each comparator, along with two unit resistors, forms a module that can be duplicated. One exception is Comp_8 . For $\text{Comp}_9, \dots, \text{Comp}_{15}$, we swap their reference inputs so that they sense $+1$ LSB, $\dots, +7$ LSB, respectively. The reference routing complexity is thus reduced considerably.

The architecture of Fig. 13.47(b) faces another routing issue: the thermometer codes on the left and right sides must somehow reach the following decoder. For this reason, the comparators should be laid out such that their horizontal dimension is as small as possible. Alternatively, as shown in Fig. 13.47(c), we can place the comparators on only one side of the ladder. In this case, we wish to minimize the *vertical* dimension of the comparators (why?). Another benefit of this approach is that it positions the first and last comparators in close proximity, shortening the routing for the MUX decoder in Section 13.3.2.

¹²The ratio is actually closer to 10 than to 5 because the capacitance seen by V_{in}^+ in Fig. 13.36 is about half of C_{GS1} (why?).

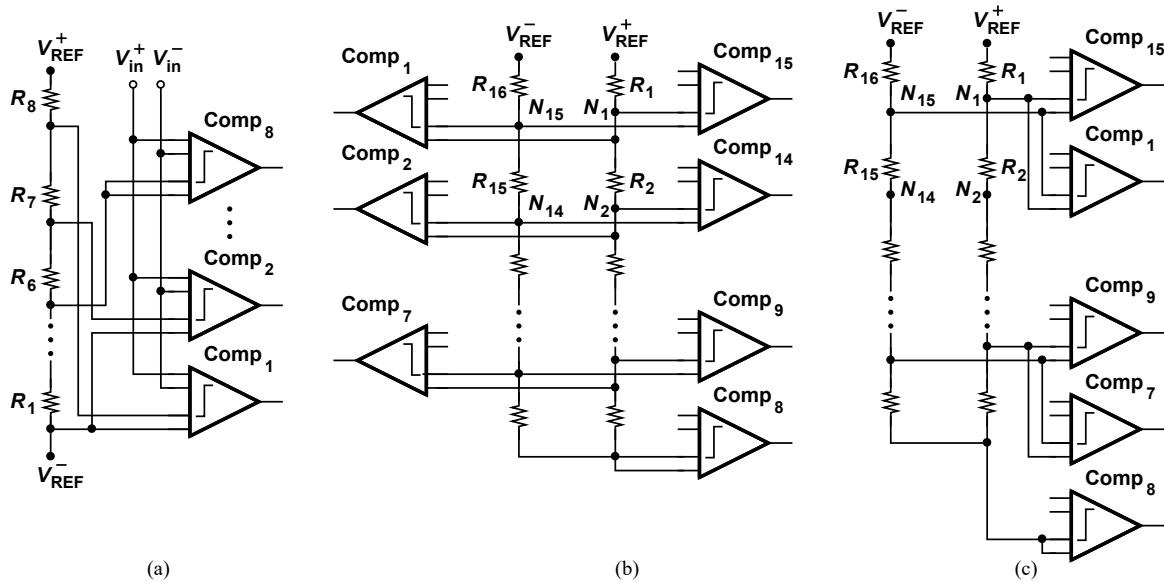


Figure 13.47 (a) Simple floor plan of a fully-differential flash ADC, (b) use of a folded ladder to reduce routing complexity, and (c) grouping of comparators to reduce decoder routing.

13.7 Flash ADC Variants

A number of architecture and circuit techniques have been invented to ease the trade-offs in flash stages. We study two here.

Recall that the input capacitance of the converter grows sharply with the resolution. It is possible to alleviate this issue through the use of “interpolation.” In this context, we view each comparator as a differential pair, A_j , followed by a latch. Let us first examine the differential pair outputs in Fig. 13.48(a) as V_{in} varies from be-

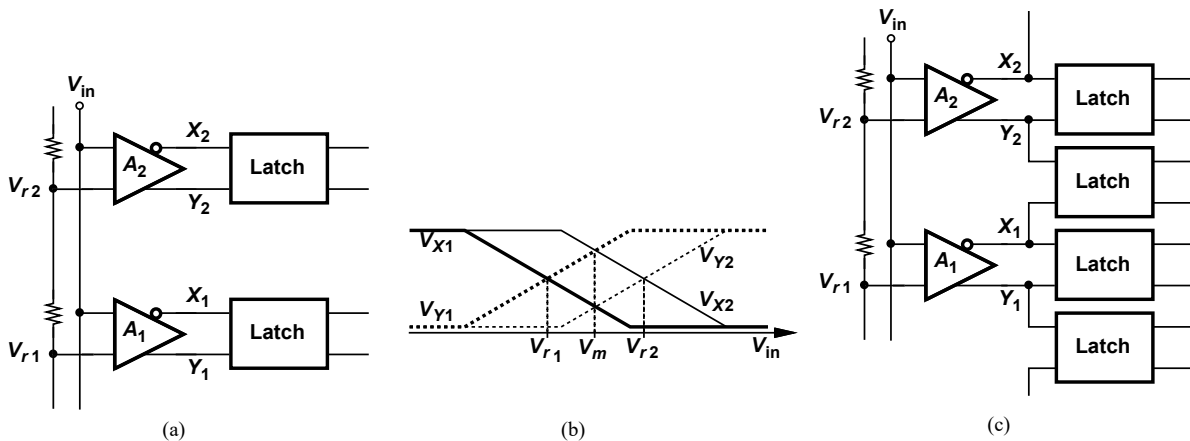


Figure 13.48 (a) Flash stage showing comparators' input differential pairs, (b) characteristics provided by the differential pairs, and (c) flash stage using interpolation.

low V_{r1} to above V_{r2} . Noting that $V_{X1} = V_{Y1}$ for $V_{in} = V_{r1}$ and $V_{X2} = V_{Y2}$ for $V_{in} = V_{r2}$, we can construct the characteristics shown in Fig. 13.48(b). Now, we recognize that $V_{X1} = V_{Y2}$ at $V_{in} = V_m = (V_{r1} + V_{r2})/2$. That is, a latch sensing these two voltages (or V_{Y1} and V_{X2}) can detect when V_{in} crosses midway between V_{r1} and V_{r2} . As depicted in Fig. 13.48(c), the interpolating flash [4] architecture doubles the resolution without doubling the number of differential pairs. Note that the latches produce a thermometer code.

The performance improvement afforded by interpolation appears almost free, but it does require each comparator include a differential pair and suffer from its power dissipation. In particular, a simple StrongArm latch would not suffice for the comparator design in this environment.

Another approach to reducing the complexity and power is illustrated in Fig. 13.49(a) [5]. Here, a front-

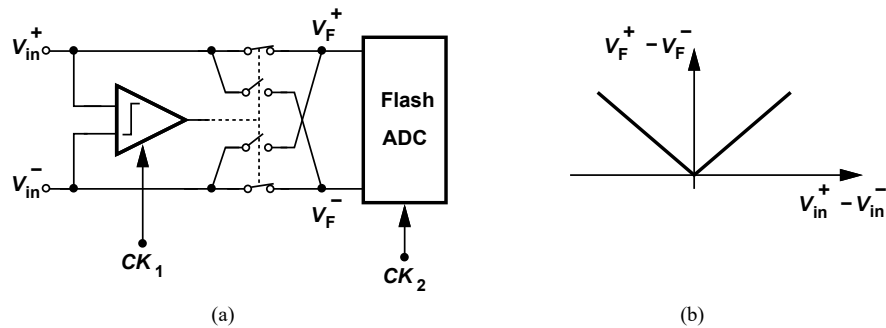


Figure 13.49 (a) Flash stage preceded by a polarity detector, and (b) the resulting characteristic.

end comparator detects, under the command of CK_1 , the *polarity* of $V_{in}^+ - V_{in}^-$ and accordingly routes the inputs through two of the switches such that the flash stage always senses a positive differential value. Once $V_F^+ - V_F^-$ settles, CK_2 strobes the flash ADC so as to generate the remaining bits. Plotted in Fig. 13.49(b), the resulting characteristic is an example of “folding” and produces the most-significant bit (MSB) from the front-end comparator. We observe that the overall input capacitance, power, and complexity are approximately halved because the number of comparators drops from 2^N to $2^{N/2} + 1$. These benefits accrue at the cost of more than twofold reduction in speed: not only must the front-end comparator respond to an input difference of, say, 0.5 LSB and activate the proper switches, but also $V_F^+ - V_F^-$ must also settle while the switches drive the input capacitance of the flash stage. Note that the offset of the front-end comparator must remain less than 1 LSB.

References

- [1]. E. Sall and M. Vesterbacka, “Comparison of two thermometer-to-binary decoders for high-performance flash ADCs,” *Proceedings of NORCHIP*, Nov. 2005.
- [2]. F. Kaess, et al, “New encoding scheme for high-speed flash ADCs,” *IEEE Proceedings of Circuits and Systems*, vol. 1, pp. 5-8, June 1997.
- [3]. R. van de Plassche and P. Baltus, “An 8-bit 100-MHz full-Nyquist analog-to-digital converter,” *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 1334-1344, Dec. 1988.
- [4]. C. Lane, “A 10-bit 60 Msps flash ADC,” *Proceedings of BCTM*, Sept. 1989.
- [5]. B. Verbruggen, J. Craninckx, M. Kuijk, P. Wambacq, and G. van der Plas, “A 2.6-mW 6 bit 2.2 GS/s fully dynamic pipeline ADC in 40-nm digital CMOS,” *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 2080-2090, Oct. 2010.

Problems

- 13.1. In a 2-bit version of the flash ADC shown in Fig. 13.1, the bottom terminal of R_1 is tied to a voltage source equal to ΔV , and V_{REF} is raised to $V_{REF} + \Delta V$. How does the ADC characteristic change?
- 13.2. In a 2-bit version of the flash ADC shown in Fig. 13.1, V_{REF} is raised to $1.1V_{REF}$. Sketch the input-output characteristic and determine whether this change causes offset, gain error, or nonlinearity.
- 13.3. Consider a 2-bit version of the flash ADC shown in Fig. 13.1. (a) If Comp_1 exhibits an offset equal to 0.2 LSB, construct the input-output characteristic and the DNL profile. (b) Repeat (a) if only Comp_2 incurs an offset equal to 0.2 LSB. (c) Repeat (a) if both Comp_1 and Comp_2 exhibit offsets equal to 0.2 LSB. Consider two cases: the two offsets have the same sign or opposite signs.
- 13.4. Explain what happens to the V_{X2} and V_{Y2} waveforms in Fig. 13.2 if M_3 and M_4 exhibit an offset.
- 13.5. Reference voltages V_{r2} and V_{r3} in Fig. 13.3(b) are accidentally swapped, i.e., V_{X2} and V_{Y2} cross when V_{in} crosses V_{r3} . Examine the thermometer code as V_{in} varies from less than V_{r1} to greater than V_{r3} .
- 13.6. Consider a 2-bit flash ADC. Construct the 1-of- n decoder and the ROM necessary to convert the thermometer code to a binary output. Verify that the output varies from 00 to 11 as the thermometer code ranges from 000 to 111.
- 13.7. Suppose the StrongArm comparator in Fig. 13.5 enters the evaluation mode at $t = 0$. Due to the large layout of the flash ADC, it is difficult to guarantee that the clock controlling the ROM's precharge devices rises at the same time. Do we prefer that this clock rises earlier or later than the comparator's clock?
- 13.8. Consider a 2-bit flash ADC. Explain whether a bubble can occur in its thermometer code output in the form of $T_3T_2T_1 = 110$ or 101 . (T_3 is the topmost entry.)
- 13.9. Consider a 3-bit flash ADC. Explain how a "double bubble" in the form of $T_8 \cdots T_1 = 00101011$ can occur.
- 13.10. We wish to raise the resolution of a flash ADC by 1 bit.
 - (a) By what factor does the input capacitance increase? Assume no comparator offset cancelation.
 - (b) By what factor does the maximum kickback-induced perturbation of the resistor ladder increase?
- 13.11. Suppose $\Delta T < 0$ in Fig. 13.29(a), i.e., Comp_{j+1} is clocked *earlier* than Comp_j is. Determine the output thermometer code.
- 13.12. Suppose a flash ADC is preceded by an explicit sampling circuit. Explain which one of these issues is alleviated: metastability, feedthrough of the analog input to the ladder, input-dependent comparator delay, and clock path mismatch between comparators.
- 13.13. A flash ADC tested in the laboratory exhibits a bubble in its thermometer code. Explain how we can determine whether comparator offset or clock path mismatch causes the bubble.
- 13.14. Determine the small-signal value of $V_X - V_Y$ in Fig. 13.36 in terms of V_{in}^+ , V_{in}^- , V_{r1} , and V_{r2} . Disregard the cross-coupled NMOS pair.
- 13.15. Suppose differential pair M_1 - M_2 in Fig. 13.43(c) represents the input of a StrongArm comparator. Can we choose $V_{CM} = V_{DD}$? Explain.
- 13.16. Determine the bandwidth of the circuit shown in Fig. 13.43(c) in the sampling mode and explain why S_1 and S_2 cannot be arbitrarily narrow MOS devices.
- 13.17. We wish to accommodate a rail-to-rail analog input in the circuit of Fig. 13.43(c). Explain which switches must incorporate both NMOS and PMOS devices.
- 13.18. Construct a 2-bit fully-differential flash ADC using a resistor ladder and the network shown in Fig. 13.43(c).
- 13.19. In the flash ADC of the previous problem, explain when transient currents are drawn from the ladder and estimate the capacitances drawing such currents.
- 13.20. Construct a 3-bit version of the U-shaped flash ADC shown in Fig. 13.47(c). Assuming that the comparator sensing a zero differential reference voltage suffers from a 0.5-LSB offset, sketch the input-output characteristic.

SAR ADC Fundamentals

Invented in the mid-twentieth century, “successive approximation register” (SAR) ADCs found renewed interest in the early 2000s and have become popular for their potentially low power consumption. In this chapter, we deal with the fundamentals of SAR ADC design. For the sake of brevity, we drop the term “register” and refer to this architecture as the “successive-approximation” ADC while retaining the abbreviation “SAR.” The reader is encouraged to first review Chapters 12 and 13.

We begin with the notion of “binary search” as a means of quantizing analog signals and therefrom develop the basic SAR topology. We then study the most popular type, namely, those using capacitor DACs. In particular, we focus on two variants of this architecture that employ top-plate or bottom-plate sampling.

14.1 General Concepts

14.1.1 Quantization by Binary Search

As seen in previous chapters, quantization can be viewed as finding an “analog estimate” of the input, V_{in} , and generating the digital equivalent of this estimate. In flash ADCs, the reference ladder provides the estimates, and the comparators identify one as the closest lower value. By virtue of its massively “parallel” operation, the flash architecture performs conversion in one clock cycle.

In SAR ADCs, on the other hand, we allow multiple clock cycles for one conversion and follow a binary search algorithm. The ADC computes analog estimates that successively converge toward V_{in} . To understand the concept, let us consider an input range from zero to V_{REF} [Fig. 14.1(a)]. Binary search begins by nominating $V_{REF}/2$ as the best analog estimate. We compare V_{in} to this estimate and act accordingly; depending on whether $V_{in} < V_{REF}/2$ or $V_{in} > V_{REF}/2$, we next identify $V_{REF}/4$ or $3V_{REF}/4$, respectively, as the best estimate. This “decision-directed” binary search requires comparing V_{in} to the most recent analog estimate and continuing the search according to the *polarity* of their difference. We call this difference the “residue” and denote it by $V_{in} - \alpha V_{REF}$, where $\alpha = 1/2, 3/4, 5/8$, etc., in Fig. 14.1(a). The goal of binary search is to reduce the residue to less than 1 LSB. Figure 14.1(b) illustrates how the decision result in one cycle leads to comparison with the proper analog estimate in the next cycle.

14.1.2 Basic SAR Structure

Our foregoing thoughts suggest that quantization by binary search requires a means of delivering the analog estimates in Fig. 14.1(a), e.g., a DAC, and a means of comparison. Let us then construct the loop shown in Fig. 14.2(a), where, after it is clocked, the comparator decides whether $V_{in} > V_{DAC}$ or $V_{in} < V_{DAC}$ and changes V_{DAC} so that $|V_{in} - V_{DAC}|$ becomes smaller. But how is the DAC’s digital input provided?! This necessitates a register that holds the data for the DAC [Fig. 14.2(b)]. The loop operates as follows. First, the input is sampled and held. Also, the register’s output is set to $100 \cdots 0$ so that V_{DAC} assumes the best estimate, $V_{REF}/2$. Next, the comparator is clocked and decides whether the register’s content should change to $0100 \cdots 0$ ($\equiv V_{REF}/4$) or $1100 \cdots 0$ ($\equiv 3V_{REF}/4$). The loop continues to travel along the decision tree

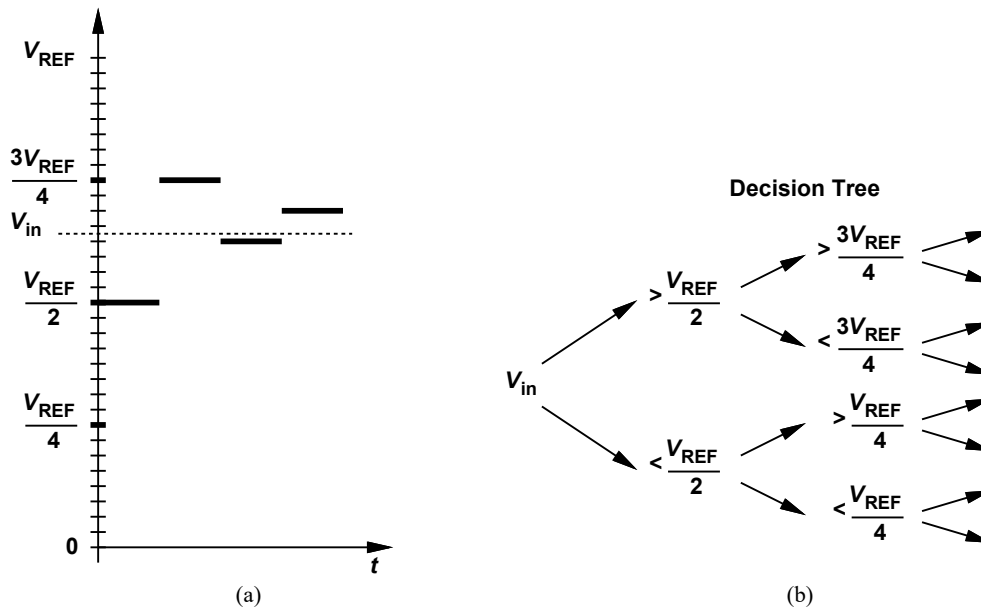


Figure 14.1 (a) ADC input range, and (b) binary search tree.

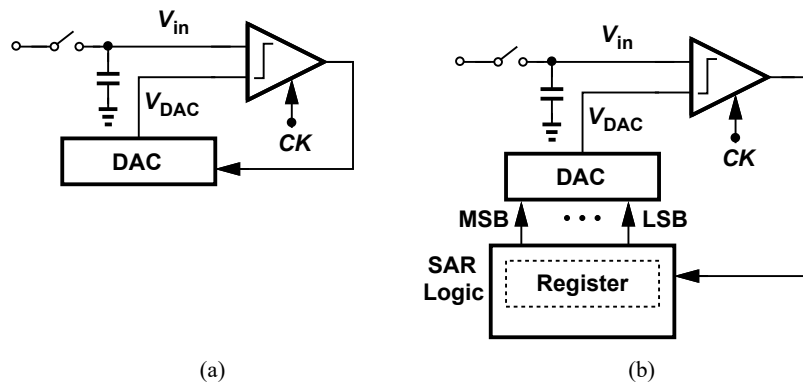


Figure 14.2 (a) Simple SAR ADC, and (b) addition of a register to hold the data for DAC.

shown in Fig. 14.1(b) until V_{DAC} settles close to V_{in} , at which point the register holds the corresponding digital output.

The arrangement of Fig. 14.2(b) can also be viewed as a negative-feedback loop that, by virtue of the comparator's high gain, attempts to minimize the "error," i.e., $V_{in} - V_{DAC}$. As explained later, this loop may settle with some "ringing."

The renewed popularity of the SAR architecture stems from its ability to operate with zero static power consumption. From previous chapters, we surmise that the StrongArm comparator and a capacitor DAC, along with CMOS logic, satisfy this principle. Moreover, the SAR ADC can achieve high resolutions, albeit at the cost of DAC complexity, and unlike "pipelined" topologies (Chapter 17), does not require linear amplification. In addition, in contrast to flash structures, the ADC can tolerate a comparator offset *greater* than 1 LSB (Chapter 15).

Example 14.1

The SAR ADC of Fig. 14.2(b) is designed for a resolution of 4 bits. If the sampled analog input is equal to $V_{in1} = 1.5$ LSB or $V_{in2} = 2.2$ LSB, sketch V_{DAC} as a function of time.

Solution

We expect the binary output to be equal to 0001 for V_{in1} and 0010 for V_{in2} . After the analog input is held, we begin with $V_{DAC} = 8$ LSB and detect the sign of $V_{in1} - V_{DAC}$, concluding that V_{DAC} must be lowered (Fig. 14.3). We then set the register value to 0100, obtaining $V_{DAC} = 4$ LSB after $t = t_1$. We now clock the

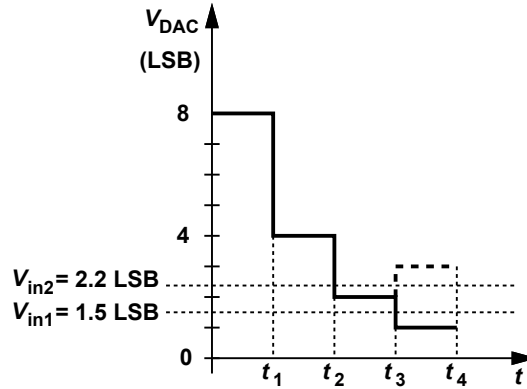


Figure 14.3 DAC voltage in a SAR ADC vs. time.

comparator again (at t_2). This decision requires that V_{DAC} be further reduced. The register is therefore set to 0010 $\equiv 2$ LSB, the comparator is clocked (at $t = t_3$), and, since $V_{in1} - V_{DAC}$ is still negative, the register's content is updated to 0001 $\equiv 1$ LSB. It appears that the digital output available after t_3 properly represents V_{in1} .

But let us now consider the case of $V_{in2} = 2.2$ LSB. As illustrated in Fig. 14.3, the V_{DAC} trajectory is the same as before until $t = t_3$. At this point, the comparator decides that $V_{in2} - V_{DAC} > 0$, commanding the register to raise its content to 0011 $\equiv 3$ LSB.¹ This value is incorrect. We note that V_{in} has changed from V_{in1} to V_{in2} by 2.2 LSB $- 1.5$ LSB $= 0.7$ LSB but the digital value available after $t = t_3$ by 2 LSB. In fact, no input value maps to 0010. A similar situation arises for all of the other even digital values. We must therefore wait for the DAC to settle after t_3 and clock the comparator one more time at t_4 . Since the comparator detects that $V_{DAC} > V_{in2}$, the LSB is set to 0. The loop thus takes four cycles for a resolution of 4 bits.

The foregoing example reveals the principal drawback of the SAR architecture: it requires N clock cycles for a resolution of N bits. In this spirit, we distinguish between the clock cycle, T_{CK} , (also called the SAR cycle) and the conversion period, T_c , which is equal to NT_{CK} . To this value we must add the acquisition time of the sampler as well.

Example 14.2

Plot the residue, $V_{in} - V_{DAC}$ (the difference sensed by the comparator), in the previous example as a function of time if $V_{in} = 2.2$ LSB.

Solution

The loop begins with $V_{in} - V_{DAC} = 2.2$ LSB $- 8$ LSB $= -5.8$ LSB (Fig. 14.4) and forces it toward zero by reducing V_{DAC} by 4 LSB and then 2 LSB. Since $V_{res} > 0$ after $t = t_2$, the loop pushes it back toward zero by lowering V_{DAC} by 1 LSB.

¹We say V_{DAC} experiences an “undershoot” between t_2 and t_3 .

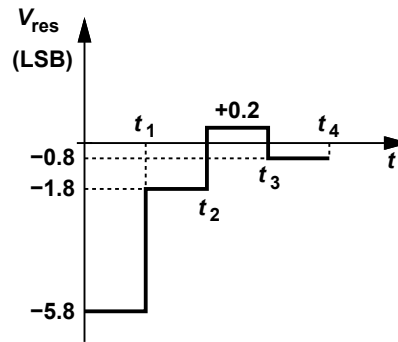


Figure 14.4 Residue voltage in a SAR ADC vs. time.

It is interesting to note that the act of setting V_{DAC} to 8 LSB in Fig. 14.3 need not occur in the *conversion* mode, we can simply set the MSB to 1 during the sampling mode, thus saving one DAC settling interval. However, this is not possible in the most common SAR architecture, namely, that based on “charge redistribution” (Section 14.2).

14.1.3 Concept of Time Trellis

In the analysis of SAR ADCs, we must often monitor the DAC output as a function of time so as to evaluate static and dynamic errors. It is therefore helpful to begin with a time diagram showing all of the possible trajectories that V_{DAC} can take. The result is a “time trellis,” shown in Fig. 14.5. According to the analog input voltage, the loop traverses a specific trajectory. For example, if $V_{in} = 12.5$ LSB, then V_{DAC} and its register follow the sequence 8 LSB, 12 LSB, 14 LSB, and 13 LSB, exhibiting an “overshoot” with respect to V_{in} .

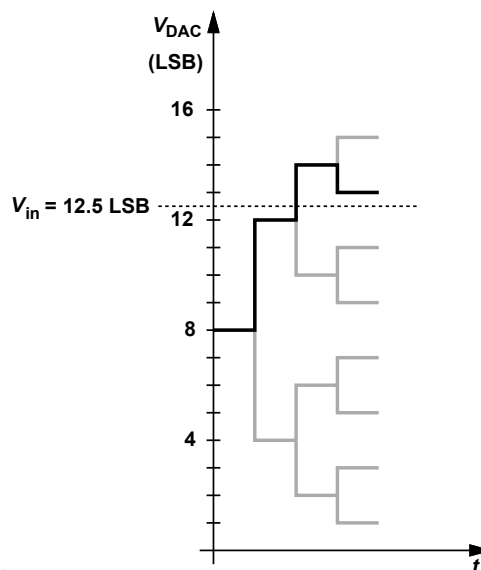


Figure 14.5 Time trellis example.

In summary, in each SAR cycle, we (a) raise the next bit in the register to a logical ONE, and (b) if V_{DAC} has experienced an overshoot or undershoot, we *undo* the last bit change.

14.1.4 Simple SAR Logic

Based on the understanding developed above, we wish to implement the SAR logic shown in Fig. 14.2(b). We articulate the loop’s binary-search operation as: (a) set V_{DAC} to $V_{REF}/2$ by writing $100 \dots 0$ to the register,

(b) if $V_{in} > V_{DAC}$, maintain $MSB = 1$ and set the next MSB in the register to 1, or (c) if $V_{in} < V_{DAC}$, reset the MSB to 0 and set the next MSB to 1. In a given SAR cycle, we are concerned with one bit in the register, concluding that the loop must remember which bit is presently of interest. We then employ another register, which we call the “pointer,” to keep track of this bit. Illustrated in Fig. 14.6(a), the complete ADC drives the register by the pointer’s 1-of- n code output. For the 12.5-LSB input example of Fig. 14.5, we obtain the sequence depicted in Fig. 14.6(b). As the pointer shifts a 1 from left to right, the corresponding bits in the register are set to proper values according to the comparator’s decision. Note that the register’s value is “refined” from 1110 to 1101 in the last cycle.

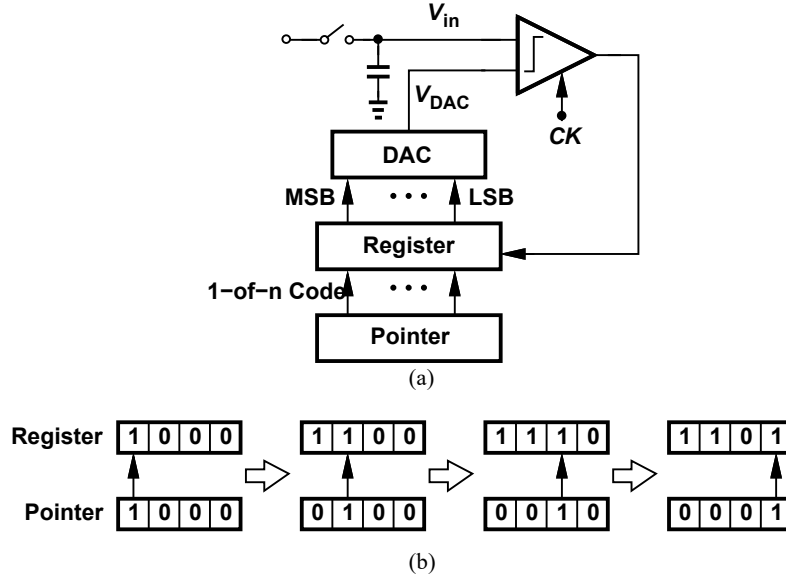


Figure 14.6 (a) Addition of pointer to SAR logic, and (b) time evolution of the pointer and register contents.

Let us now implement the register and the pointer. The register’s cells must satisfy two conditions, namely, be enabled or disabled by the pointer’s output, and generate a 1 or a 0 under the command of the comparator. These cells are therefore formed by RS latches that also have an enable input [Fig. 14.7(a)].² In the RS latch example of Fig. 14.7(b), we observe that the circuit holds the previous state if Enable is low and responds to

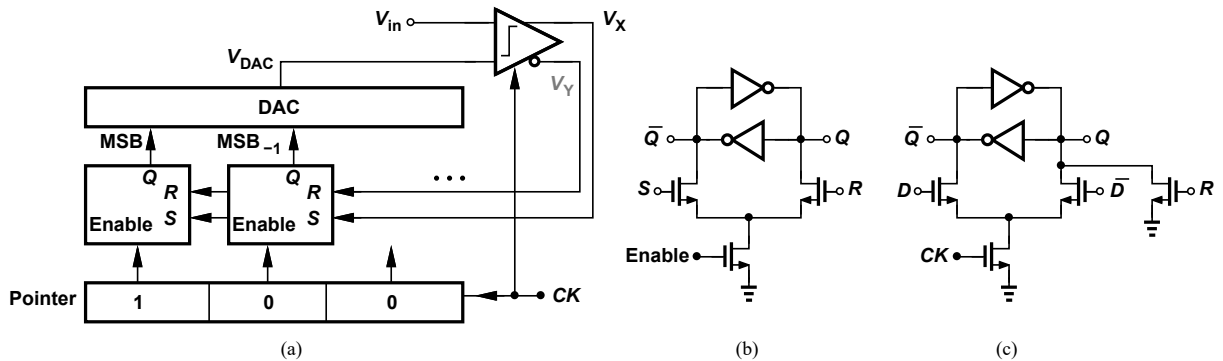


Figure 14.7 (a) Realization of SAR logic, (b) RS latch example, and (c) pointer cell example.

the R and S inputs if it is high. In the first SAR cycle, the pointer enables the leftmost cell and the comparator is clocked. If $V_{in} < V_{DAC} = V_{REF}/2$, then the comparator output goes high, the set input of this cell rises,

²This RS latch is derived from that studied in Chapter 5.

and its ONE state is retained. On the other hand, for $V_{in} > V_{DAC}$, the reset input is activated. To begin the next SAR cycle, the 1 in the pointer is shifted to the right, and the loop's action is repeated.

Example 14.3

As explained in Chapter 5, the StrongArm comparator is typically followed by an RS latch. Is that necessary in Fig. 14.7(a)?

Solution

No, it is not, provided that the comparator outputs reaching the register remain *low* in the precharge mode, as was accomplished in Chapter 5 by inverters. The register cells then retain their states.

The pointer in Fig. 14.7(a) consists of cascaded flipflops that shift a single 1 from left to right in each SAR cycle. We recognize that both the register and the pointer must be set to $100 \cdots 0$ while the ADC is in the sampling mode (why?). The pointer cells thus need set and reset inputs as well. Figure 14.7(c) depicts an example, where the circuit acts as a latch with D and $\overline{D}$ serving as its complementary inputs and Q and $\overline{Q}$ as its complementary outputs. The R input can reset the latch asynchronously.

14.1.5 Computing V_{in} from D_{out}

In our SAR ADC studies, we must construct the input-output characteristic while considering circuit imperfections. For example, we wish to determine the decision thresholds if the DAC suffers from offset (Chapter 15). Such tasks require that we obtain the input voltage range corresponding to a given digital output value. We develop the necessary skills in this section.

Suppose an ideal 4-bit SAR ADC generates $D_{out} = 0101 = 5$ LSB. To obtain the corresponding range of V_{in} , we work *backwards* and begin with the least significant bit of D_{out} , bearing in mind from Example 14.1 that, at the end of conversion, the LSB is set to 1 if $V_{DAC} < V_{in}$. Thus, V_{DAC} can be slightly less than V_{in} , as shown in Fig. 14.8(a) between t_3 and t_4 . We then turn to the next bit (LSB₊₁ or MSB₋₂) and decide

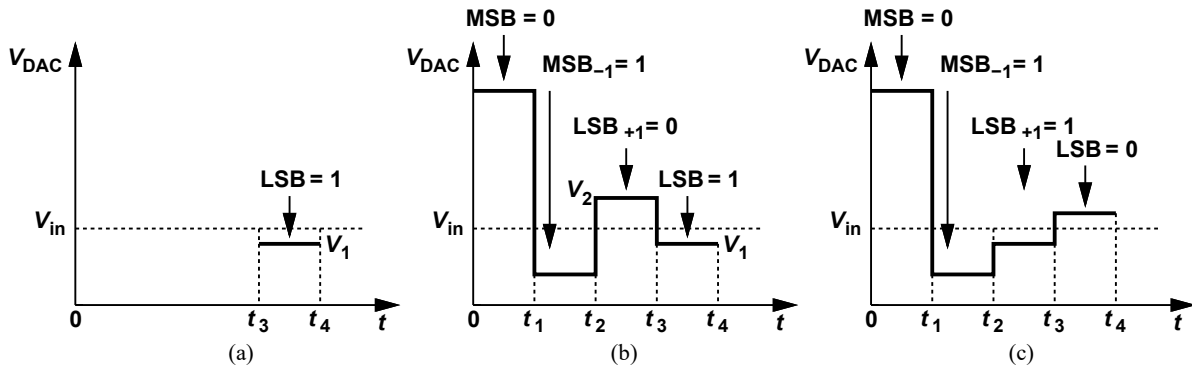


Figure 14.8 DAC voltage (a) for LSB= 1, (b) vs. time for $D_{out} = 0101$, and (c) vs. time for $D_{out} = 0110$.

that V_{DAC} must have been *greater* than V_{in} if this bit was set to 0. Similarly, MSB₋₁ = 1 indicates that $V_{DAC} < V_{in}$, and MSB = 0 implies that $V_{DAC} > V_{in}$. These observations yield the V_{DAC} trajectory shown in Fig. 14.8(b). We also recognize that (a) the steps in V_{DAC} are equal to 2^j LSB, where j is equal to 0 for the last jump and equal to 2 for the first, (b) V_{DAC} begins with a value of 8 LSB at $t = 0$, and (c) $V_2 = 6$ LSB and $V_1 = 5$ LSB.

The V_{DAC} behavior in Fig. 14.8(b) readily reveals the input range that translates to $D_{out} = 0101$: V_{in} can be slightly greater than V_1 or slightly less than V_2 . That is, $5 \text{ LSB} < V_{in} < 6 \text{ LSB}$.

Let us consider the case of $D_{out} = 0110 = 6$ LSB. As shown in Fig. 14.8(c), the comparator decision at t_4 has delivered a logical ZERO, suggesting that $V_{DAC} > V_{in}$. Working backwards, we determine from LSB₊₁ = 1 that $V_{DAC} < V_{in}$ at t_3 and also at t_2 . We continue to construct the trajectory while bearing in

mind that the steps in V_{DAC} must have a height of 2^j LSB. Since $MSB = 0$, we decide that $V_{DAC} > V_{in}$ before t_1 . It follows that V_{in} can range from a minimum of $8 \text{ LSB} - 4 \text{ LSB} + 2 \text{ LSB} = 6 \text{ LSB}$ to a maximum of $8 \text{ LSB} - 4 \text{ LSB} + 2 \text{ LSB} + 1 \text{ LSB} = 7 \text{ LSB}$.

The foregoing examples appear rather obvious, but the ability to construct the V_{DAC} trajectory from D_{out} proves valuable in our subsequent studies.

14.2 Charge Redistribution SAR ADCs

14.2.1 SAR ADCs with CDACs

The zero static power consumption of capacitor DACs makes them attractive for SAR ADCs. Figure 14.9 conceptually illustrates such an arrangement. The operation is similar to that of the generic loop in previous sections: the logic sets the DAC MSB to 1, thus generating $V_{DAC} = V_{REF}/2$, and then the comparator is clocked. The search continues until V_{DAC} settles close to the input. The SAR logic output can be in binary form, driving binary-weighted units within the CDAC. For resolutions above 4 bits, however, we must incorporate thermometric coding (Chapter 11). The presence of DAC capacitors in Fig. 14.9 naturally raises

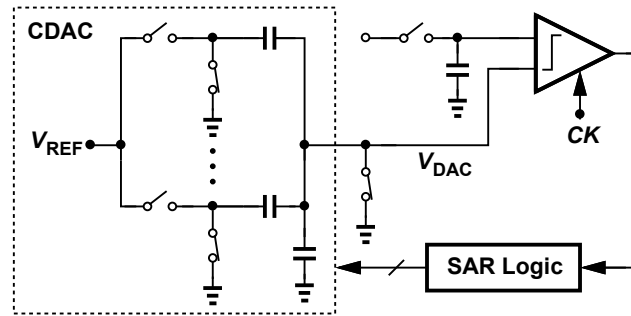


Figure 14.9 SAR ADC using a CDAC.

a question: can we combine the sampling circuit with the DAC? This is, in fact, possible, endows certain advantages, and merits a detailed study. The reader is encouraged to review the switched-capacitor concepts in Chapter 2 before continuing with this development.

14.2.2 Basic Top-Plate Sampling

Initial Thoughts Consider the simple circuit shown in Fig. 14.10(a), where C_1 has an initial condition of V_0 . If the bottom plate of the capacitor jumps down by V_{REF} , so does the top plate because the charge on C_1 cannot change. Thus, V_X goes from V_0 to $V_0 - V_{REF}$. Figure 14.10(b) modifies this idea so that an input, V_{in} , is first applied to C_1 . After S_0 and S_a turn off and S_b turns on, V_X reaches $V_{in} - V_{REF}$. Along with a comparator attached to X , the circuit can determine whether $V_{in} > V_{REF}$ or $V_{in} < V_{REF}$, which is not quite what SAR operation needs.

In the next step, we turn to the binary-weighted array in Fig. 14.10(c), where the bottom plate of the capacitor on the left swings from V_{REF} to zero after V_{in} is sampled. We analyze the equivalent circuit shown in Fig. 14.10(d) by superposition, where V_{in} and the step from V_{REF} to zero are considered the “inputs” to the system. First, we assume the bottom plate of the capacitance on the left does not change, recognizing that V_X is equal to the sampled input, V_{in} . Then, we assume a zero V_{in} is sampled, concluding that a change of $-V_{REF}$ on the bottom plate translates to a change of $-V_{REF}/2$ in V_X . Summing these two results reveals that V_X goes from V_{in} to $V_{in} - V_{REF}/2$. The polarity of this difference can be detected by a comparator. The reader is encouraged to determine why the rightmost unit capacitor in Fig. 14.10(c) is necessary.

Simple Example The foregoing thoughts lead to the 3-bit “top-plate sampling” ADC depicted in Fig. 14.11(a), where $C_0 = C_u$. Before analyzing the loop, we recall from Section 14.1.3 that in each SAR cycle, we raise the next bit in the register to a logical ONE and undo the last bit change if there has been an

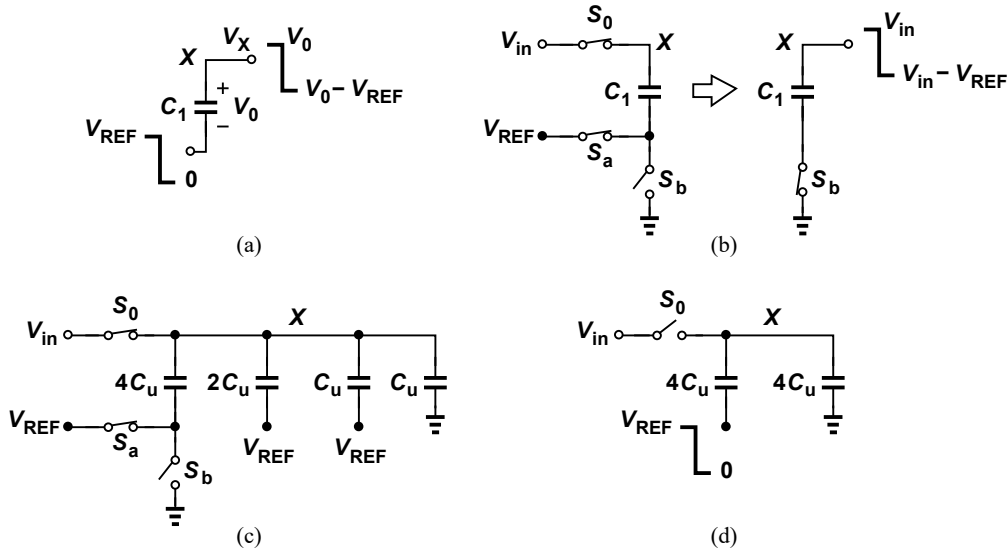


Figure 14.10 (a) Voltage step at the bottom plate of capacitor, (b) addition of switches, (c) simple CDAC with top-plate sampling, and (d) equivalent circuit.

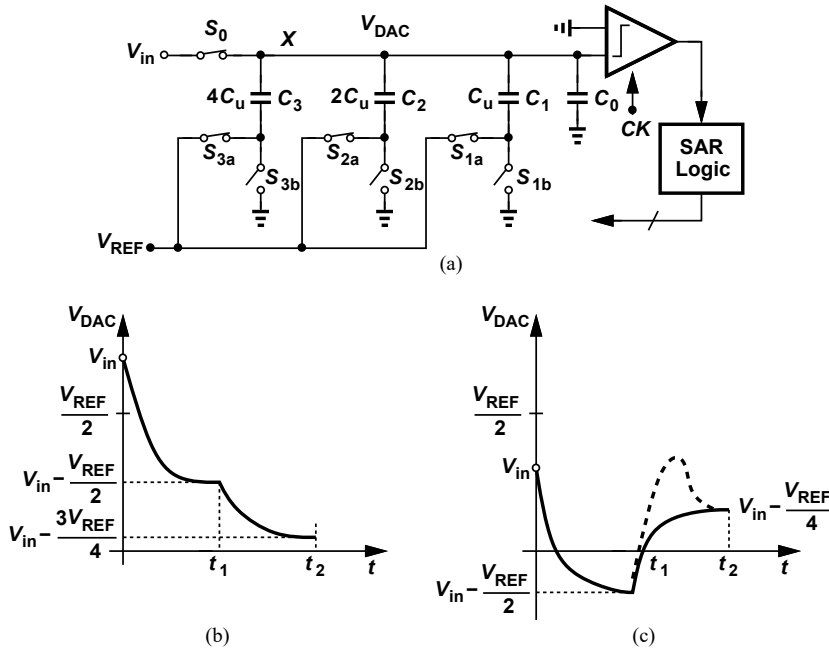


Figure 14.11 (a) SAR ADC with top-plate sampling, (b) DAC waveform for $V_{in} > V_{REF}/2$, and (c) DAC waveform for $V_{in} < V_{REF}/2$.

overshoot or undershoot. In the sampling mode, the bottom plates of C_1 - C_3 are tied to V_{REF} . After S_0 and S_{3a} turn off, S_{3b} turns on, allowing V_{DAC} to settle to $V_{in} - V_{REF}/2$. Next, the comparator is clocked, distinguishing between two cases. (1) If $V_{in} > V_{REF}/2$, S_{2a} turns off and S_{2b} turns on. The reader can show that this jump causes V_{DAC} to fall by $V_{REF}/4$, reaching $V_{in} - 3V_{REF}/4$ [Fig. 14.11(b)]. (2) If $V_{in} < V_{REF}/2$, S_{3b} turns off and S_{3a} turns on again, yielding $V_{DAC} = V_{in}$ and, at the same time, S_{2a} turns off and S_{2b} turns on. Consequently, $V_{DAC} = V_{in} - V_{REF}/4$ [Fig. 14.11(c)]. We add the dashed V_{DAC} waveform here, conceptually showing that V_{DAC} returns to V_{in} and then drops to $V_{in} - V_{REF}/4$, but in reality, it simply

travels from $V_{in} - V_{REF}/2$ to $V_{in} - V_{REF}/4$. Note that so far, the DAC has settled twice and the comparator has been clocked once.

Example 14.4

Explain how the loop in Fig. 14.11(a) should proceed after $t = t_2$.

Solution

The comparator is clocked at $t = t_2$, and the loop must subtract $V_{REF}/8$ from the present V_{DAC} value by switching the bottom plate of C_1 from V_{REF} to zero. In addition, based on the comparator's decision, the previous bit may need to be reset so as to undo the recent bit change. Specifically, in Fig. 14.11(b), if $V_{DAC} = V_{in} - 3V_{REF}/4$ happens to be *less* than zero, then the bottom plate of C_2 must be switched back to V_{REF} after $t = t_2$. Similarly, the case in Fig. 14.11(c) calls for the same action if $V_{DAC} = V_{in} - V_{REF}/4$ happens to be positive.

These observations indicate that the negative-feedback action around the SAR loop in Fig. 14.11(a) drives V_{DAC} toward zero, as if it were a “virtual ground.” This stands in contrast to the operation of the ADC in Fig. 14.7(a). In the generic architecture, V_{DAC} assumes values such as $V_{REF}/2$, $V_{REF}/4$, etc., and the comparator senses $V_{in} - V_{DAC}$. In Fig. 14.11(a), on the other hand, V_{DAC} itself appears as $V_{in} - V_{REF}/2$, $V_{in} - V_{REF}/4$, etc., and the comparator compares it to zero.

Under what condition does V_{DAC} in Fig. 14.11(a) fall monotonically? We recognize that all of its intermediate values, i.e., $V_{in} - V_{REF}/2$, $V_{in} - V_{REF}/2 - V_{REF}/4$ and $V_{in} - V_{REF}/2 - V_{REF}/4 - V_{REF}/8$, must be positive. That is, if V_X is slightly less than V_{REF} , V_{DAC} exhibits no overshoot or undershoot.

The SAR loop of Fig. 14.11(a) exemplifies a “charge-redistribution” ADC; it can be seen that C_j carries a charge equal to $C_j(V_{in} - V_{REF})$ at the beginning, but, as the loop converges, some of this charge may be redistributed. In essence, C_1 - C_3 “pump” negative voltage jumps into X so as to bring V_{DAC} toward zero.

In order to solidify our understanding, let us examine the loop in Fig. 14.11(a) for two input values, namely, $V_{in1} = 4.7$ LSB and $V_{in2} = 3.7$ LSB. In the former case, V_{DAC} evolves as shown in Fig. 14.12(a), falling by 4 LSB and then by 2 LSB. We thus far have $MSB = 1$ and $MSB_{-1} = 1$.³ The undershoot requires that we set MSB_{-1} to zero while raising the next bit, $MSB_{-2} = LSB$ to 1. Now, $V_{DAC} = -0.3$ LSB. The DAC

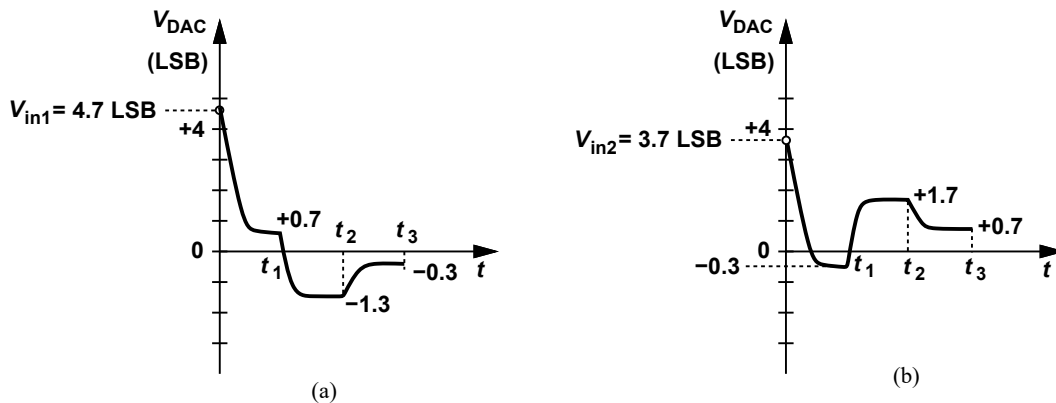


Figure 14.12 DAC waveforms for (a) $V_{in1} = 4.7$ LSB, and (b) $V_{in2} = 3.7$ LSB.

has settled three times but the comparator has been clocked at only t_1 and t_2 . However, the register holds a value of 101, which is incorrect. Thus, the comparator must be clocked once more at t_3 ; as a result, MSB_{-2} is returned to zero, yielding a final output equal to 100.

³Note that a logical ONE switches the bottom plate of the corresponding capacitor from V_{REF} to zero.

Turning to the second case, $V_{in2} = 3.7$ LSB, we follow the trajectory depicted in Fig. 14.12(b). With $MSB = 1$, an undershoot occurs, necessitating that, at $t = t_1$, the MSB be set to zero while MSB_{-1} is raised to 1. Next, at $t = t_2$, MSB_{-2} is also raised, subtracting 1 LSB from V_{DAC} . The register therefore provides an output equal to 011, which is correct. Nonetheless, the comparator must be clocked once more to ascertain that V_{DAC} has not become negative between t_2 and t_3 .

14.2.3 Single-Ended Top-Plate-Sampling ADC

The SAR ADC of Fig. 14.11(a) forces V_{DAC} toward zero, inevitably generating a *negative* voltage at the comparator input in some cases. For the comparator to operate properly, its input common-mode level must be chosen around $V_{DD}/2$, e.g., around $V_{REF}/2$. We can simply tie its upper input to this voltage, hoping that the loop directs V_{DAC} accordingly. But the difficulty is that the switching of the capacitor bottom plates performs only digital-to-analog conversion and does not provide a level shift. That is, the -0.3 -LSB undershoot in Fig. 14.12(b) still occurs because the first DAC switching action is not driven by the comparator.

Figure 14.13(a) depicts the modified 3-bit SAR example. We note three changes: one comparator input is

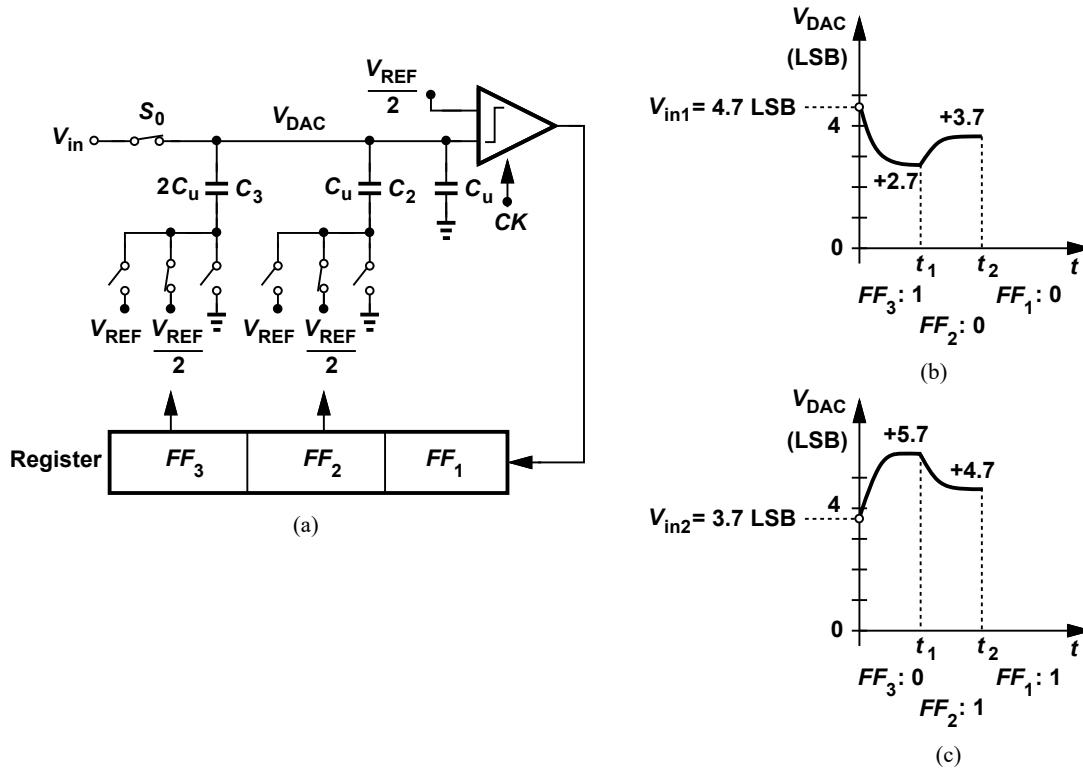


Figure 14.13 (a) Modified SAR ADC, (b) DAC waveform for $V_{in1} = 4.7$ LSB, and (c) DAC waveform for $V_{in2} = 3.7$ LSB.

tied to $V_{REF}/2$, the capacitor bottom plates can be connected to zero, $V_{REF}/2$, or V_{REF} , and the DAC has a resolution of only 2 bits! Thus, the DAC is driven by two flip-flops. In the sampling mode, C_3 and C_2 sense a voltage difference of $V_{in} - V_{REF}/2$. A fourth difference with respect to the ADC of Fig. 14.11(a) is that here the conversion begins by clocking the comparator. We surmise that, in any SAR cycle, if $V_{DAC} > V_{REF}/2$, the DAC must lower its output voltage by switching a bottom plate from $V_{REF}/2$ to zero. Conversely, if $V_{DAC} < V_{REF}/2$, a bottom plate must swing from $V_{REF}/2$ to V_{REF} .

Let us study the circuit's operation for two cases, namely, $V_{in1} = 4.7$ LSB and $V_{in2} = 3.7$ LSB. We assume that, for the bottom plate of, e.g., C_3 to swing from $V_{REF}/2$ to zero, a ONE must be written onto FF_3 . In the sampling mode, the bottom plates of C_2 and C_3 are at $V_{REF}/2$ while their top plates sense V_{in} .

Next, S_0 turns off and the comparator is clocked. Since $V_{DAC} = V_{in1} > V_{REF}/2$, the loop reduces V_{DAC} by switching the bottom plate of C_3 from $V_{REF}/2$ to zero. As a result, V_{DAC} falls by $V_{REF}/4 = 2$ LSB [Fig. 14.13(b)]. For this to occur, a logical ONE is written onto FF_3 . At $t = t_1$, the comparator is clocked again, deciding that V_{DAC} must increase, i.e., FF_2 generates a logical ZERO and the bottom plate of C_2 swings to V_{REF} . Finally, at $t = t_2$, the comparator is clocked for a third time, revealing that $V_{DAC} < V_{REF}/2$ and setting FF_1 to a logical ZERO. The loop then delivers an output equal to 100.

Before elaborating on some of the interesting properties of this architecture, we examine its operation for $V_{in2} = 3.7$ LSB as well. Figure 14.13(c) illustrates that the first comparator decision drives V_{DAC} up by 2 LSB, and the second brings it down by 1 LSB. That is, FF_3 holds a ZERO and FF_2 a ONE. At $t = t_2$, the comparator is clocked, writing a ONE to FF_1 . The digital output is thus equal to 011.

The top-plate sampling architecture of Fig. 14.13(a) offers two remarkable properties. First, to resolve N bits, its DAC need settle only $N - 1$ times. Second, the number of unit capacitors in the DAC is equal to 2^{N-1} rather than 2^N . These benefits accrue because the ADC allows *three* possible bottom-plate voltages for the capacitors. One drawback is that the channel charge of S_0 depends on the input level, an effect that is present to some extent even with bootstrapping (Chapter 3).

Example 14.5

Examine the operation of a 4-bit top-plate sampling SAR loop for $V_{in1} = 8.7$ LSB and $V_{in2} = 7.7$ LSB.

Solution

Shown in Fig. 14.14(a), the ADC employs a 3-bit binary-weighted DAC. The loop acts according to the

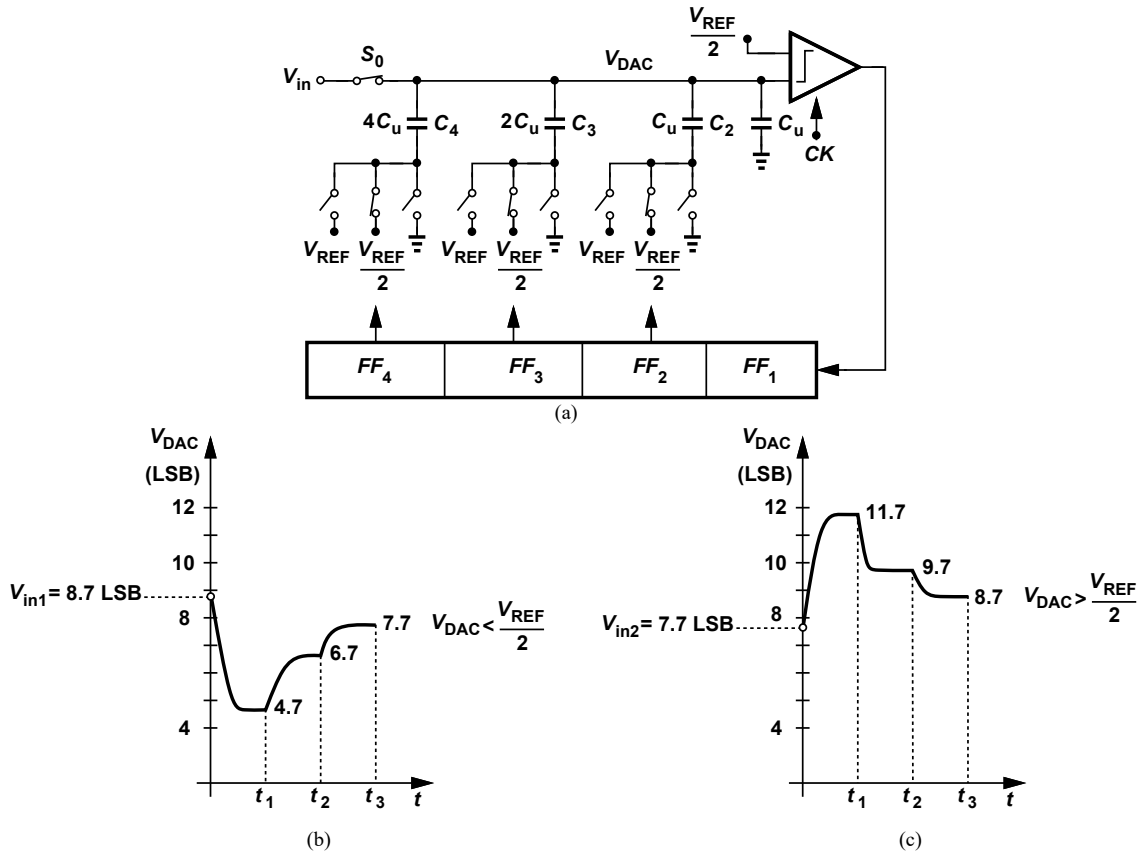


Figure 14.14 (a) SAR ADC with top-plate sampling, (b) DAC waveform for $V_{in1} = 8.7$ LSB, and (c) DAC waveform for $V_{in2} = 7.7$ LSB.

following principles. (1) The MSB capacitor can change V_{DAC} by $\pm V_{REF}/4$; lower-rank capacitors cause proportionally smaller changes. (2) If $V_{DAC} > V_{REF}/2$, the bit is set to 1 and the corresponding bottom plate switches *down*. (3) If $V_{DAC} < V_{REF}/2$, the bit is set to 0 and the corresponding bottom plate switches *up*. (4) For the LSB, the bit is changed but no additional DAC settling is necessary. As S_0 turns off and the circuit enters the conversion mode, the comparator is clocked. Depicted in Fig. 14.14(b) for $V_{in1} = 8.7$ LSB, V_{DAC} jumps down by 4 LSB, and this undershoot is followed by two rising transitions by 2 LSB and 1 LSB. At $t = t_3$, the comparator is clocked for the fourth time, writing a logical ZERO onto FF₁. The digital output is then equal to 1000.

The case of $V_{in2} = 7.7$ LSB is shown in Fig. 14.14(c). The comparator makes a decision and sets the MSB to 0, raising V_{DAC} by 4 LSB. This overshoot is then compensated by successively setting the next two MSBs to 1 and reducing V_{DAC} by 2 LSB and 1 LSB. At $t = t_3$, the comparator decides that $V_{DAC} > V_{REF}/2$, setting FF₁ to 1. The output is equal to 0111.

The top-plate sampling architectures of Figs. 14.13(a) and 14.14(a) merit three remarks. First, their flipflops' contents are set only once. This point stands in contrast to the generic SAR ADC studied in Section 14.1.2, where a given bit must be reset if a DAC overshoot or undershoot has occurred. Second, as explained in the next section, the loop accommodates differential operation with a two-input comparator, whereas the differential flash ADC in Chapter 13 requires a comparator with four inputs. Third, the value of $V_{REF}/2$ must be sufficiently precise, an issue that manifests itself at high resolutions. The following example delves into this aspect, but it also serves as a good exercise for the analysis of SAR imperfections (Chapter 15).

Example 14.6

Suppose the voltage $V_{REF}/2$ in Fig. 14.14(a) is generated from V_{REF} by a resistive divider and incurs an error, $\Delta V = 1.2$ LSB. Explain how the converter's output is affected for $V_{in1} = 8.7$ LSB and $V_{in2} = 7.7$ LSB.

Solution

Shown in Fig. 14.15(a), the ADC creates *unequal* rising and falling transitions in V_{DAC} . Specifically, if MSB = 1 and the bottom plate of C_4 swings from $V_{REF}/2 + \Delta V = 9.2$ LSB to zero, then V_{DAC} changes by 4.6 LSB, but for MSB = 0, the change is equal to $[V_{REF} - (V_{REF}/2 + \Delta V)]/2 = (V_{REF}/2 - \Delta V)/2$, which amounts to 3.4 LSB. For the next MSB, the corresponding jumps are 2.3 LSB and 1.7 LSB, respectively. Finally, for the LSB, V_{DAC} can fall by 1.15 LSB or rise by 0.85 LSB. These results are summarized in Fig. 14.15(b).

We now study the circuit's operation for $V_{in1} = 8.7$ LSB and $V_{in2} = 7.7$ LSB. Sketched in Fig. 14.15(c) for the former, V_{DAC} rises from 8.7 LSB by 3.4 LSB because the comparator decides that the initial value of V_{DAC} is lower than $V_{REF}/2 + \Delta V = 9.2$ LSB. In the next two cycles, V_{DAC} falls by 2.3 LSB and 1.15 LSB. At t_3 , the comparator is clocked, yielding a logical ZERO for the LSB. Thus, $D_{out} = 0110$.

For $V_{in2} = 7.7$ LSB, we note from Fig. 14.15(d) that V_{DAC} rises by 3.4 LSB, falls by 2.3 LSB, and rises by 0.85 LSB, settling at 9.65 LSB. It follows that $D_{out} = 0101$.

We make two observations: (1) the digital outputs in this case are 2 LSB less than those in Fig. 14.14, but (2) as V_{in} increases by 1 LSB, so does D_{out} . We therefore wonder whether the 1.2-LSB error in $V_{REF}/2$ simply shifts the entire input-output characteristic down by 2 LSB, manifesting itself as an offset.

We must perform a careful analysis, but let us consider $V_{in3} = 9.7$ LSB as another example. Illustrated in Fig. 14.15(e), V_{DAC} now falls by 4.6 LSB and then rises by 1.7 LSB and 0.85 LSB, reaching 7.65 LSB. That is, $D_{out} = 1000$. Unfortunately, as the analog input increases from 8.7 LSB to 9.7 LSB, D_{out} changes by 2 LSB, indicating that the characteristic is distorted rather than simply offset.

This example reveals that the 1.2-LSB error in $V_{REF}/2$ gives rise to nonlinearity. To compute the DNL, we must determine the ADC's decision thresholds. For example, we ask, for what range of V_{in} does the converter

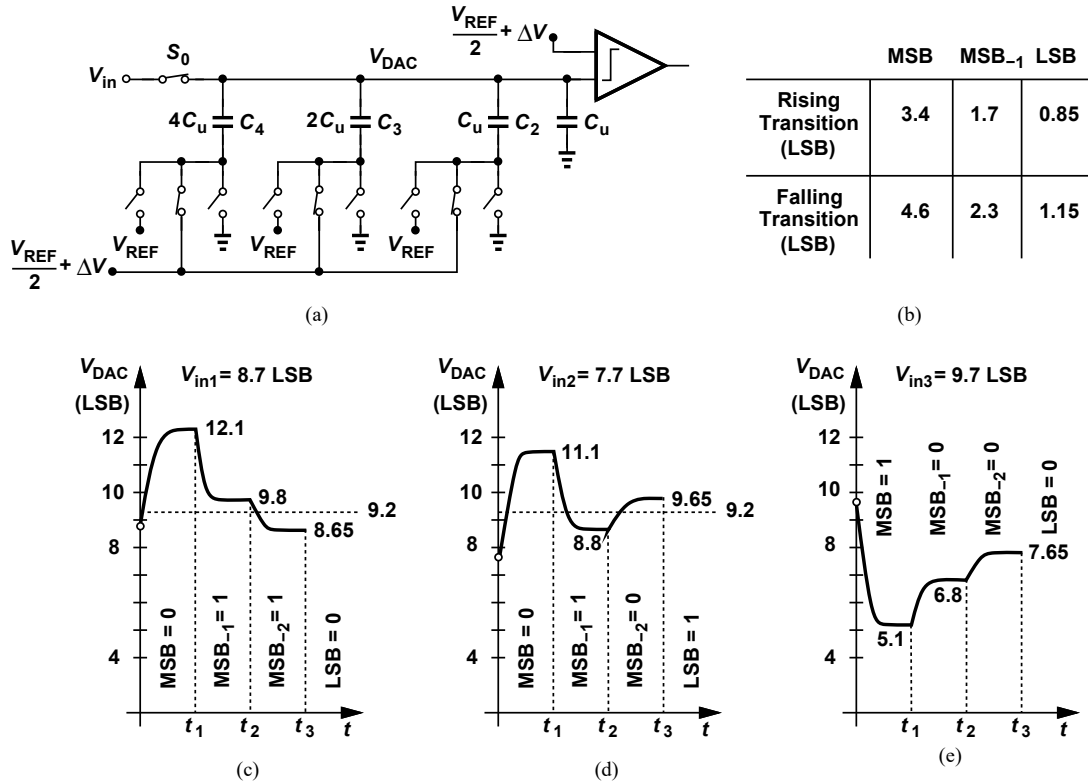


Figure 14.15 (a) SAR ADC with an error in $V_{REF}/2$, (b) DAC voltage changes corresponding to bit transitions, (c) DAC waveform for $V_{in1} = 8.7$ LSB, (d) DAC waveform for $V_{in2} = 7.7$ LSB, and (e) DAC waveform for $V_{in3} = 9.7$ LSB.

deliver $D_{out} = 0101$? This interesting and challenging task is pursued in the next example. The reader is encouraged to review Section 14.1.5 before proceeding here and work through this example patiently.

Example 14.7

For the ADC in Example 14.6, study the input-output characteristic as V_{in} varies from 7 LSB to 10 LSB.

Solution

Figure 14.15(d) reveals that $V_{in2} = 7.7$ LSB translates to $D_{out} = 0101$. By how much can we lower V_{in} while maintaining the same output? The last transition, at $t = t_2$, must guarantee $V_{DAC} > 9.2$ LSB. That is, V_{in} can drop by nearly $9.65 \text{ LSB} - 9.2 \text{ LSB} = 0.45 \text{ LSB}$ [Fig. 14.16(a)]. The input can therefore be as low as $7.7 \text{ LSB} - 0.45 \text{ LSB} = 7.25 \text{ LSB}$. For the upper bound, we observe that the transition after $t = t_1$ must keep V_{DAC} below 9.2 LSB. We can then raise V_{in} by $9.2 \text{ LSB} - 8.8 \text{ LSB} = 0.4 \text{ LSB}$, obtaining a maximum V_{in} of $7.7 \text{ LSB} + 0.4 \text{ LSB} = 8.1 \text{ LSB}$ [Fig. 14.16(b)]. Figure 14.16(c) summarizes these results, implying a DNL of $8.1 \text{ LSB} - 7.25 \text{ LSB} - 1 \text{ LSB} = -0.15 \text{ LSB}$.

We continue our analysis by examining the ADC's behavior for greater values of V_{in} . We predict that $V_{in} = 8.2$ LSB yields $D_{out} = 0110$, as evident from Fig. 14.16(d), where V_{DAC} settles to slightly higher than 9.2 LSB after $t = t_1$. We must now determine the next decision threshold, i.e., the value of V_{in} that gives $D_{out} = 0111$. Let us raise the trajectory in Fig. 14.16(d) so that the LSB becomes a ONE [Fig. 14.16(e)]. Does D_{out} now equal to 0111 ? No, unfortunately, the 1.05 -LSB upward shift lifts the corresponding V_{in} from 8.2 LSB to 9.25 LSB, a value *greater* than the comparator's reference voltage of 9.2 LSB. The black

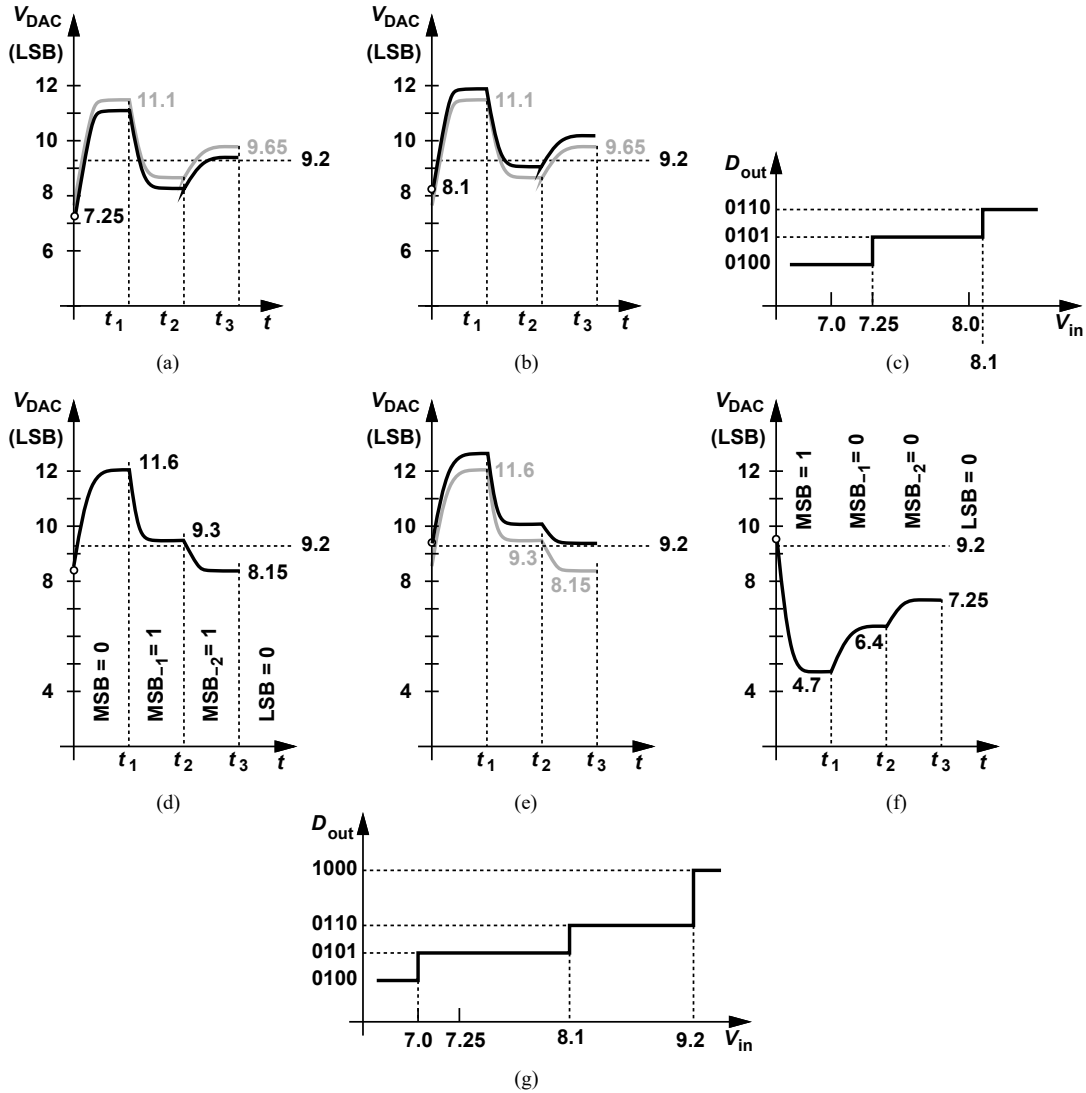


Figure 14.16 DAC voltages and the resulting input-output characteristic.

trajectory is therefore not possible.

Why does this happen? We recognize that the 3.4-LSB rising transition at the beginning is *less* than the sum of the subsequent falling transitions, $2.3 \text{ LSB} + 1.15 \text{ LSB} = 3.45 \text{ LSB}$, disallowing V_{DAC} to begin below 9.2 LSB and finish above it. Thus, no input value yields $D_{out} = 0111$. Let us increase V_{in} slightly above 9.2 LSB and examine the operation. As illustrated in Fig. 14.16(f), for $V_{in} = 9.3 \text{ LSB}$, V_{DAC} falls by 4.6 LSB and rises by 1.7 LSB and 0.85 LSB, generating $D_{out} = 1000$. The input-output characteristic is shown in Fig. 14.16(g), revealing the missing code as V_{in} crosses 9.2 LSB. The reader can prove that the next decision threshold occurs at $V_{in} = 11.25 \text{ LSB}$.

It is instructive to determine the maximum ΔV in Fig. 14.15(a) that avoids a missing code. From the preceding example, we note that the first rising V_{DAC} transition, which is equal to $[V_{REF} - (V_{REF}/2 + \Delta V)]/2 = (V_{REF}/2 - \Delta V)/2$, must be greater than the sum of the subsequent

falling transitions, namely, $(V_{REF}/2 + \Delta V)/4$ and $(V_{REF}/2 + \Delta V)/8$. It follows that

$$\left(\frac{1}{4} - \frac{1}{8} - \frac{1}{16}\right) V_{REF} > \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8}\right) \Delta V \quad (14.1)$$

and hence

$$\Delta V < \frac{V_{REF}}{14}. \quad (14.2)$$

Of course, for higher resolutions, this constraint becomes tighter. As a rule of thumb, we say ΔV must be less than 1 LSB. Fortunately, this issue is avoided in the fully-differential version.

14.2.4 Differential Top-Plate-Sampling ADC

We generally prefer fully-differential operation and must modify the ADC architecture of Fig. 14.14(a) accordingly. Recall that the capacitor bottom plates switch from $V_{REF}/2$ to zero or V_{REF} based on the comparator's decision. Illustrated in Fig. 14.17, the transformation to a differential architecture is fairly straightforward: two capacitor DACs sample the inputs, V_{in}^+ and V_{in}^- , and the bottom plates switch from V_{CM} to V_{REF}^+ or V_{REF}^- . The value of V_{CM} is roughly midway between V_{REF}^+ and V_{REF}^- . In this case, the comparator simply detects the polarity of $V_{DAC}^+ - V_{DAC}^-$.

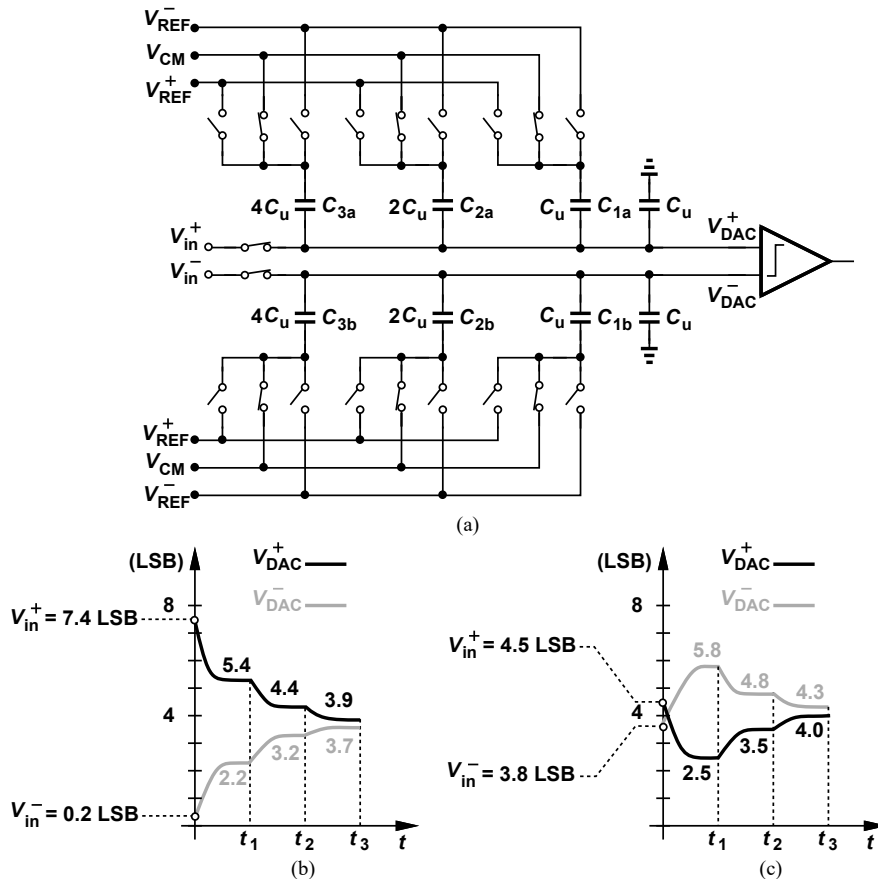


Figure 14.17 (a) Differential ADC with top-plate sampling, (b) DAC voltages for $V_{in}^+ = 7.4$ LSB and $V_{in}^- = 0.2$ LSB, and (c) DAC voltages for $V_{in}^+ = 4.5$ LSB and $V_{in}^- = 3.8$ LSB.

It is important to recall from our study of fully-differential ADCs in Chapter 13 that the differential input of this circuit, $V_{in}^+ - V_{in}^-$, ranges from $-(V_{REF}^+ - V_{REF}^-)$ to $+(V_{REF}^+ - V_{REF}^-)$. For 4 bits of resolution, therefore, we have $V_{REF}^+ - V_{REF}^- = 8 \text{ LSB}$ (rather than 16 LSB).

Let us analyze the 4-bit ADC for $V_{in}^+ - V_{in}^- = 7.2 \text{ LSB}$, assuming that V_{in}^+ and V_{in}^- swing around a common-mode near V_{CM} , and $V_{CM} = (V_{REF}^+ - V_{REF}^-)/2$. Upon entering the conversion mode, the loop decides to decrease this difference. Thus, the bottom plate of C_{3a} swings from V_{CM} to V_{REF}^- and that of C_{3b} to V_{REF}^+ . Consequently, V_{DAC}^+ changes by $(V_{REF}^- - V_{CM})/2$ and V_{DAC}^- by $(V_{REF}^+ - V_{CM})/2$; i.e., $V_{DAC}^+ - V_{DAC}^-$ drops by $(V_{REF}^+ - V_{REF}^-)/2 = 4 \text{ LSB}$ [Fig. 14.17(b)].

Next, the comparator forces the bottom plates of C_{2a} and C_{2b} to switch properly, causing V_{DAC}^+ to fall by 1 LSB and V_{DAC}^- to rise by 1 LSB. At $t = t_2$, C_{1a} and C_{1b} come into action and reduce $V_{DAC}^+ - V_{DAC}^-$ by 1 LSB. This difference is still positive at $t = t_3$, the comparator is clocked once more, and we have $D_{out} = 1111$.

As another example, suppose $V_{in}^+ - V_{in}^- = 0.7 \text{ LSB}$. Plotted in Fig. 14.17(c), the V_{DAC}^+ and V_{DAC}^- trajectories first exhibit 2-LSB jumps in opposite directions, yielding a difference equal to -3.3 LSB . The loop thus attempts to correct this overshoot by lowering V_{DAC}^- and raising V_{DAC}^+ by 1 LSB and then by 0.5 LSB. We now have $D_{out} = 1000$.

It is interesting to note that the changes in $V_{DAC}^+ - V_{DAC}^-$ in Fig. 14.17 are independent of V_{CM} , a point of contrast to the operation of the single-ended ADC in Fig. 14.14(a). Thus, V_{CM} need not be very precise. Nevertheless, generating V_{CM} consumes power. Depicted in Fig. 14.18 is an example corresponding to Fig. 14.17(a). Do R_1 and R_2 affect the acquisition time? If V_{in}^+ and V_{in}^- vary differentially and $R_1 = R_2$, then

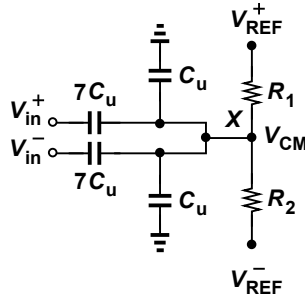


Figure 14.18 Simple resistive network for CM generation.

node X is a virtual ground and the resistors do not limit the sampling process. Thus, R_1 and R_2 can be fairly large.

14.2.5 Bottom-Plate-Sampling ADC

It is possible to apply the SAR ADC input to the *bottom* plates of the DAC capacitors in the sampling mode. This method provides certain advantages over the top-plate sampling architecture studied in the previous sections.

Consider the 3-bit system shown in Fig. 14.19(a), where the bottom plates of C_0 - C_3 are tied to V_{in} while S_T is on. The loop behaves in a manner similar to the generic SAR ADC analyzed in Section 14.1.2: the MSB in the register is set to 1, the DAC settles, the comparator is clocked, the next bit is set to 1, and, in the case of DAC overshoot or undershoot, the MSB is reset to 0.

We examine the circuit's operation by noting that, at the end of the sampling mode, S_{3a} - S_{0a} and S_T turn off, S_{3b} turns on (because the MSB is set to 1), and S_{2c} - S_{0c} also turn on (because their corresponding bits are set to 0). Suppose the sampled input is equal to 4.7 LSB. From the equivalent circuit depicted in Fig. 14.19(b), we have, with the aid of superposition:

$$V_{DAC} = \frac{V_{REF} - V_{in}}{2} - \frac{V_{in}}{2} \quad (14.3)$$

$$= \frac{V_{REF}}{2} - V_{in}. \quad (14.4)$$

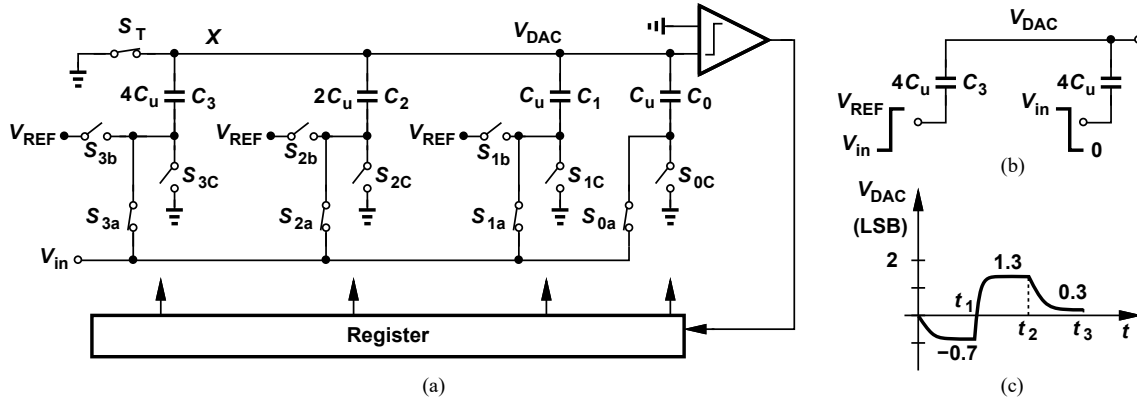


Figure 14.19 (a) SAR ADC with bottom-plate sampling, (b) simplified DAC, and (c) DAC waveform example.

The DAC output can thus reveal whether $V_{in} > V_{REF}/2$ or $V_{in} < V_{REF}/2$. As illustrated in Fig. 14.19(c), V_{DAC} settles to -0.7 LSB if $V_{in} = 4.7$ LSB and $V_{REF} = 8$ LSB. It is important to observe that, from this time on, C_2 and C_1 can only *raise* V_{DAC} because their bottom plates reside at zero and can only swing to V_{REF} . That is, if V_{DAC} is to decrease, then the previous bit must be reset to 0. The comparator is clocked at t_1 , MSB₋₁ is set to 1, and the bottom plate of C_2 switches to V_{REF} , allowing V_{DAC} to rise by 2 LSB. This is followed by a comparison at t_2 , a 1-LSB fall in V_{DAC} ,⁴ and another comparison at t_3 , which indicates the LSB must be set to 0. We therefore have $D_{out} = 100$. We recognize that the comparator is clocked three times and V_{DAC} also settles three times—a point of contrast to the top-plate-sampling architecture in Fig. 14.13(a). Note that C_0 switches only from V_{in} to zero.

Example 14.8

Repeat the foregoing analysis if $V_{in} = 1.7$ LSB.

Solution

From Eq. (14.4), we have $V_{DAC} = +2.3$ LSB after the bottom plate of C_3 in Fig. 14.19(a) swings to V_{REF} (Fig. 14.20). The comparator is clocked at $t = t_1$, the MSB is set to 0, the bottom plate of C_3 switches to ground, and MSB₋₁ is set to 1, causing V_{DAC} to fall by 2 LSB. This is followed by a comparison at t_2 , a 1-LSB fall in V_{DAC} , and the last comparison at t_3 . We obtain $D_{out} = 001$.

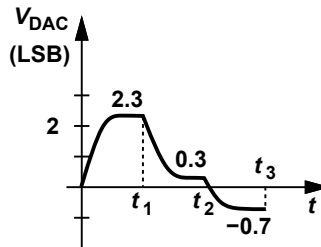


Figure 14.20 DAC waveform for $V_{in} = 1.7$ LSB.

⁴Note that this fall is obtained by setting MSB₋₁ to 0 and raising the next bit to 1.

It is instructive to contrast the SAR operation in top-plate and bottom-plate sampling loops. The former begins by clocking the comparator, thereby saving one DAC settling. The latter cannot because it presents a zero difference to the comparator as it enters the conversion mode. Also, the former requires an $(N - 1)$ -bit DAC for an N -bit ADC. On the other hand, bottom-plate sampling offers two advantages. First, it does not require an additional reference voltage [V_{CM} in Fig. 14.17(a)]. Second, it is much less susceptible to the sampling switch imperfections. This point is studied in detail below as it proves useful in other ADC architectures as well (Chapter 17).

Charge Injection in Bottom-Plate Samplers We wish to show that bottom-plate sampling in SAR ADCs is free of input-dependent charge injection. This discussion is somewhat complex and demands that we proceed in small steps. As the reader will see, the “top” and “bottom” plates of capacitors prove confusing here. Let us begin with the simple circuit in Fig. 14.21(a), where V_{in} changes from V_1 to V_2 . We note that the charge on C_a varies from $C_a V_1$ to $C_a V_2$, with no dependence upon the values of V_{in} between t_1 and t_2 . Next, we

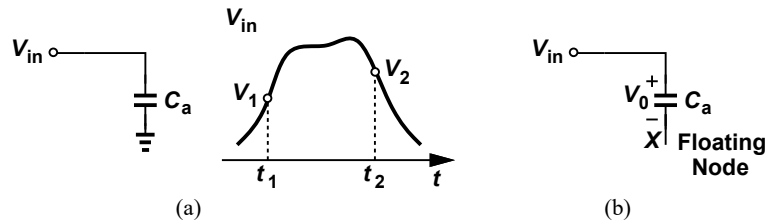


Figure 14.21 (a) A capacitor sensing a time-varying voltage, and (b) illustration of floating bottom plate.

turn to the situation in Fig. 14.21(b), where C_a has an initial condition of V_0 but its right plate is not attached to any other node and is considered “floating.” We observe that, as V_{in} changes with time, the charge on this plate remains equal to $-C_a V_0$ because no path exists to allow the flow of charge onto this plate. Thus, the left plate too maintains a constant amount of charge equal to $+C_a V_0$ even if V_{in} varies.

Now, consider the basic sampling circuit shown in Fig. 14.22(a) and recall from Chapter 3 that the input-dependent channel charge of M_1 , Q_{ch} , introduces distortion in the voltage held on C_a . We presented the

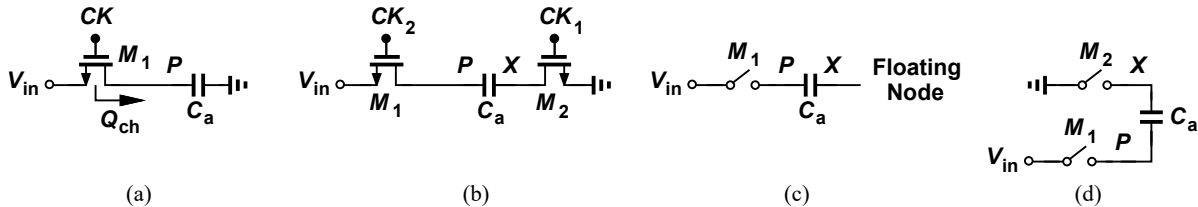


Figure 14.22 (a) Effect of switch charge injection, (b) use of a switch on the right plate, (c) equivalent circuit while M_1 turns off, and (d) alternate drawing of the circuit.

concept of bootstrapping as a method of minimizing this dependence, but observed that the threshold variation of M_1 still translates to some nonlinear input dependence according to $Q_{ch} = (WL)_1 C_{ox} (V_{GS} - V_{TH1})$. It is possible to avoid this issue through the use of “bottom-plate” sampling and a proper clocking sequence. To this end, let us insert another switch in series with the right plate of C_a [Fig. 14.22(b)]. In the sampling mode, both are on; at the end, both turn off. In a well-designed circuit, $V_X \approx 0$ near the end of the acquisition mode so that V_{in} is impressed across C_a accurately. But the key concept behind “bottom-plate” sampling is that M_2 turns off *before* M_1 does. This sequence avoids the problem of input-dependent charge injection, a point that we will prove by means of some observations. We assume the clocks swing between zero and V_{DD} and the switches are realized by NMOS devices.

Here are the observations. First, the channel charge of M_1 depends on V_{in} , but that of M_2 does not because $V_X \approx 0$. Second, as M_2 turns off, it injects some of its channel charge, $(WL)_2 C_{ox} (V_{DD} - V_{TH2})$, onto C_a , creating a constant voltage change, i.e., an offset. We disregard this offset for now. Third, after M_2 turns off,

the right plate of C_a floats, and its charge cannot change [Fig. 14.22(c)]; thus, the charge on the left plate cannot change, either. We then surmise that none of the channel charge of M_1 can enter C_a , and hence the sampled voltage is free from input-dependent charge injection.

We articulate “bottom-plate” sampling as a circuit in which the switch that is *not* connected to V_{in} turns off first. We hereafter draw the circuit as shown in Fig. 14.22(d) and call it a “bottom-plate” sampler if M_2 turns off before M_1 does.

The reader may find the third observation not quite convincing: if M_1 turns off very rapidly, how does the channel charge “know” not to exit from the terminal on the right?! To avoid this quandary, we take into account the parasitic capacitance at node X , e.g., that due to M_2 in Fig. 14.22(b), and reexamine the operation. As illustrated in Fig. 14.23(a), this capacitance, C_X , simply appears in series with C_a and allows the charge

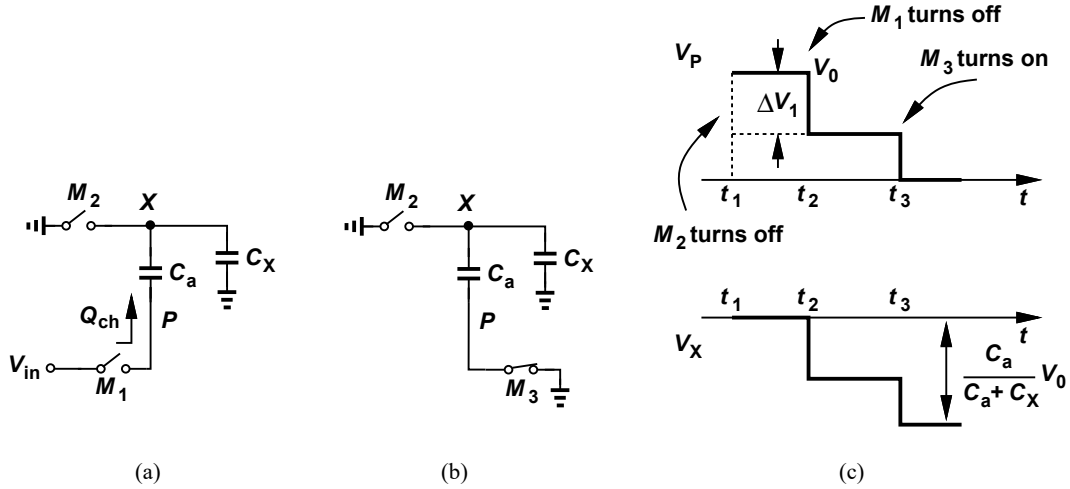


Figure 14.23 (a) Charge injection in presence of parasitic C_X , (b) the circuit with the bottom plate grounded, and (c) the voltage waveforms.

on both plates of C_a to change. That is, some of the channel charge of M_1 *does* enter C_a . Nonetheless, we recall that the capacitor bottom plates in the architecture of Fig. 14.19(a) eventually switch to another voltage, e.g., to ground [Fig. 14.23(b)]. Let us follow the voltages at nodes P and X carefully, neglecting the charge injected by M_2 for simplicity. As illustrated in Fig. 14.23(c), M_2 turns off at $t = t_1$, when $V_{in} = V_0$. Next, at $t = t_2$, M_1 turns off, its channel charge creating a step, ΔV_1 , in V_P . Finally, at $t = t_3$, M_3 turns on to bring P to zero. Is the change in V_X affected by the channel charge of M_1 ? Since V_P has traveled from V_0 to zero,

$$V_X = -\frac{C_a}{C_a + C_X} V_0. \quad (14.5)$$

What exactly happens to the charge injected by M_1 ? It is sent to ground when M_3 turns on.

We summarize the preceding principle as follows: the channel charge of M_1 plays no role in the circuit because the bottom plate goes from one driven point (V_{in}) to another (ground) while there is no path to change the total charge at node X . The term “driven point” means that the bottom plate is driven by a voltage source in both cases. The top plate, on the other hand, “floats” (has a high impedance) after M_1 turns off.

We now relate the foregoing development to the bottom-plate-sampling ADC of Fig. 14.19(a). Figure 14.24 shows the correspondence between the two. Note P is “driven” and node X floats after M_2 turns off.

We should remark that the “bottom plate” does not necessarily refer to that plate of a capacitor which is *drawn* on the bottom. Recall from Chapter 2 that some capacitor structures exhibit a greater parasitic associated with one of their plates; that plate can be considered the “bottom” one (Definition 1). More fundamentally, in our studies here, the bottom plate refers to that which is tied to the input voltage during sampling and released *second* as the circuit begins to enter the hold mode (Definition 2). Given that C_X in Fig. 14.24 must be minimized, we conclude that the two definitions collapse into one, i.e., the terminal having a greater parasitic preferably senses the input rather than serves as the top plate.

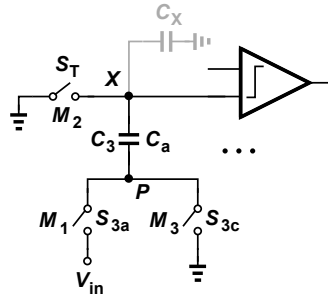


Figure 14.24 Illustration of driven and floating nodes.

14.2.6 Differential Bottom-Plate-Sampling ADC

The single-ended SAR architecture of Fig. 14.19(a) can be modified to accommodate differential input signals. Before delving into the details, we should recall the binary search algorithm and note that, while the loop in Fig. 14.2(a) first compares V_{in} to $V_{REF}/2$ to determine the MSB, differential operation requires that we compare $V_{in}^+ - V_{in}^-$ to zero. After this result is obtained, we compare the input difference to $+(V_{REF}^+ - V_{REF}^-)/2$ or $-(V_{REF}^+ - V_{REF}^-)/2$, etc.

Shown in Fig. 14.25(a) is a 3-bit differential realization. The bottom plates can connect to the analog input, the reference, or the common-mode level, V_{CM} . The CM level of the analog inputs is also equal to V_{CM} . In the sampling mode, the top plates⁵ tie to V_{CM} and the bottom plates to their respective analog inputs. Two observations prove helpful here. First, neglecting parasitic capacitances, we note that, if the bottom plates of all C_a capacitors swing by ΔV , so does V_{DAC}^+ . Second, if a bit is set to 1, then the corresponding bottom plates in the top and bottom DACs switch to V_{REF}^+ and V_{REF}^- , respectively.

Let us examine the ADC's operation by assuming $V_{REF}^+ = 900$ mV, $V_{REF}^- = 100$ mV, $V_{CM} \approx (V_{REF}^+ + V_{REF}^-)/2 = 500$ mV, $V_{in}^+ = 740$ mV, and $V_{in}^- = 260$ mV. The analog LSB is equal to 200 mV because $V_{in}^+ - V_{in}^-$ ranges from -800 mV to $+800$ mV; hence $V_{in}^+ - V_{in}^- = +2.4$ LSB. We expect the loop to compare $V_{in}^+ - V_{in}^-$ with zero, then with $(V_{REF}^+ - V_{REF}^-)/2$, and finally with $3(V_{REF}^+ - V_{REF}^-)/4$ (why?). Upon entering the conversion mode, the ADC switches all of the bottom plates to V_{CM} , yielding

$$V_{DAC}^+ = V_{CM} + (V_{CM} - V_{in}^+) \quad (14.6)$$

$$= 2V_{CM} - V_{in}^+ = 260 \text{ mV} \quad (14.7)$$

and

$$V_{DAC}^- = 2V_{CM} - V_{in}^- = 740 \text{ mV}. \quad (14.8)$$

That is, $V_{DAC}^+ - V_{DAC}^- = -(V_{in}^+ - V_{in}^-)$ [Fig. 14.25(b)]. The comparator is clocked at $t = t_1$, deciding that V_{DAC}^- must fall and V_{DAC}^+ must rise. The loop then sets the MSB to 1 so that the bottom plates of C_{3a} and C_{3b} swing to V_{REF}^+ and V_{REF}^- , respectively. That is, V_{DAC}^+ rises by 1 LSB = 200 mV and V_{DAC}^- falls by the same amount.

Figure 14.25(c) summarizes the switching actions thus far. Using superposition, we have

$$V_{DAC}^+ = 2V_{CM} - V_{in}^+ + \frac{V_{REF}^+ - V_{CM}}{2} \quad (14.9)$$

$$= 3\frac{V_{CM}}{2} + \frac{V_{REF}^+}{2} - V_{in}^+ \quad (14.10)$$

$$= 460 \text{ mV} \quad (14.11)$$

⁵The “top-plates” of the upper DAC capacitors are drawn on the bottom (!), but we so call them because they are not “driven.”

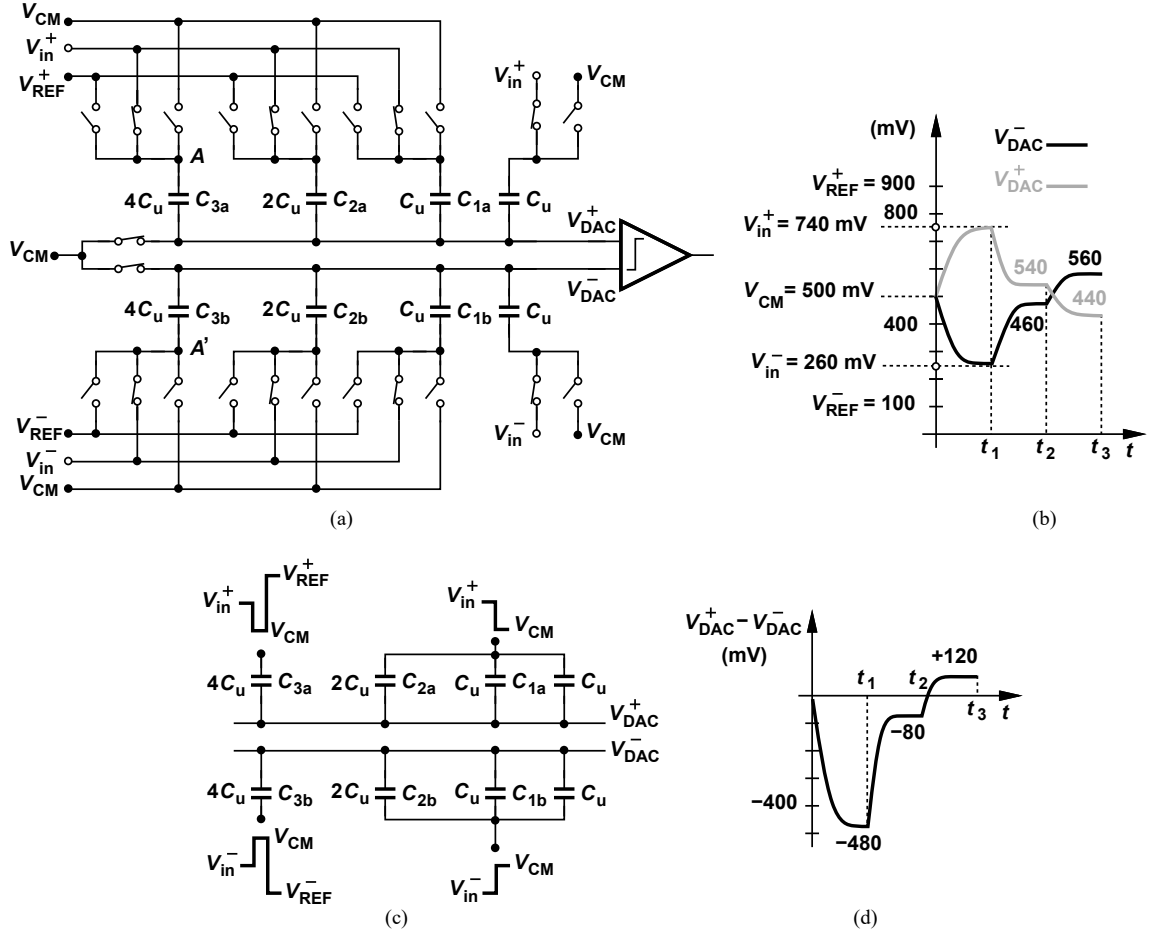


Figure 14.25 (a) Differential SAR ADC with bottom-plate sampling, (b) DAC voltages for $V_{in}^+ = 740$ mV and $V_{in}^- = 260$ mV, (c) illustration of bottom-plate voltage changes, and (d) DAC differential voltage waveform.

and

$$V_{DAC}^- = 3 \frac{V_{CM}}{2} + \frac{V_{REF}^-}{2} - V_{in}^- \quad (14.12)$$

$$= 540 \text{ mV}. \quad (14.13)$$

As expected, $V_{DAC}^+ - V_{DAC}^-$ changes by 2 LSB, reaching $(V_{REF}^+ - V_{REF}^-)/2 - (V_{in}^+ - V_{in}^-) = -80$ mV. The next decision occurs at $t = t_2$ and aims to further reduce this difference. Since the bottom plates of C_{2a} and C_{2b} swing from V_{CM} to V_{REF}^+ and V_{REF}^- , respectively, we simply add $(V_{REF}^+ - V_{CM})/4$ to (14.10) and $(V_{REF}^- - V_{CM})/4$ to (14.12). It follows that

$$V_{DAC}^+ = \frac{5}{4} V_{CM} + \frac{3}{4} V_{REF}^+ - V_{in}^+ = 560 \text{ mV} \quad (14.14)$$

and

$$V_{DAC}^- = \frac{5}{4} V_{CM} + \frac{3}{4} V_{REF}^- - V_{in}^- = 440 \text{ mV}. \quad (14.15)$$

In other words, $V_{DAC}^+ - V_{DAC}^-$ is raised by 1 LSB = 200 mV. At $t = t_3$, the comparator is clocked once more, concluding that the LSB must be equal to 0. Thus, $D_{out} = 110$. Figure 14.25(d) plots $V_{DAC}^+ - V_{DAC}^-$.

Example 14.9

Must the value of V_{CM} in Fig. 14.25(a) be precisely equal to $(V_{REF}^+ + V_{REF}^-)/2$?

Solution

No, it need not be. We observe from the foregoing results that $V_{DAC}^+ - V_{DAC}^-$ remains independent of V_{CM} . The reader is encouraged to study how the common-mode level of V_{DAC}^+ and V_{DAC}^- is affected if V_{CM} departs from $(V_{REF}^+ + V_{REF}^-)/2$.

Our architecture design in Fig. 14.25(a) proves incomplete: if the bottom plates of C_{3a} , C_{2a} , and C_{1a} can swing only from V_{CM} to V_{REF}^+ , then they cannot *reduce* V_{DAC}^+ . This situation arises if, for example, $V_{in}^+ = 260$ mV and $V_{in}^- = 740$ mV. We predict that the ADC should compare $V_{in}^+ - V_{in}^- = -480$ mV with zero, with $(V_{REF}^- - V_{REF}^+)/2 = -400$ mV, and with $3(V_{REF}^- - V_{REF}^+)/2 = -600$ mV in successive cycles. After the bottom plates swing to V_{CM} , we have $V_{DAC}^+ = 2V_{CM} - V_{in}^+ = 740$ mV and $V_{DAC}^- = 2V_{CM} - V_{in}^- = 260$ mV. Since $V_{DAC}^+ - V_{DAC}^- > 0$, the comparator decision at $t = t_1$ sets the MSB to 0, requiring that V_{DAC}^+ fall and V_{DAC}^- rise. But this is not possible in Fig. 14.25(a). We must therefore modify the architecture as shown in Fig. 14.26(a), where the bottom plates of $C_{3a,b}$, $C_{2a,b}$, and

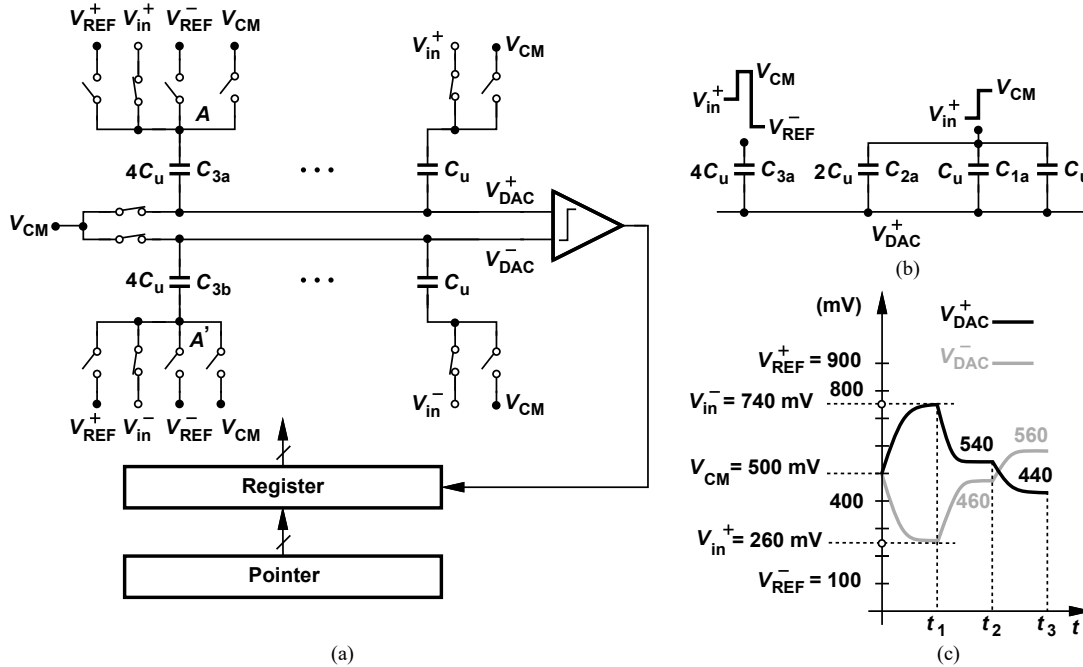


Figure 14.26 (a) Modified differential ADC with bottom-plate sampling, (b) simplified DAC, and (c) DAC waveforms.

$C_{1a,b}$ can connect to one of *four* voltages: the analog input, V_{REF}^+ , V_{REF}^- , and V_{CM} . We recognize that, in this example, the bottom plates of C_{3a} and C_{3b} must switch to V_{REF}^- and V_{REF}^+ , respectively, yielding from Fig. 14.26(b)

$$V_{DAC}^+ = V_{CM} + \frac{V_{REF}^- - V_{in}^+}{2} + \frac{V_{CM} - V_{in}^+}{2} \quad (14.16)$$

$$= \frac{3}{2}V_{CM} + \frac{V_{REF}^-}{2} - V_{in}^+ = 540 \text{ mV} \quad (14.17)$$

and

$$V_{DAC}^- = \frac{3}{2}V_{CM} + \frac{V_{REF}^+}{2} - V_{in}^- = 460 \text{ mV}. \quad (14.18)$$

That is, $V_{DAC}^+ - V_{DAC}^- = -(V_{REF}^+ - V_{REF}^-)/2 - (V_{in}^+ - V_{in}^-) = +80 \text{ mV}$, as desired [Fig. 14.26(c)]. The comparator decides at $t = t_2$ that this difference must further decrease, setting the next MSB to 0 and allowing the bottom plate of C_{2a} to swing to V_{REF}^- and that of C_{2b} to V_{REF}^+ . Thus, V_{DAC}^+ experiences a change of $(V_{REF}^- - V_{CM})/4 = -100 \text{ mV}$ and V_{DAC}^- a change of $(V_{REF}^+ - V_{CM})/4 = +100 \text{ mV}$. The decision at $t = t_3$ sets the LSB to 1, producing $D_{out} = 001$.

In summary, the ADC of Fig. 14.26(a) sets the bit of interest to 0 if $V_{DAC}^+ - V_{DAC}^- > 0$ and switches the bottom plate of the upper DAC capacitor to V_{REF}^- and that of the lower DAC capacitor to V_{REF}^+ . For $V_{DAC}^+ - V_{DAC}^- < 0$, the bit is set to 1 and the bottom plates swing to V_{REF}^+ and V_{REF}^- , respectively.

Example 14.10

As explained in Section 14.1.5, it is instructive to work back from the final DAC output value and obtain the corresponding digital output. Suppose in Fig. 14.26(a), we have $V_{DAC}^+ = +510 \text{ mV}$ and $V_{DAC}^- = +490 \text{ mV}$ after the second comparison [e.g., after $t = t_2$ in Fig. 14.26(c)]. Determine all possible digital codes for this situation.

Solution

The second decision compares $V_{in}^+ - V_{in}^-$ to $+(V_{REF}^+ - V_{REF}^-)/4$, $+3(V_{REF}^+ - V_{REF}^-)/4$, $-(V_{REF}^+ - V_{REF}^-)/4$, or $-3(V_{REF}^+ - V_{REF}^-)/4$. That is, the final differential output of the DAC is given by either

$$V_{DAC}^+ - V_{DAC}^- = \pm \frac{V_{REF}^+ - V_{REF}^-}{4} - (V_{in}^+ - V_{in}^-) \quad (14.19)$$

or

$$V_{DAC}^+ - V_{DAC}^- = \pm \frac{3(V_{REF}^+ - V_{REF}^-)}{4} - (V_{in}^+ - V_{in}^-). \quad (14.20)$$

Since $V_{DAC}^+ - V_{DAC}^- = +20 \text{ mV}$, we have

$$V_{in}^+ - V_{in}^- = +190 \text{ mV}, -210 \text{ mV}, +590 \text{ mV}, -610 \text{ mV}. \quad (14.21)$$

From Fig. 14.27, we have

$$D_{out} = 100, 010, 110, 000. \quad (14.22)$$

The reader can construct the DAC output trajectories for these cases.

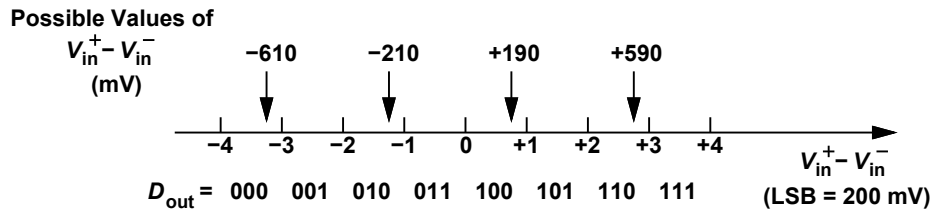


Figure 14.27 Possible input voltages for a given $V_{DAC}^+ - V_{DAC}^-$.

SAR Logic The need for connecting the capacitor bottom plates in Fig. 14.26(a) to four voltages suggests greater digital complexity. Let us ponder this issue and design the logic.

We make three observations. First, when the loop enters the conversion mode, the pointer and the register must remain reset and the bottom plates must connect to V_{CM} . Second, when the comparator is clocked, the leftmost bit in the pointer is raised to 1, and the switches tied to V_{CM} are disabled for the remainder of the binary search. Third, depending on the comparator's decision, the MSB in the register is set to 1 or 0, node A switches to V_{REF}^+ or V_{REF}^- , and node A' connects to V_{REF}^- or V_{REF}^+ . We assume the ADC sampling clock, CK , goes low in the conversion mode.

Figure 14.28(a) depicts the control logic for the CM switches. An RS latch driven by the pointer's leftmost bit, B_n , begins with a high output so that, when $\overline{CK}$ goes high, the CM switch turns on. The comparator is then clocked and, at the same time, B_n is raised to 1. The RS latch thus provides a zero output even after B_n falls to 0 (in the next SAR cycle). Next, we design the logic for switching the bottom plates to V_{REF}^+

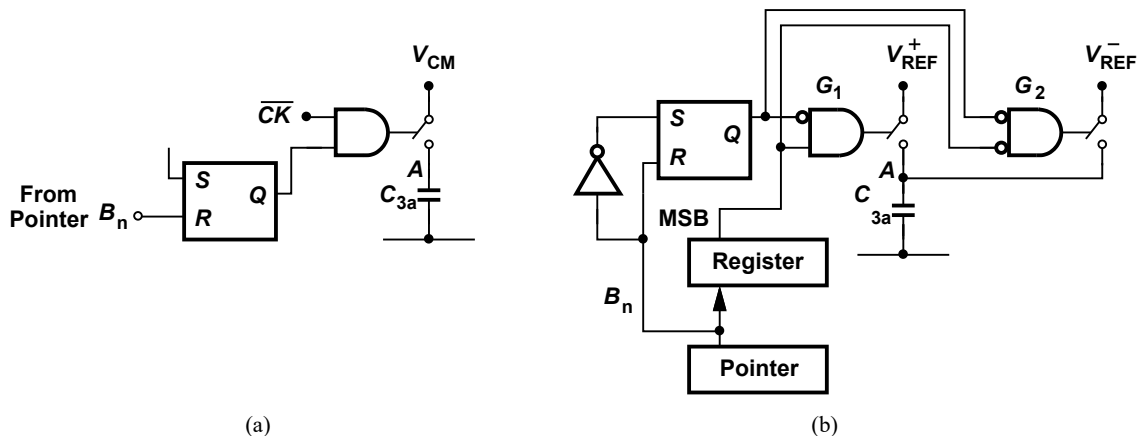


Figure 14.28 (a) Simple control logic for CM switches, and (b) control logic for reference switches.

or V_{REF}^- according to the comparator's decision and hence the corresponding bit in the register. The circuit must be activated only when the CM switches turn off, i.e., when the RS latch output goes low. Illustrated in Fig. 14.28(b), the logic simply consists of AND gates G_1 and G_2 . If $MSB = 1$, then G_1 generates a high output, tying A to V_{REF}^+ . Conversely, if $MSB = 0$, G_2 connects A to V_{REF}^- . The same arrangement can be applied to the other bits. The ADC can employ a single RS latch for this control.

14.3 Clock Generation

We have seen in this chapter that the SAR cycle time is a fraction of the conversion period. The ADC therefore requires two different clock frequencies, one for sampling and another for SAR cycles. For example, suppose the top-plate-sampling architecture studied in Section 14.2.4 is designed for a resolution of N bits and runs at a conversion rate of f_S . This type of loop requires $N - 1$ cycles. We also allocate a sampling time equal to one SAR cycle, thereby arriving at the clocking scheme shown in Fig. 14.29(a). Here, CK_S serves as the sampling clock and is high for $T_1 = T_S/N$, where $T_S = 1/f_S$. Also, CK_C controls the loop and provides $N - 1$ cycles equal to T_1 . At $t = t_a$, the DAC enters the hold mode and the ADC begins to convert. How do we generate CK_S and CK_C in Fig. 14.29(a)? We begin with a clock, CK_{in} , at a frequency of Nf_S [Fig. 14.29(b)] and apply it to a $\div N$ circuit (a counter), which divides the frequency by a factor of N . As explained below, this divider is designed such that it produces one pulse of width T_1 for every N input cycles. The result can thus serve as CK_S . The conversion clock, CK_C , is generated by applying CK_{in} and CK_S to a NAND gate [Fig. 14.29(c)].

The frequency divider in Fig. 14.29(b) must deliver an output pulse width equal to one input period. This is accomplished through the use of a "ring counter," shown in Fig. 14.30. The circuit consists of N flipflops placed in a loop. Upon power-up, one FF output is set to a logical ONE and the rest to a logical ZERO. Then,

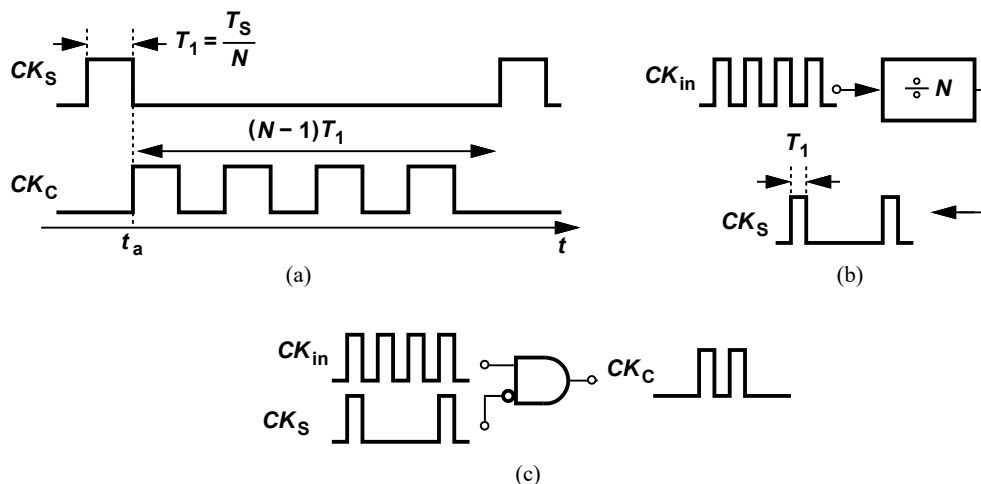


Figure 14.29 (a) Sampling and conversion clocks, (b) use of a divide-by- N circuit, and (c) use of a NAND to generate CK_C .

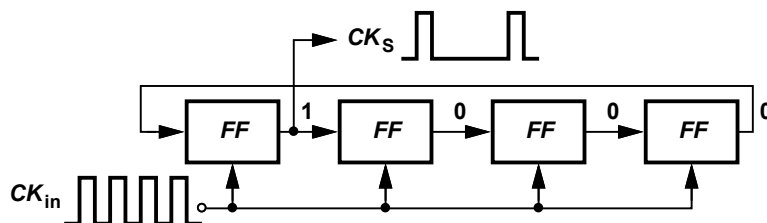


Figure 14.30 Use of a ring counter for generating nonoverlapping phases.

as the flipflops are clocked, the sequence 1000 shifts around the loop. That is, CK_S remains high for one input cycle and low for $N - 1$ input cycles.

The need for generating a clock at Nf_S makes conventional SAR architectures less attractive. For example, a 10-bit 1-GHz ADC would require a 10-GHz clock with a sufficiently low jitter. This issue is alleviated by two techniques, namely, “asynchronous” SAR operation (Chapter 16) and time-interleaving (Chapter 20).

Problems

- 14.1. The SAR ADC of Fig. 14.2(b) is designed for a resolution of 4 bits. If the sampled analog input is equal to $V_{in1} = 7.6$ LSB or $V_{in2} = 8.6$ LSB, sketch V_{DAC} as a function of time. Verify that the register output correctly represents the input in each case.
- 14.2. Plot the residue, $V_{in} - V_{DAC}$, for the two cases in the previous problem.
- 14.3. Suppose the comparator in the ADC studied in Problem 14.1 incurs an offset equal to 0.7 LSB. Sketch V_{DAC} for $V_{in1} = 7.6$ LSB and $V_{in2} = 8.6$ LSB.
- 14.4. The SAR ADC of Fig. 14.6(a) is designed for a resolution of 4 bits. If $V_{in} = 2.3$ LSB or 12.3 LSB, determine the contents of the pointer and the register as a function of time.
- 14.5. Repeat the previous problem if the comparator exhibits an offset of 0.7 LSB.
- 14.6. Determine V_X in Fig. 14.10(c) if the bottom plates of both $4C_u$ and $2C_u$ swing from V_{REF} to zero.
- 14.7. A student accidentally omits the rightmost unit capacitor in Fig. 14.10(c). Determine V_X after the bottom plate of $4C_u$ swings from V_{REF} to zero.
- 14.8. If V_{in} in Fig. 14.11(a) is equal to 2.3 LSB or 5.3 LSB, sketch V_{DAC} as a function of time. Determine the final contents of the SAR register.
- 14.9. Repeat the previous problem if C_0 is accidentally omitted.
- 14.10. Sketch V_{DAC} in Fig. 14.13 if $V_{in} = 2.3$ LSB or 5.3 LSB. Determine the final contents of the flipflops.
- 14.11. Suppose the upper input terminal of the comparator in Fig. 14.13(a) is tied to $V_{REF}/2 + \Delta V$ rather than to V_{REF} . If the capacitor bottom plates still swing from $V_{REF}/2$ to zero or V_{REF} , sketch V_{DAC} as a function of time for $V_{in1} = 4.7$ LSB and $V_{in2} = 3.7$ LSB.
- 14.12. In the SAR ADC of Fig. 14.13(a), C_3 is accidentally chosen equal to C_u rather than $2C_u$. Sketch V_{DAC} for $V_{in1} = 4.7$ LSB and $V_{in2} = 3.7$ LSB. Determine the final contents of the flipflops in each case.
- 14.13. Sketch V_{DAC}^+ and V_{DAC}^- in the ADC of Fig. 14.17(a) if $V_{in}^+ - V_{in}^- = -7.2$ LSB or -0.7 LSB.
- 14.14. Sketch V_{DAC} in Fig. 14.19(a) if the sampled input is equal to 6.7 LSB. Determine the final contents of the register.
- 14.15. Repeat the previous problem if the comparator exhibits an offset equal to 0.2 LSB.
- 14.16. Redraw the waveforms in Fig. 14.23(c) if a parasitic capacitance, C_P , is added from P to ground.
- 14.17. Sketch V_{DAC}^+ and V_{DAC}^- in Fig. 14.25(a) if $V_{REF}^+ = 900$ mV, $V_{REF}^- = 100$ mV, $V_{CM} \approx 500$ mV, $V_{in}^+ = 660$ mV, and $V_{in}^- = 340$ mV.
- 14.18. Repeat the previous problem if the comparator has an offset of 0.3 LSB.
- 14.19. Repeat the analysis illustrated in Fig. 14.26 if $V_{in}^+ = 360$ mV and $V_{in}^- = 640$ mV.
- 14.20. Suppose the DAC voltages in Fig. 14.26(a) have settled to $V_{DAC}^+ = 490$ mV and $V_{DAC}^- = 510$ mV. Determine all possible digital codes for this situation.

SAR ADC Nonidealities

With our fundamental understanding of SAR ADCs established in Chapter 14, we now focus on various imperfections that limit their performance. In particular, we deal with nonidealities associated with the sampling network, the DAC, the comparator, and the logic. Some of the issues are the same as those studied for samplers in Chapters 3 and 4, for comparators in Chapters 5 and 6, and for capacitor DACs in Chapter 11. We also introduce methods of alleviating the imperfections.

15.1 Sampling Nonidealities

As seen in Chapter 14, the use of capacitor DACs as input samplers obviates the need for an explicit front-end sample-and-hold circuit. Nonetheless, both top-plate and bottom-plate sampling architectures entail other challenges.

15.1.1 Switch Resistance

Consider the top-plate sampling circuit shown in Fig. 15.1. Unlike the topologies studied in Chapter 3, this structure places *two* switches in series with C_1 , roughly doubling the acquisition time constant. Since the

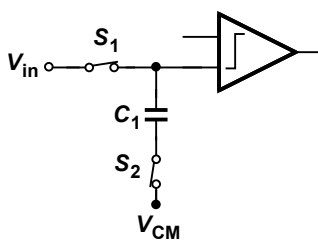


Figure 15.1 Top-plate sampling circuit.

on-resistance and channel charge of S_1 vary with V_{in} , we must apply bootstrapping to this switch. But how about S_2 ? This device sustains a relatively constant gate-source voltage and, in principle, need not employ bootstrapping. However, since V_{CM} is typically around half of the supply voltage, an NMOS or PMOS device serving alone as S_2 would suffer from a high on-resistance. We can realize S_2 by a complementary pair of wide transistors. Alternatively, we can bootstrap this switch so as to lower its on-resistance.

Example 15.1

A 10-bit SAR ADC incorporates a differential capacitor DAC comprising a 6-bit thermometric segment and a 4-bit binary array. Explain how bootstrapping is realized for the common-mode switches.

Solution

Even though V_{CM} is constant, we may use bootstrapping to provide a greater overdrive for the switches. Depicted in Fig. 15.2 is one-half of the DAC, revealing that all of the CM switches can share a single bootstrapping network. Unfortunately, capacitor C_B must be large because it should experience minimal charge sharing while driving about 70 switches and their gate interconnects, i.e., the long clock lines traveling across the arrays.

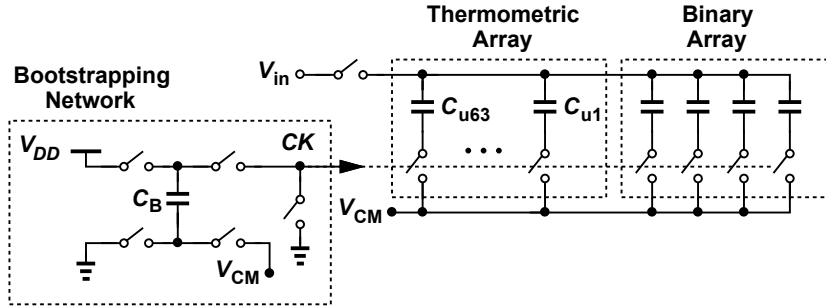


Figure 15.2 Bootstrapping network for a DAC.

Another approach is to employ “clock boosting.” Illustrated in Fig. 15.3(a), the idea is to shift the average (dc) level of the clock up to approximately V_{DD} so as to provide a large gate-source overdrive voltage for the switches. We note that if $R_B C_B$ is much greater than $T_{CK}/2$, node X exhibits a nearly square wave toggling between $V_{DD}/2$ and $3V_{DD}/2$ (if the clock has a 50% duty cycle). In other words, V_{GS} varies between zero and V_{DD} even though the sources reside at V_{CM} .

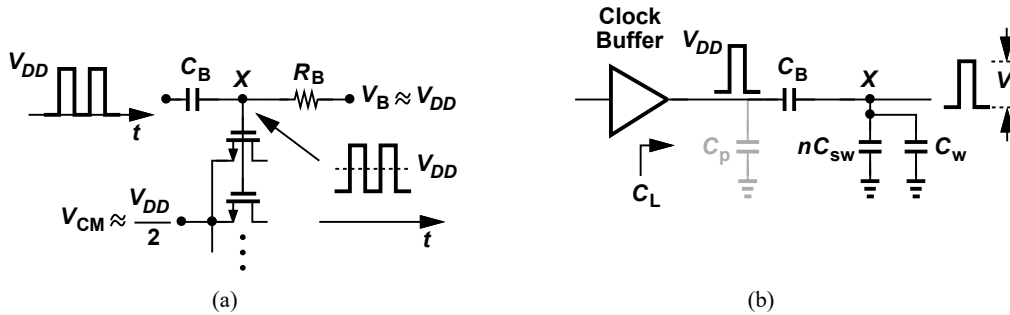


Figure 15.3 (a) Clock boosting, and (b) effect of parasitics.

Clock boosting, too, requires that C_B be large. This is quantified with the aid of Fig. 15.3(b), where nC_{sw} denotes the gate capacitance of n switches and C_w the entire wiring capacitance associated with these gates. For the swing at X , V_1 , to remain close to V_{DD} , we have $C_B \gg nC_{sw} + C_w$. But this also means that the parasitic of C_B itself, C_p , now loads the clock buffer heavily. For example, if $C_B \approx 10(nC_{sw} + C_w)$ and $C_p \approx 0.05C_B$, we have $C_p \approx 0.5(nC_{sw} + C_w)$; i.e., clock boosting raises the capacitance presented to the buffer by 50%. We observe a proportional increase in the buffer’s power consumption, $P = f_{CK} C_L V_{DD}^2$, where $f_{CK} = 1/T_{CK}$. Selection of wide transistors, bootstrapping, or clock boosting for common-mode switches depends on the targeted speed.

The bottom plate of C_1 in Fig. 15.1 must also switch to V_{REF}^+ and V_{REF}^- . If these voltages are close to the supply rails, a single PMOS device suffices for the former and a single NMOS device for the latter.

We now turn to bottom-plate sampling (Fig. 15.4). Recall from Chapter 14 that here S_T turns off before S_1 does so as to avoid the input-dependent charge injection of the latter. In principle, therefore, S_1 need not be bootstrapped but, in practice, it often is for speed reasons. This is because maintaining a constant on-

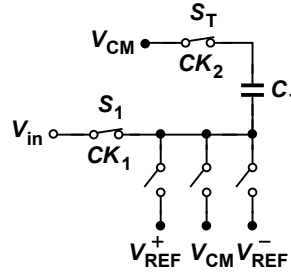


Figure 15.4 Bottom-plate sampling.

resistance generally reduces the acquisition time and the distortion. The implementation of the other switches follows the same considerations as those for top-plate sampling.

15.1.2 Switch Junction Capacitance

As noted in Chapter 3, if the sampling switch width is increased, the nonlinearity of its source or drain junction capacitance eventually manifests itself. This issue afflicts top-plate switches in both top-plate and bottom-plate sampling. Interestingly, SAR operation suppresses the nonlinearity in the latter but not in the former. Before delving into this point, let us make an observation.

Example 15.2

The DAC in a single-ended top-plate-sampling ADC is modeled as shown in Fig. 15.5, where, at the end of SAR conversion, k units have switched from V_{REF} to zero while $m - k$ have not. Determine the value of k (the ADC digital output) if V_A starts at V_{in} and ends at V_{CM} .

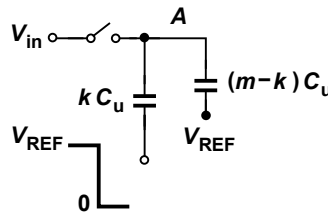


Figure 15.5 Simplified DAC circuit.

Solution

The capacitive divider yields an output voltage change equal to $-(k/m)V_{REF}$, which must yield the desired value of $V_{CM} - V_{in}$. It follows that

$$k = \frac{V_{in} - V_{CM}}{V_{REF}} m. \quad (15.1)$$

It is helpful to view $V_{in} - V_{CM}$ as the “input” because k is proportional to it. As explained below, this result incurs an error in the presence of capacitance nonlinearity. Note that the slope of the ADC input-output characteristic is given by the coefficient of V_{in} , which is equal to m/V_{REF} .

Consider the differential top-plate-sampling ADC shown in Fig. 15.6, where C_{ja} and C_{jb} denote the junction capacitances of S_{1a} and S_{1b} , respectively. Recall from Chapter 14 that the high SAR loop gain forces $V_{DAC}^+ - V_{DAC}^-$ toward zero as successive approximations progress. Thus, the voltages across C_{ja} and C_{jb} change from, respectively, V_{in}^+ and V_{in}^- to approximately V_{CM} . The voltage-dependent nature of these capacitances thus leads to DAC nonlinearity. To quantify this effect, we must first return to fundamentals.

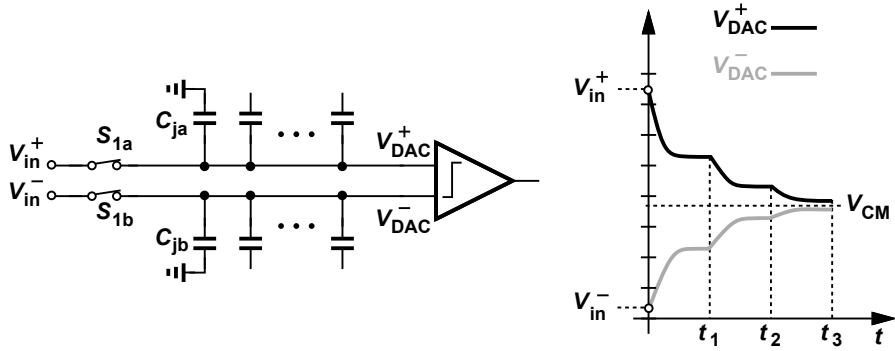


Figure 15.6 Top-plate sampling including switch junction capacitances.

Definition of Capacitance For a linear capacitor, C_1 , we have $Q = C_1V$ and $I = dQ/dt = C_1dV/dt$. For a nonlinear capacitor, C_2 , on the other hand, the situation is less straightforward. Depicted in Fig. 15.7(a) are the Q - V characteristics for the two cases. Suppose the charge on C_2 is given by

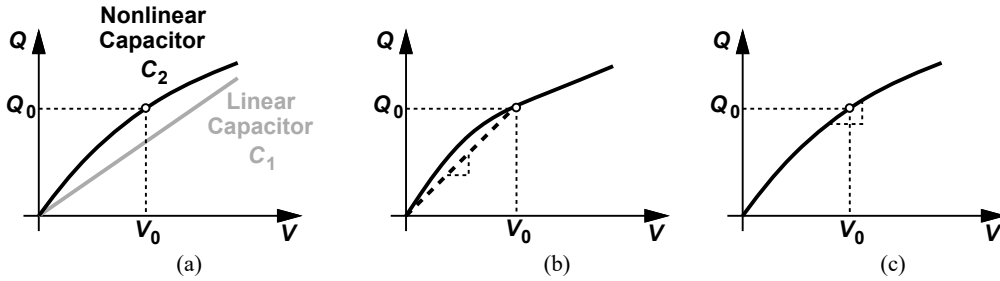


Figure 15.7 (a) Charge-voltage characteristics, and (b) one definition of capacitance, and (c) another definition.

$$Q = f(V) = \beta_1 V + \beta_2 V^2 + \beta_3 V^3. \quad (15.2)$$

We ask, how do we define the value of C_2 at $V = V_0$? We could simply write $Q = C_2V$ and hence $C_2 = Q_0/V_0$ [Fig. 15.7(b)]. That is,

$$C_2(V_0) = \beta_1 + \beta_2 V_0 + \beta_3 V_0^2. \quad (15.3)$$

But we also have

$$I = \frac{dQ}{dt} \quad (15.4)$$

$$= (\beta_1 + 2\beta_2 V + 3\beta_3 V^2) \frac{dV}{dt}. \quad (15.5)$$

If viewed in the form of $I = C_2 dV/dt$, this equation yields $C_2 = \beta_1 + 2\beta_2 V + 3\beta_3 V^2$, which is inconsistent with Eq. (15.3). One can argue that the relation $I = CdV/dt$ does not hold for nonlinear capacitors and Eq. (15.3) is still the correct definition of capacitance. Alternatively, one could define $C_2 = dQ/dV$ [Fig. 15.7(c)] and hence

$$C_2 = \beta_1 + 2\beta_2 V + 3\beta_3 V^2, \quad (15.6)$$

which agrees with Eq. (15.5). We proceed with the latter definition, noting that the total charge on a nonlinear capacitor at a voltage of V_0 is written as

$$Q(V_0) = \int_0^{V_0} C(V)dV, \quad (15.7)$$

and *not* as $Q(V_0) = C(V_0)V_0$.

Effect of Capacitance Nonlinearity Let us simplify the arrangement of Fig. 15.6 to that in Fig. 15.8(a), where C_{DAC} denotes the total DAC capacitance and is assumed linear. If the voltage at node A changes from

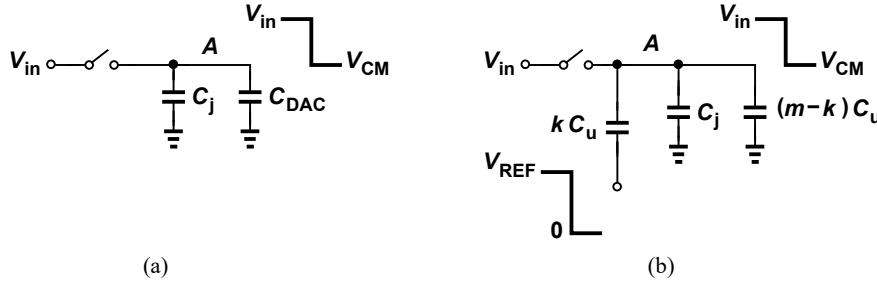


Figure 15.8 (a) Sampling circuit including junction capacitance, and (b) simplified DAC circuit.

V_{in} to V_{CM} , the charge delivered to $C_j + C_{DAC}$ is given by (15.7):

$$Q = \int_{V_{in}}^{V_{CM}} (C_{DAC} + C_j)dV. \quad (15.8)$$

For mathematical convenience, we write Eq. (15.6) differently, $C_j(V) = C_0(1 + \alpha V + \alpha' V^2)$ and drop the second-order term for now. It follows that

$$Q = (V_{CM} - V_{in})(C_{DAC} + C_0) + \alpha C_0 \int_{V_{in}}^{V_{CM}} V dV \quad (15.9)$$

$$= (V_{CM} - V_{in})(C_{DAC} + C_0) + \frac{\alpha}{2} (V_{CM}^2 - V_{in}^2) C_0. \quad (15.10)$$

The term $-(\alpha/2)V_{in}^2 C_0$ reveals the nonlinearity in Q .

In a SAR environment, however, we are interested not in Q but in the digital output. We then return to Example 15.2 and ask how k must be calculated if C_j loads the DAC. Drawing the circuit as shown in Fig. 15.8(b), we recognize that the voltage across $C_j + (m-k)C_u$ changes from V_{in} to V_{CM} , requiring a certain amount of charge that can be expressed from (15.10):

$$Q = (V_{CM} - V_{in})[(m-k)C_u + C_0] + \frac{\alpha}{2} (V_{CM}^2 - V_{in}^2) C_0. \quad (15.11)$$

This charge must be provided by kC_u in Fig. 15.8(b). Since kC_u is assumed linear, its charge change is proportional to its voltage change:

$$\Delta Q = kC_u[(0 - V_{CM}) - (V_{REF} - V_{in})], \quad (15.12)$$

where $0 - V_{CM}$ and $V_{REF} - V_{in}$ are respectively the final and initial voltages across kC_u . Equating (15.11) and (15.12) gives the necessary value of k (the digital output) for V_A to go from V_{in} to V_{CM} :

$$k = \frac{V_{in} - V_{CM}}{V_{REF}} m + \frac{V_{in} - V_{CM}}{V_{REF}} \frac{C_0}{C_u} + \frac{\alpha}{2} \frac{V_{in}^2 - V_{CM}^2}{V_{REF}} \frac{C_0}{C_u}. \quad (15.13)$$

The first term on the right-hand side agrees with (15.1) and represents the ideal case. The second term is the gain error as it alters the slope of the input-output characteristic from m/V_{REF} to $m/V_{REF} + (C_0/C_u)/V_{REF}$. The third term signifies nonlinearity. We can say the SAR loop “distorts” k so that the top plate can still go from V_{in} to V_{CM} . Note that these calculations assume a single-ended DAC; for a differential topology, this effect is suppressed unless we write $C(V) = C_0(1 + \alpha_1 V + \alpha_2 V^2)$ and focus on α_2 .

Example 15.3

Assuming $C_j \approx C_0$ (a constant value) in Fig. 15.8(b) and viewing C_0 as capacitive loading at the DAC output, explain whether (a) the DAC gain error is positive or negative, and (b) the ADC gain error is positive or negative.

Solution

The loading attenuates the voltage changes generated by the DAC. As the digital input to the DAC goes from zero to m , its analog output range is reduced due to C_0 [Fig. 15.9(a)].

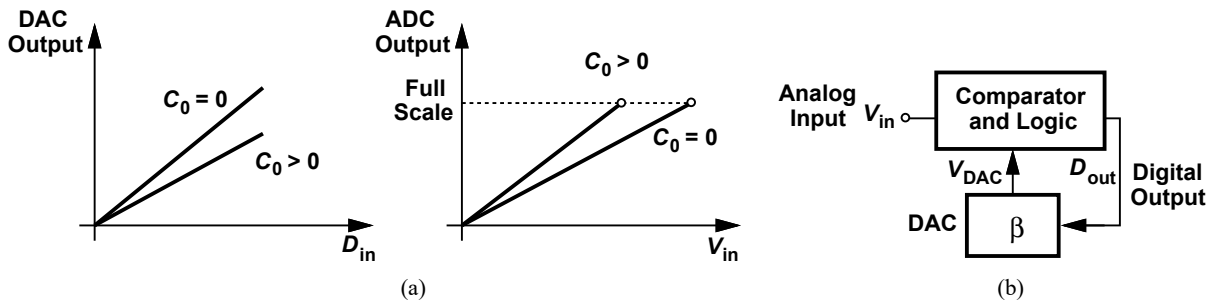


Figure 15.9 (a) Effect of capacitive loading on DAC and ADC characteristics, and (b) simple loop model.

The situation is different for the ADC. We must determine how the *digital* output behaves as the analog input varies. The DAC attempts to produce an analog voltage change equal to $V_{CM} - V_{in}$. The SAR loop must therefore swing the bottom plates of *more* unit capacitors from V_{REF} to zero if C_j is present. That is, the necessary value of k is *greater* than that in the ideal case, suggesting a positive gain error. Figure 15.9(a) conceptually illustrates the two cases. This point can also be seen in the generic SAR loop shown in Fig. 15.9(b) by noting that $V_{DAC} = \beta D_{out}$ and hence $D_{out} \approx V_{in}/\beta$. If the DAC incurs a negative gain error, $-\epsilon$, we have $V_{DAC} = (\beta - \epsilon)D_{out}$, observing that $D_{out} \approx V_{in}/(\beta - \epsilon)$.

The junction capacitance nonlinearity of the top-plate sampling switches manifests itself in high-speed designs. Since these switches must be wide enough to allow fast acquisition for a given DAC capacitance, their junction nonlinearity can limit the performance, a serious issue for resolutions above approximately 8 bits. In Problem 15.4, we explain why the junction capacitance of the bottom-plate switches in a top-plate-sampling loop does not affect the ADC’s linearity.

We now consider the bottom-plate-sampling SAR ADC architecture (Chapter 14). Shown in Fig. 15.10, the loop begins with $V_{DAC} = V_{CM}$ and ends with $V_{DAC} \approx V_{CM}$. The key observation here is that the total change in the charge on the junction capacitance of S_1 is nearly equal to zero.¹ Thus, C_j contributes negligible nonlinearity. From another point of view, since the DAC output node acts as a virtual ground (Chapter 14), it suppresses the nonlinearity associated with C_j . This is another advantage of this architecture.

¹We assume the nonlinearity is not so large as to “mislead” the loop.

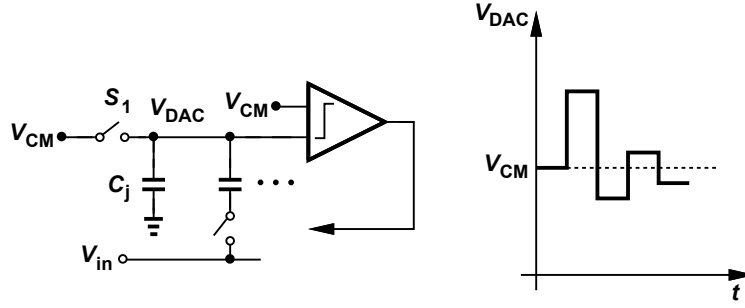


Figure 15.10 Effect of switch junction capacitance on bottom-plate sampling.

15.2 DAC Nonidealities

Our study of capacitor DACs in Chapters 8 and 11 indicates imperfections such as offset, gain error, nonlinearity, reference settling, kT/C noise, and speed limitations. In this section, we examine these phenomena in the context of a SAR ADC environment.

15.2.1 DAC kT/C Noise

Consider the kT/C noise deposited by the top-plate switches onto the DAC capacitors in Fig. 15.11(a). Since these voltages simply add to $V_{DAC}^+ - V_{DAC}^-$, we can place them in series with the comparator inputs. As such,

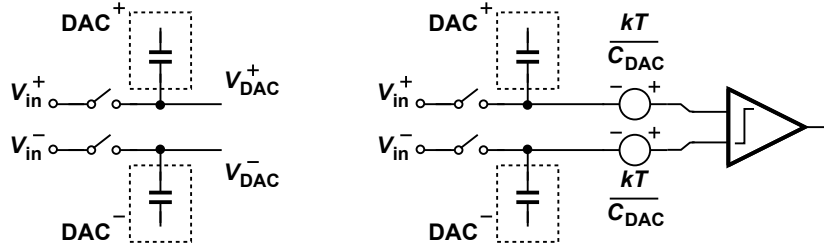


Figure 15.11 (a) kT/C noise deposited by switches onto DACs, and (b) circuit model.

these noise components simply add to the sampled value of $V_{in}^+ - V_{in}^-$. We therefore return to our kT/C noise analysis in Chapter 12 and express the ADC SNR as

$$\text{SNR} = \frac{\frac{A^2}{2}}{\frac{\Delta^2}{12} + \frac{2kT}{C_{DAC}}}, \quad (15.14)$$

where A denotes the peak amplitude of the differential input sinusoid and Δ the LSB. The choice of C_{DAC} is governed by how much SNR *penalty* we can tolerate. We define the penalty as the ratio of the “ideal” SNR and that in (15.14):

$$\text{SNR Penalty} = \frac{\frac{A^2}{2}}{\frac{\Delta^2}{12}} \div \frac{\frac{A^2}{2}}{\frac{\Delta^2}{12} + \frac{2kT}{C_{DAC}}} \quad (15.15)$$

$$= 1 + \frac{24kT}{\Delta^2 C_{DAC}}. \quad (15.16)$$

For example, if $\Delta = 1$ mV, then a DAC capacitance of 0.5 pF yields a penalty of $1.2 \equiv 0.8$ dB at $T = 300$ K. For an N -bit ADC, the unit capacitance is equal to $C_u = 0.5 \text{ pF}/2^N$ in this case. It is possible that the minimum value of C_u is dictated by matching rather than by kT/C noise constraints (Chapter 11).

15.2.2 DAC Offset and Gain Error

Capacitor DACs typically exhibit small offsets, the main component arising from the mismatch between the top-plate switches. As depicted in Fig. 15.12(a), S_1 and S_2 introduce an offset between V_{DAC}^+ and V_{DAC}^- as they turn off because their overlap capacitances and their channel charge incur some random mismatches. This point holds for both top-plate-sampling and bottom-plate-sampling architectures.

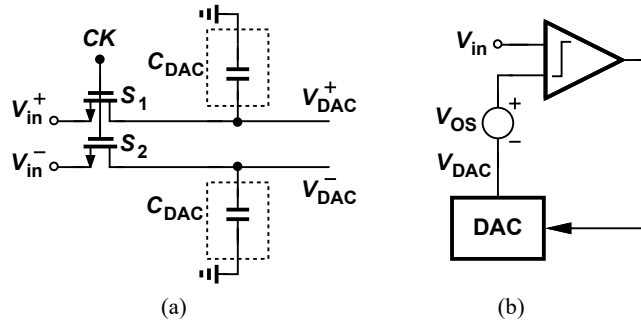


Figure 15.12 (a) Differential sampler, and (b) effect of offset.

The DAC offset, V_{OS} , in a SAR environment simply translates to an offset in the ADC characteristic. This is because V_{OS} can be modeled as a voltage source in series with the DAC output and hence in series with the comparator input [Fig. 15.12(b)]. Since the comparator senses $V_{in} - (V_{OS} + V_{DAC}) = (V_{in} - V_{OS}) - V_{DAC}$, we can view V_{OS} as a component in series with the ADC input.

We have seen in Chapter 11 that capacitor DACs experience gain error if their output node is loaded with parasitic capacitances. Figure 15.13 illustrates these components: (1) the top-plate switch(es) introduce C_{GD} and C_{DB} ; (2) each unit capacitor suffers from a top-plate parasitic, C_T ; (3) the routing within the DAC adds a wire capacitance, C_w ; and (4) the comparator presents its input capacitance, C_{comp} . The gain error can thus reach 5% to 10%.

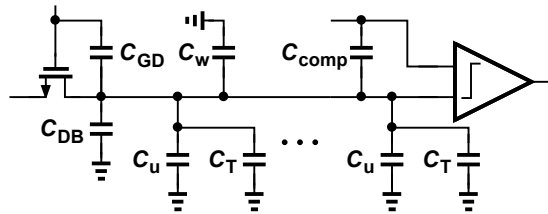


Figure 15.13 Gain error due to parasitic top-plate capacitances.

As explained in Example 15.3, a negative DAC gain error translates to a positive ADC gain error. This means that the digital output reaches its maximum for an input level *less* than the nominal full scale. In other words, the input range is narrower and the analog LSB smaller. This is the principal consequence of DAC gain error, indicating that a value of 5% to 10% may be tolerable.

Example 15.4

The DAC in a SAR ADC incurs a gain error of -20% . Explain the undesirable effect of this large error.

Solution

Since the analog LSB is 20% less than the nominal, we must revisit the DAC kT/C noise calculations in

Section 15.2.1. If Δ drops by 20% in Eq. (15.16), the SNR penalty may be significant. For example, with $C_{DAC} = 0.5$ pF, we have a penalty of 0.8 dB for $\Delta = 1$ mV and 1.2 dB for $\Delta = 0.8$ mV. Of course, the ADC input range is also reduced by 20%.

The foregoing example and Eq. (15.16) imply that the SNR degradation due to the gain error can be alleviated by increasing C_{DAC} . For example, if Δ falls from 1 mV to 0.8 mV, C_{DAC} can be raised from 0.5 pF to $0.5 \text{ pF} / (0.8)^2 = 1.56 \times 0.5 \text{ pF} = 0.78$ pF. But, to maintain the speed, this remedy requires that almost the entire SAR ADC be enlarged by a factor of 1.56: the sampling and DAC switches, the logic, the reference generator bypass capacitor(s), etc. The power consumption thus rises. In fact, if all of the parasitics in Fig. 15.13 scale linearly with C_u , then the gain error does not decrease!

15.2.3 DAC Nonlinearity

We wish to determine how the DAC DNL and INL manifest themselves in the overall ADC characteristic. Consider the DAC behavior shown in Fig. 15.14(a), where Δ_1 denotes the DAC output LSB, and a differential nonlinearity of DNL_j occurs at D_j . We sweep the ADC input, V_{in} , and see how the DAC responds, noting that

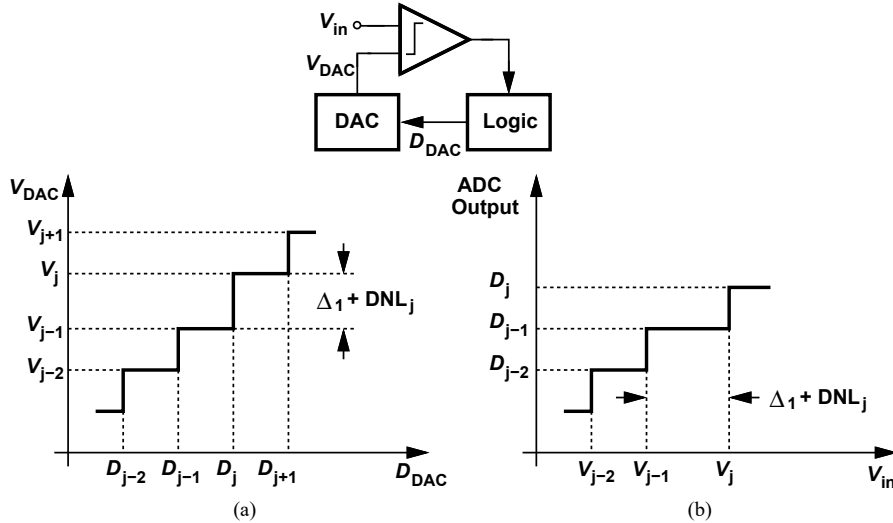


Figure 15.14 DNL in (a) DAC, and (b) overall ADC.

the high SAR loop gain forces V_{DAC} to approach V_{in} . For simplicity, we assume that the DAC output always settles to a value *lower* than V_{in} . If $V_{j-2} < V_{in} < V_{j-1}$, we have $V_{DAC} = V_{j-2}$ and hence $D_{DAC} = D_{j-2}$. This is the ADC output code. As V_{in} exceeds V_{j-1} , D_{DAC} jumps to D_{j-1} . Now, as V_{in} increases from V_{j-1} , no change occurs until V_{in} reaches V_j , at which point the DAC generates V_j and the ADC output is equal to D_j [Fig. 15.14(b)]. Since the ADC input must rise from V_{j-1} to V_j for the digital output to change by 1 LSB, we conclude that the ADC also incurs a DNL equal to DNL_j .

For the DAC integral nonlinearity, we first recall from Chapter 8 that

$$INL = V_{DAC} - V_1, \quad (15.17)$$

where V_1 denotes the straight line passing through the end points of the DAC's characteristic. That is, the INL can be modeled by a voltage source, V_{INL} , in series with an ideal DAC [Fig. 15.15(a)].² Placing this model in a SAR loop as shown in Fig. 15.15(b), we recognize that the comparator senses $V_{in} - (V_{INL} + V_1)$, i.e.,

²An ideal DAC generates $V_{out} = V_1$.

That is,

$$t_1 = \frac{m}{4} R_{out} C_u \ln \frac{m}{1.6}. \quad (15.19)$$

Note that t_1 must not be longer than the time allocated to the DAC's settling. As explained in Chapter 9, the reference output impedance rises at high frequencies, necessitating simulations to refine these results. We have also seen in Chapter 9 that a sufficiently large bypass capacitor tied between V_{REF}^+ and V_{REF}^- can alleviate the settling to some extent. We should remark that the DAC also creates reference bounce when it is reset, an issue if the bounce does not subside before the next SAR cycle.

15.2.5 Reference Noise

The circuit that generates the reference voltage(s) for a DAC inevitably introduces random noise as well. At resolutions of 10 bits or higher, this noise can degrade the performance.

Consider the situation depicted in Fig. 15.17(a), where k unit DAC capacitors are switched to V_{REF} and $\overline{V_{n,R}^2}$ models the reference noise spectrum. The noise contains both flicker and thermal components

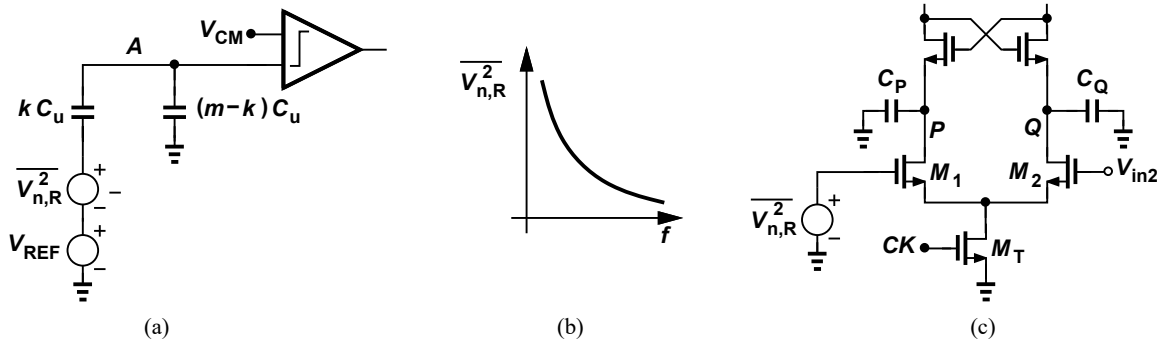


Figure 15.17 (a) Effect of reference noise, (b) noise spectrum, and (c) comparator sensing this noise.

[Fig. 15.17(b)]. We wish to determine how the comparator decision is corrupted by $V_{n,R}$. Let us make two observations. First, the worst case occurs if $V_{n,R}$ reaches node A unattenuated, i.e., if $k = m$ and the DAC input is equal to $11 \cdots 1$.³ Second, since $V_{n,R}$ travels through the input stage of the comparator along with the signal, we must subject its spectrum to the corresponding frequency response. To understand this point, suppose the StrongArm topology in Fig. 15.17(c) receives $\overline{V_{n,R}^2}$ as it evaluates its input difference. We expect that the low-pass response created by C_P and C_Q limits the bandwidth of $\overline{V_{n,R}^2}$. We also recognize that $\overline{V_{n,R}^2}$ simply appears in series with the gate of M_1 and is therefore indistinguishable from the gate-referred noise of this transistor. In other words, the comparator noise calculations in Chapter 6 apply here as well if we add the thermal component of $\overline{V_{n,R}^2}$ to $2 \times (4kT\gamma/g_{m1})$. The flicker noise of $\overline{V_{n,R}^2}$ and the comparator transistors typically contributes negligibly toward the total integrated noise.

Our method of merging the reference noise with the comparator noise in Fig. 15.17 is somewhat pessimistic as it assumes $V_{n,R}$ appears at A unattenuated in all SAR cycles. This is, of course, not true. In fact, in some of the SAR architectures of Chapter 14, the DAC is never configured with all capacitors switched to V_{REF} before the comparator is clocked. To obtain a more accurate estimate of the reference noise's effect on the ADC performance, transient noise simulations are necessary.

Example 15.5

Is it possible that reference noise in a differential SAR loop appears as a common-mode effect at the input of the comparator?

³We neglect the parasitic capacitances at A.

Solution

Looking back at the SAR ADCs in Chapter 14, we note that it is possible that some bottom plates of both DACs are switched to V_{REF}^+ (or V_{REF}^-) in a given cycle. Thus, a fraction of the reference noise becomes a common-mode component and does not affect the comparator decision.

15.3 Comparator Nonidealities

In addition to limiting the conversion speed, the comparator in a SAR ADC contributes offset and noise. Fortunately, the offset, V_{OS} , simply adds to the ADC input [Fig. 15.18(a)] and does not translate to DNL,

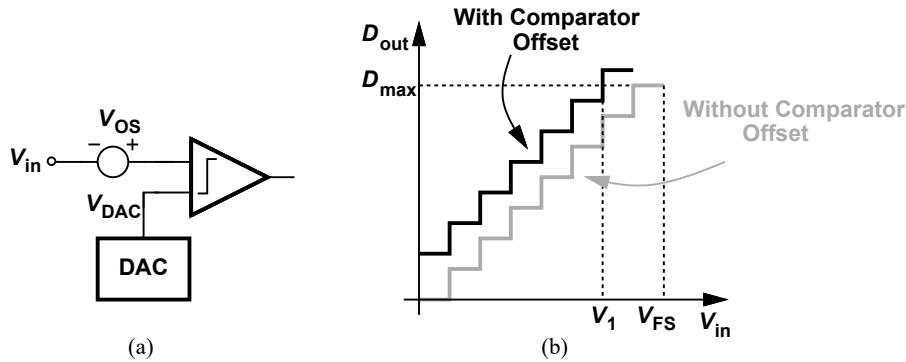


Figure 15.18 (a) Comparator offset, and (b) its effect of SAR ADC.

a point of contrast to the behavior of flash ADCs (Chapter 13). SAR converters can therefore tolerate a comparator offset greater than 1 LSB. The principal effect of V_{OS} is to limit the input range: as depicted in Fig. 15.18(b), V_{in} can reach only $V_1 = V_{FS} - V_{OS}$ for D_{out} to approach its maximum value. We say the ADC dynamic range is reduced by an amount equal to V_{OS} . For this reason, we prefer to design the comparator for an offset of no more than a few LSB.

The foregoing points are valid if V_{OS} is *constant*. On the other hand, in some SAR ADCs, the comparator input common-mode level, $V_{in,CM}$, changes during the conversion period (Chapter 16), potentially allowing V_{OS} to vary from one SAR cycle to the next. To understand this phenomenon, consider the StrongArm comparator input stage shown in Fig. 15.19(a), assuming $V_{in1} \approx V_{in2} \approx V_{in,CM}$. We recall from Chapter 5 that the offsets of the devices beyond M_1 and M_2 (e.g., M_3 and M_4) are divided by a certain gain as they are referred to the input. The key point here is that this gain has some dependence on the input CM level, a second-order effect neglected previously. We observe that the transconductance of M_1 and M_2 begins to fall as they enter the triode region, which occurs when V_P and V_Q drop to $V_{in,CM} - V_{TH1,2}$ [Fig. 15.19(b)]. In other words, at a higher CM level, M_1 and M_2 begin to lose transconductance *earlier*, yielding a lower overall gain and hence a greater offset.⁴ Thus, careful simulations are necessary to determine how much the offset varies for a given change in $V_{in,CM}$.

The comparator input-referred noise degrades the ADC SNR as it simply adds to the input signal (Fig. 15.20). From this viewpoint, we can write

$$\text{SNR} = \frac{\frac{A^2}{2}}{\frac{\Delta^2}{12} + V_{n,rms}^2}, \quad (15.20)$$

⁴This trend is, in fact, more complex: a higher input CM level also leads to a greater tail current and hence a *higher* transconductance.

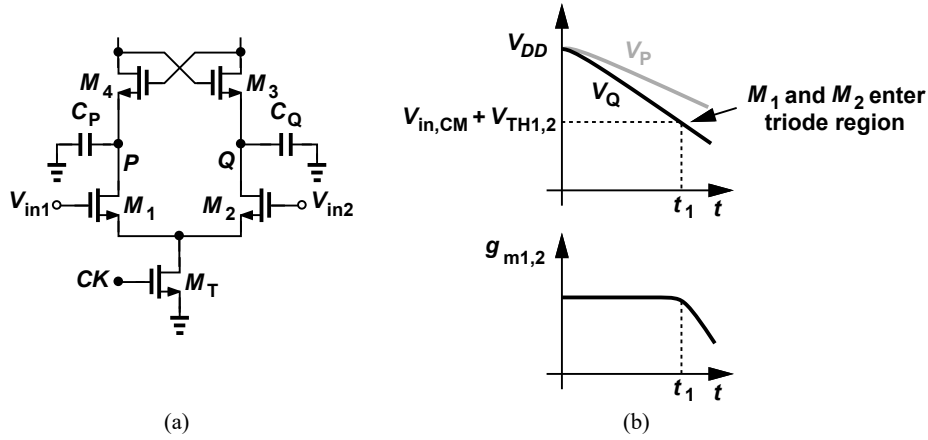


Figure 15.19 (a) Simplified StrongArm comparator, and (b) its waveforms.

where A is the input sinusoid peak amplitude and $V_{n,rms}$ the comparator rms noise. For example, if $V_{n,rms} = \Delta/5$, then the SNR degrades by 1.7 dB. If the DAC experiences a gain error, the SNR drops further.

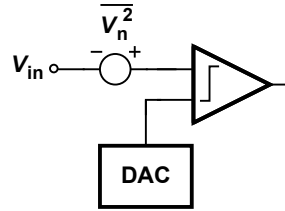


Figure 15.20 Effect of comparator noise.

15.4 Speed Limitations

The conversion speed of SAR ADCs is determined by three components: the comparator response, the logic delay, and the DAC settling time. Denoting these quantities by t_{comp} , t_{logic} , and t_{DAC} , respectively, we note that an N -bit generic SAR ADC takes $N(t_{comp} + t_{logic} + t_{DAC})$ seconds to complete one conversion period. This is the principal drawback of this architecture. By contrast, the flash ADC requires only t_{comp} seconds.⁵

15.4.1 Comparator Response

We have studied the speed limitations of comparators in Chapter 6, where we noted that t_{comp} rises as the input difference falls. Since this difference can be arbitrarily small, t_{comp} does not have a clear definition, but we commonly refer to its value when $|V_{in1} - V_{in2}| = 0.5$ LSB. In a SAR ADC environment, $|V_{in1} - V_{in2}|$ is greater than this amount in most of the cycles, but the conversion period must be long enough to accommodate this worst case.⁶

Let us examine a SAR loop in which the comparator response is longer than expected. Considering a StrongArm topology as an example, we observe that the outputs remain close to each other for some time [Fig. 15.21(a)]. We select a low threshold point for the inverters following the core such that this common mode level still yields low voltages for X' and Y' . During this time, the inverters generate equal levels at

⁵The delay of the logic following the comparators in flash ADCs does not directly affect the speed because the comparator and the logic can be pipelined.

⁶If $|V_{in1} - V_{in2}|$ is very small in a SAR cycle, the loop becomes metastable.

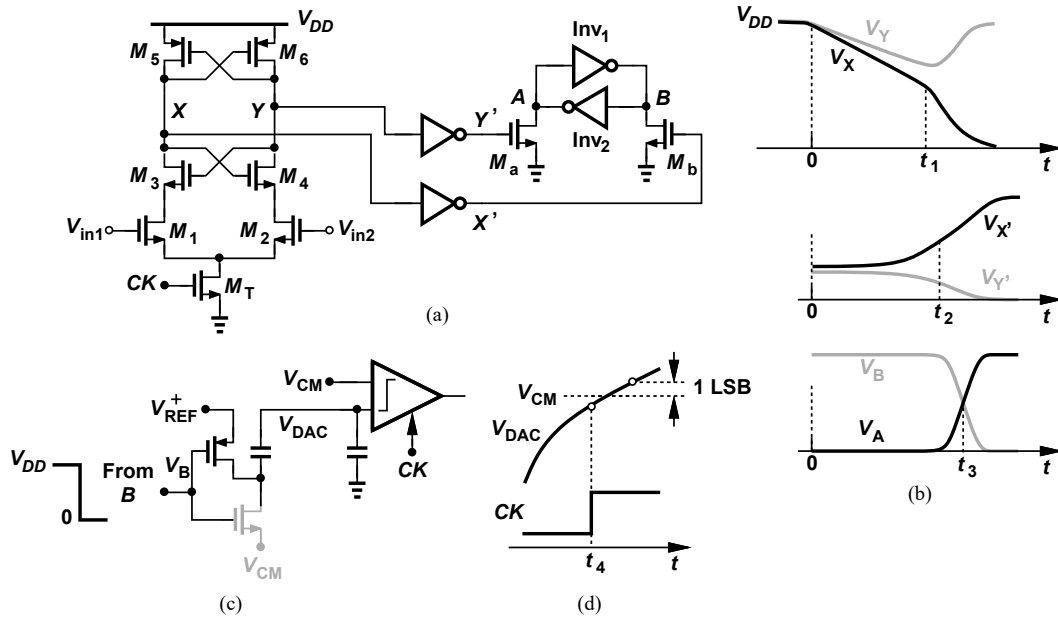


Figure 15.21 (a) Comparator for delay analysis, (b) internal waveforms, (c) simplified switch network, and (d) DAC and clock waveforms.

X' and Y' , typically no more than the threshold voltage of M_a and M_b [Fig. 15.21(b)]. That is, the RS latch remains undisturbed. As V_X and V_Y depart significantly and one falls below the threshold point of its corresponding inverter, X' or Y' rises and the RS latch begins to change its state. In summary, the comparator takes t_1 seconds to activate one of the inverters, the inverters trigger the latch around $t = t_2$, and the latch delivers valid logical levels at A and B after $t = t_3$. We denote their voltages by V_A and V_B , respectively.

The change in V_A or V_B arrives at the DAC input, as shown in Fig. 15.21(c), causing some capacitor bottom plates to switch. Now, V_{DAC} must settle to the required precision (e.g., within 0.5 LSB of its final value) before the comparator is clocked again. We recognize that an excessively long comparator response leaves insufficient time for the DAC, potentially introducing an error. Shown in Fig. 15.21(d) is an example where V_{DAC} intends to reach $V_{CM} + 1$ LSB but is sensed by the comparator at $t = t_4$.

Metastability The effect of comparator speed on the ADC performance is ultimately quantified by the metastability error rate. Since $|V_{in} - V_{DAC}|$ can be arbitrarily small with a finite probability, some SAR cycles make incorrect or no decisions. The reader is referred to [1, 2] for a detailed analysis.

15.4.2 DAC Response

It is instructive to study how a SAR ADC behaves when its DAC fails to settle properly. We consider a 4-bit single-ended top-plate-sampling architecture as an example. Recall from Chapter 14 that such a system employs a 3-bit DAC and requires that the DAC settle three times. For simplicity, we neglect the comparator and logic response times. Suppose $V_{in} = 8.7$ LSB in Fig. 15.22(a), where the comparator is clocked at $t = 0$, and V_{DAC} jumps down by 4 LSB [Fig. 15.22(b)]. At $t = t_1$, the comparator commands the loop to compensate for the undershoot by raising V_{DAC} by 2 LSB. This is followed by another upward jump of 1 LSB. The digital output is thus equal to 1000. Now, we assume the loop is excessively slow and V_{DAC} does not fall to 4.7 LSB before $t = t_1$ [Fig. 15.22(c)]. We model V_{DAC} by a first-order response, noting that

$$V_{DAC}(t) = V_{\infty} + (V_0 - V_{\infty}) \exp \frac{-t}{\tau}, \quad (15.21)$$

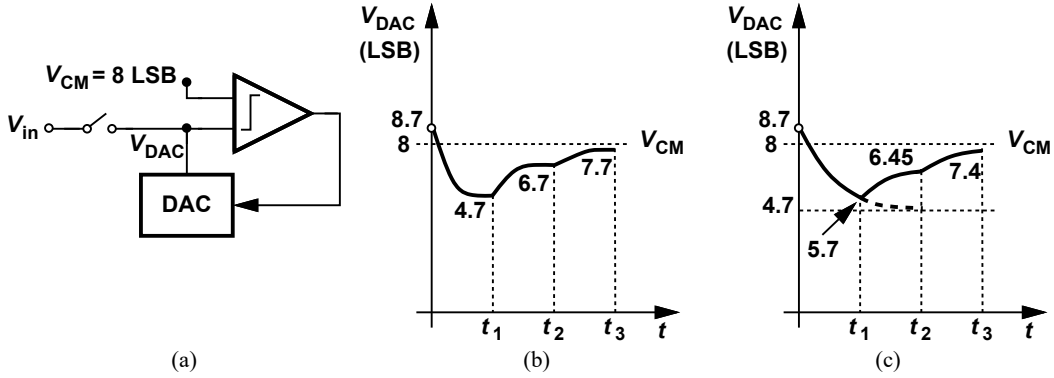


Figure 15.22 (a) SAR loop, (b) response for a slow DAC, and (c) response for a very slow DAC.

where V_{∞} and V_0 denote the final and initial values, respectively. At $t = t_1$,

$$V_{DAC}(t_1) = 4.7 \text{ LSB} + (8.7 \text{ LSB} - 4.7 \text{ LSB}) \exp \frac{-t_1}{\tau}. \quad (15.22)$$

For example, $V_{DAC}(t_1) = 5.7 \text{ LSB}$, which means $t_1 \approx 1.4\tau$ or $\tau = 0.71t_1$. The comparator makes a correct decision in this case, allowing V_{DAC} to rise. With $t_2 - t_1 = t_1 - 0 = 1.4\tau$, V_{DAC} changes as follows

$$V_{DAC}(t_2) = 6.7 \text{ LSB} + (5.7 \text{ LSB} - 6.7 \text{ LSB}) \exp \frac{-(t_2 - t_1)}{\tau} \quad (15.23)$$

$$= 6.45 \text{ LSB}. \quad (15.24)$$

Similarly, we have $V_{DAC}(t_3) = 7.4 \text{ LSB}$. All of the comparator decisions are correct for this particular input level.

But suppose $V_{in} = 4.6 \text{ LSB}$. Plotted in Fig. 15.23(a) is the correct response: V_{DAC} jumps up by 4 LSB, down by 2 LSB, and up by 1 LSB. The digital output is equal to 0100. With a time constant of $\tau = t_1/1.4$,

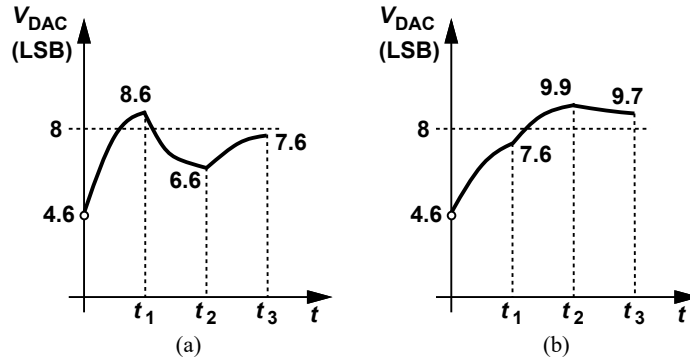


Figure 15.23 Loop operation with (a) sufficient speed, and (b) excessively slow response.

on the other hand, the DAC register jumps up by 4 LSB, but the DAC output does not:

$$V_{DAC}(t_1) = 8.6 \text{ LSB} + (4.6 \text{ LSB} - 8.6 \text{ LSB}) \exp \frac{-t_1}{\tau} \quad (15.25)$$

$$= 7.6 \text{ LSB}. \quad (15.26)$$

As shown in Fig. 15.23(b), the comparator decides that $V_{DAC} < V_{CM}$ and commands the register to raise V_{DAC} by 2 LSB. This means that the final value of V_{DAC} would be higher than $V_{in} = 4.6 \text{ LSB}$ by

4 LSB + 2 LSB = 6 LSB. Since V_{DAC} climbs from an actual value of 7.6 LSB toward 10.6 LSB,

$$V_{DAC}(t_2) = 10.6 \text{ LSB} + (7.6 \text{ LSB} - 10.6 \text{ LSB}) \exp \frac{-(t_2 - t_1)}{\tau} \quad (15.27)$$

$$= 9.9 \text{ LSB}. \quad (15.28)$$

The next comparator decision requires that V_{DAC} fall by 1 LSB, i.e., fall toward 9.6 LSB, yielding

$$V_{DAC}(t_3) = 9.6 \text{ LSB} + (9.9 - 9.6) \exp \frac{-(t_3 - t_2)}{\tau} \quad (15.29)$$

$$= 9.7 \text{ LSB}. \quad (15.30)$$

The digital output is thus equal to 0011 (why?). The 4-bit ADC incurs an error of 1 LSB in this case. The trajectory of Fig. 15.23(b) reveals that the loop does attempt to correct for the first decision but does not quite succeed.

Example 15.6

Determine the maximum tolerable τ in the above study if the loop must not incur an error.

Solution

The DAC output in Fig. 15.23(b) must cross 8 LSB at $t = t_1$:

$$8 \text{ LSB} = 8.6 \text{ LSB} + (4.6 \text{ LSB} - 8.6 \text{ LSB}) \exp \frac{-t_1}{\tau}. \quad (15.31)$$

That is,

$$\tau \leq \frac{t_1}{\ln(4/0.6)} \quad (15.32)$$

$$\leq 0.53t_1. \quad (15.33)$$

Note that we have neglected the comparator and logic response times.

In the foregoing failure study, the DAC's first incomplete settling caused the second comparator decision to be incorrect. Is there an input value for which the DAC's *second* incomplete settling leads to an incorrect decision? Figure 15.24 depicts the situation. For a fast DAC, V_{DAC} begins from $V_{in} = V_0$, rises to $V_0 +$

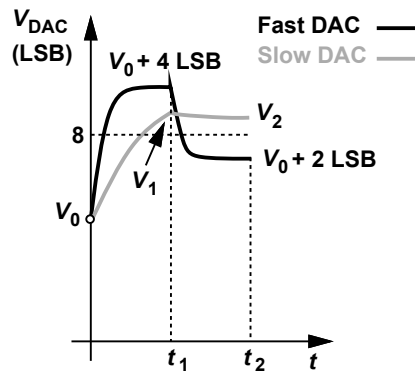


Figure 15.24 Problem of incomplete settling in the second cycle.

4 LSB > 8 LSB, and falls to $V_0 + 2 \text{ LSB} < 8 \text{ LSB}$. If the DAC is slow, V_{DAC} reaches $V_1 > 8 \text{ LSB}$ at t_1 but falls to $V_2 > 8 \text{ LSB}$ at t_2 . The third comparator decision is then incorrect. As before, let us assume $t_1 = 1.4\tau$. To obtain the value of V_0 that leads to this error, we write from Eq. (15.21):

$$V_1 = V_0 + 4 \text{ LSB} + (V_0 - V_0 - 4 \text{ LSB}) \exp \frac{-t_1}{\tau} \quad (15.34)$$

$$= V_0 + 3 \text{ LSB}. \quad (15.35)$$

For V_2 , we have

$$V_2 = V_0 + 2 \text{ LSB} + (V_0 + 3 \text{ LSB} - V_0 - 2 \text{ LSB}) \exp \frac{-(t_2 - t_1)}{\tau} \quad (15.36)$$

$$= V_0 + 2.25 \text{ LSB}, \quad (15.37)$$

For both V_1 and V_2 to be greater than 8 LSB, we have $V_0 > 5.75 \text{ LSB}$. Of course, we also have $V_0 < 8 \text{ LSB}$ (why?).

15.4.3 Tapering and Minimum Delay

We have observed that the total delay around a SAR loop is given by $t_{comp} + t_{logic} + t_{DAC}$, where t_{DAC} denotes the DAC settling time. We also know that the DAC unit capacitors require a certain minimum width for the switches if t_{DAC} is to be minimized. Shown in Fig. 15.25 is a conceptual diagram raising the following

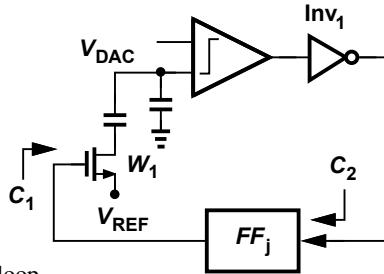


Figure 15.25 Tapering around a SAR loop.

question: how can we reduce the total delay? We can select relatively wide transistors in FF_j so as to drive C_1 , but this implies that Inv_1 drives a large capacitance, C_2 . Thus, Inv_1 must employ wide devices and so must the comparator core. These thoughts suggest that the transistors in the inverters and the flipflop can be tapered so as to minimize the total delay. For example, the device widths in FF_j can be about $W_1/2$ and those in Inv_1 and Inv_2 around $W_1/4$.

Example 15.7

Can we widen the comparator's transistors to reduce its delay for a given fanout?

Solution

Yes, but we face two issues. First, the comparator input capacitance rises and so does the DAC gain error. Second, the power consumption also climbs.

We have learned that the total DAC capacitance, C_{DAC} , is dictated by kT/C noise and/or matching requirements. With the value of C_{DAC} known, we can estimate a lower bound for the delay around the loop and hence the minimum possible SAR cycle time.

Recall from basic logic design that a cascade of stages can be tapered so as to minimize the total delay while the chain drives a certain load capacitance, C_L [Fig. 15.26(a)]. It is shown that, ideally, each stage is

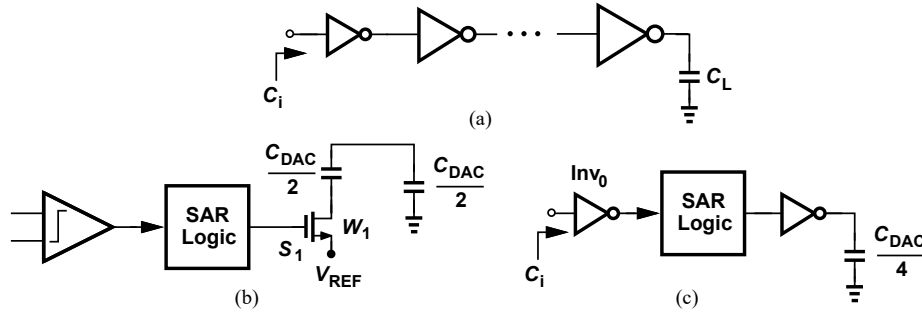


Figure 15.26 (a) Tapered inverters, (b) chain of blocks within the loop, and (c) simplified circuit for tapering computation.

scaled up by a factor of $e \approx 2.72$ with respect to the preceding one, and the optimum number of stages is equal to $\ln(C_L/C_i)$, where C_i is the input capacitance of the first inverter.⁷ If the gate delay with a fanout of 1 is denoted by T_d , the chain in Fig. 15.26(a) exhibits a delay of eT_d per stage and hence a total delay of $eT_d \ln(C_L/C_i)$.

In a SAR environment, the comparator and the logic form a critical path that drives the DAC. As shown in Fig. 15.26(b), the worst case occurs when half of the DAC capacitance must switch. Lumping the corresponding DAC switches into S_1 , we can view this device as a gate driving a load capacitance of $C_{DAC}/4$. We now model the comparator by an inverter of equal strength and hence a certain input capacitance, C_i [Fig. 15.26(c)]. The minimum delay is thus given by $T_{min} = eT_d \ln[C_{DAC}/(4C_i)]$, where T_d is the delay of Inv_0 .⁸ In an N -bit ADC, T_{min} is multiplied by at least $N - 1$ for one conversion. We can express this bound as $(N - 1)eT_d \ln[C_{DAC}/(4C_i)]$, which is relatively independent of power consumption (and excludes the DAC settling time).

Example 15.8

A 10-bit SAR ADC incorporates a total DAC capacitance of 500 fF. If the equivalent inverter in Fig. 15.26(c) exhibits $C_i = 10$ fF and its unity-fanout delay is 10 ps, compute the minimum conversion period.

Solution

We have $(N - 1)eT_d \ln[C_{DAC}/(4C_i)] = 618$ ps. Even if we neglect the comparator response time and the DAC settling, the conversion period cannot fall below this value.

In summary, we make several remarks. (1) The comparator can exhibit a long response for *any* SAR cycle, depending on the analog input value. In Fig. 15.24, for example, V_0 can be chosen such that $V_0 + 4$ LSB is very close to 8 LSB in the first cycle. (2) Near the end of conversion, the DAC output steps become smaller, and so does the input difference sensed by the comparator. Thus, the DAC requires less time to settle whereas the comparator takes longer to respond. The dependence of the DAC settling time, t_{DAC} , upon the DAC voltage change can be formulated as follows. Suppose V_{DAC} must reach within 0.5 LSB of a final value, V_∞ . From Eq. (15.21), we have

$$V_\infty - 0.5 \text{ LSB} = V_\infty + (V_0 - V_\infty) \exp \frac{-t_{DAC}}{\tau}. \quad (15.38)$$

⁷This approximation neglects the drain junction capacitances in the inverters.

⁸ Inv_0 in this diagram represents the cascade of the comparator and Inv_1 in Fig. 15.25.

That is,

$$t_{DAC} = \tau \ln \frac{V_{\infty} - V_0}{0.5 \text{ LSB}}. \quad (15.39)$$

Thus, the DAC settling is longest in the first SAR cycle. Determined by the resistance of the DAC switches, this point is, however, not valid if the DAC reference suffers from a finite output impedance (as exemplified by Fig. 15.16).

References

- [1]. D. Bankman et al, “Understanding metastability in SAR ADCs: Part I: Synchronous,” *IEEE Solid-State Circuits Magazine*, vol. 11, pp. 86-97, Spring 2019.
- [2]. D. Bankman et al, “Understanding metastability in SAR ADCs: Part II: Asynchronous,” *IEEE Solid-State Circuits Magazine*, vol. 11, pp. 16-32, Summer 2019.

Problems

- 15.1.** The clock boosting circuit of Fig. 15.3(a) may “stress” the transistors, i.e., it may cause their V_{GS} or V_{GD} to exceed the maximum allowable value. Determine the range of V_{CM} that ensures $V_{GS}, V_{GD} \leq V_{DD}$.
- 15.2.** Repeat the analysis of Fig. 15.8(b) if $C_j = C_0(1 + \alpha_1 V + \alpha_2 V^2)$ and obtain the nonlinear term in Q .
- 15.3.** Consider the circuit of Fig. 15.5. (a) Construct a differential version of the circuit and explain how many capacitors in the other branch must switch from V_{REF} to zero. (b) Now, assume each branch of the differential circuit is loaded by a junction capacitance [as in Fig. 15.8(a)], and find the charge delivered to $C_j + C_{DAC}$ on each side if $C_j = C_0(1 + \alpha_1 V + \alpha_2 V^2)$. Denote the results by Q^+ and Q^- and find their nonlinear terms.
- 15.4.** Consider the top-plate sampling circuit of Fig. 15.1. Explain why the junction capacitance nonlinearity of S_2 does not affect the performance. (Hint: after S_2 turns off, the bottom plate of C_1 switches to another voltage.)
- 15.5.** A 10-bit differential SAR ADC must not incur an SNR penalty greater than 1 dB due to kT/C noise. Determine the minimum allowable DAC capacitance.
- 15.6.** The DAC kT/C noise calculation in Fig. 15.11 assumes top-plate sampling. Does the result hold for bottom-plate sampling as well?
- 15.7.** If we wish to increase the resolution of a SAR ADC by 1 bit, do we need to reduce its DAC gain error by a factor of 2?
- 15.8.** The DAC in a SAR ADC exhibits a DNL of 0.4 LSB with an ideal comparator. If the actual comparator presents a capacitance to the DAC, does the ADC still display DNL = 0.4 LSB?
- 15.9.** Repeat the reference bounce analysis of Fig. 15.16(c) if each DAC unit capacitor has a series switch resistance equal to R_{sw} .
- 15.10.** We wish to study the effect of reference noise [Fig. 15.17(a)] on a single-ended 3-bit SAR ADC with top-plate sampling. Explain what function of $V_{n,R}^2$ appears at the comparator output in different SAR cycles.
- 15.11.** Repeat the previous problem for a differential ADC. In which cycles does the reference noise become a partially common-mode effect?
- 15.12.** Determine the SNR of a single-ended SAR ADC employing a total DAC capacitance of C_{DAC} and a comparator with an integrated input-referred noise of $V_{n,rms}^2$.
- 15.13.** Repeat the analysis of Fig. 15.22(b) if $V_{in} = 7.8$ LSB. Compute the digital output and compare it to the ideal value.
- 15.14.** If $V_{in} = 2.1$ LSB in Fig. 15.22(b), in which cycle(s) does the comparator make an incorrect decision?
- 15.15.** Consider the waveform in Fig. 15.22(c) and assume that the DAC time constant, τ , is equal to the SAR cycle time, $t_2 - t_1$. Explain how the loop incurs an error.
- 15.16.** Suppose the SAR loop of Fig. 15.22(a) is designed for a resolution of 5 bits. If $V_{in} = 8.7$ LSB or 4.6 LSB and $\tau = 0.5(t_2 - t_1)$, sketch the DAC waveforms and explain whether any errors occur.
- 15.17.** Repeat the analysis of Fig. 15.22(b) if the 3-bit ADC employs bottom-plate sampling.
- 15.18.** The comparator in Fig. 15.22(a) is clocked at $t = 0, t_1$, etc. In which cycle in Fig. 15.22(b) does the comparator exhibit the longest response? How about in Fig. 15.23(a)?
- 15.19.** Consider the DAC settling times in Fig. 15.22(b) between $t = 0$ and t_1 and between t_1 and t_2 . If we take into account the comparator response time, which cycle poses a more stringent requirement on the SAR cycle time?
- 15.20.** In a N -bit SAR ADC with top-plate sampling and a reference voltage of V_{REF} , the DAC voltage begins with a value equal to V_{in} and must rise or fall by $V_{REF}/4$. Determine the worst-case settling time in terms of V_{in} and express the result as a function of the DAC time constant, τ .

Advanced SAR ADCs

In the past two decades, a great deal of research has been directed to improving the performance of SAR ADCs. The efforts have addressed issues such as the limited conversion speed, DAC complexity and precision, and power consumption. In this chapter, we study examples of these high-performance techniques.

We begin with the matter of speed and explore three methods of raising the conversion rate, namely, “asynchronous” operation, use of multi-bit quantizers, and use of a front-end flash stage. Next, we deal with DAC complexity and introduce approaches that reduce the number of its unit capacitors.

16.1 Speed Improvement

The conversion rate of SAR ADCs can be enhanced by four techniques: asynchronous operation, the use of multi-bit quantizers, the use of a flash sub-ADC, and by interleaving. We study the first three here and the last in Chapter 20.

16.1.1 Asynchronous Operation

Our analysis of SAR loops in Chapter 14 reveals that, in a given design, some SAR cycles *can* be shorter than others. This is because the comparator response time depends on its input difference, and the DAC settling time is longer for greater voltage steps. In other words, an optimum loop must allot *different* times to different cycles. Given the random nature of the input signal, we note that the ADC must automatically decide which cycles need more time.

It is possible to adjust the SAR cycle times by asynchronous operation. Depicted in Fig. 16.1, the idea is to determine when the comparator decision is completed and use the result, called “Ready,” to clock the comparator again [1]. The ADC thus creates a longer cycle time when the comparator responds slowly and vice versa. We can identify a “digital” loop formed by the comparator and the completion detector, and an

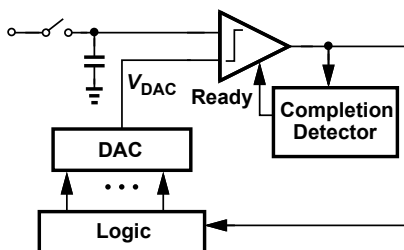


Figure 16.1 Basic asynchronous SAR ADC.

“analog” loop consisting of the comparator, the logic, and the DAC. Since the DAC must settle before the comparator enters the evaluation mode, we say the former loop must *not* be faster than the latter.

The reader may raise several questions about the conceptual illustration in Fig. 16.1. First, how exactly do we generate Ready? Second, for a StrongArm structure, how does Ready both reset and activate the comparator? Third, how do we ensure that the comparator begins to sense $V_{in} - V_{DAC}$ after V_{DAC} has settled to the necessary precision? Fourth, by how much does asynchronous operation improve the speed? We address these questions below.

Let us first implement the completion detector in Fig. 16.1 for a StrongArm comparator. Referring to Fig. 16.2(a), we expect Ready to rise when $V_X - V_Y$ reaches an acceptably large value. We recognize that

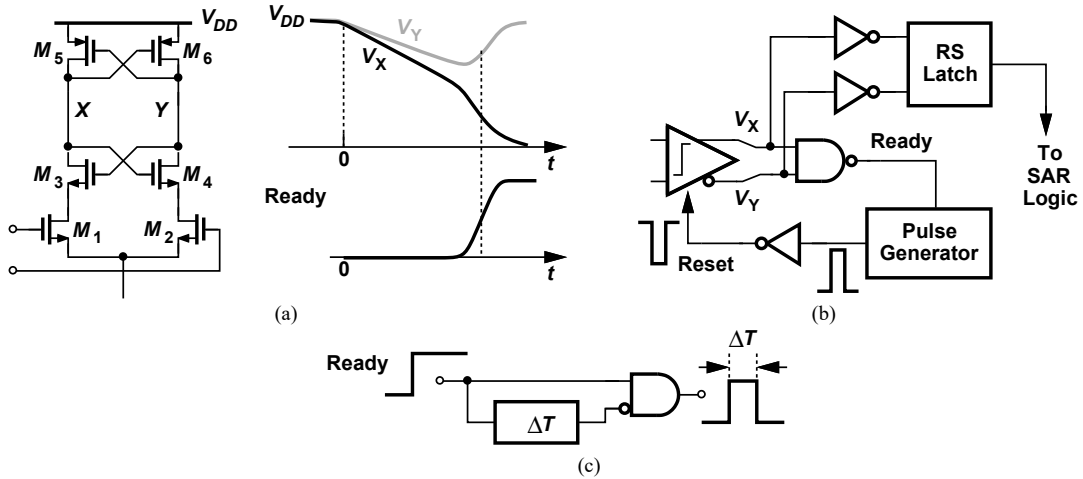


Figure 16.2 (a) Ready signal for a StrongArm comparator, (b) use of a NAND gate for Ready generation, and (c) conversion of Ready to a pulse.

Ready can be generated by applying V_X and V_Y to a NAND gate. But what should happen when Ready goes high? We expect that the comparator must be reset and then return to the evaluation mode. In other words, Ready must remain high for some time and subsequently fall back to zero. We therefore follow the completion detector by a pulse generator so as to define the reset time for the comparator [Fig. 16.2(b)]. Since the typical StrongArm implementation requires a low level for reset (precharge), the pulse is inverted. The pulse generator can be formed as shown in Fig. 16.2(c), with the delay stage, ΔT , defining the output pulse width. We have thus far answered the first two questions.

How long should ΔT in Fig. 16.1(c) be? Let us follow the waveforms around the SAR loop. As illustrated in Fig. 16.3, V_X and V_Y exhibit a reasonable difference, V_M , at t_1 , causing Ready to go high. Also, the RS latch in Fig. 16.2(b) captures the comparator decision and informs the DAC. After some delay, Reset goes low. The comparator is thus reset at t_2 and begins to evaluate at t_3 . In the meantime, V_{DAC} begins to change and approach its final value, V_∞ . The comparator must then remain in the reset mode until V_{DAC} reaches within 0.5 LSB of V_∞ , at $t = t_3$. Denoting the DAC settling time by t_S , we have $t_S \approx t_3 - t_1$ and hence $\Delta T = t_S - (t_2 - t_1)$, where $t_2 - t_1$ signifies the delay from the comparator output to the reset generator output. With the NAND gate in Fig. 16.2(b) and the AND gate in Fig. 16.2(c), $t_2 - t_1$ amounts to about two gate delays.¹ In summary, the reset pulsewidth should be roughly equal to the DAC settling time minus two gate delays. Figure 16.4 summarizes our thoughts thus far. The sampling lasts for T_1 seconds and conversion for $T_S - T_1$ seconds.

Example 16.1

How accurately can ΔT in Fig. 16.2(c) match the DAC settling time, t_S ?

¹The inverter in Fig. 16.2(b) can be merged with the AND in Fig. 16.2(c).

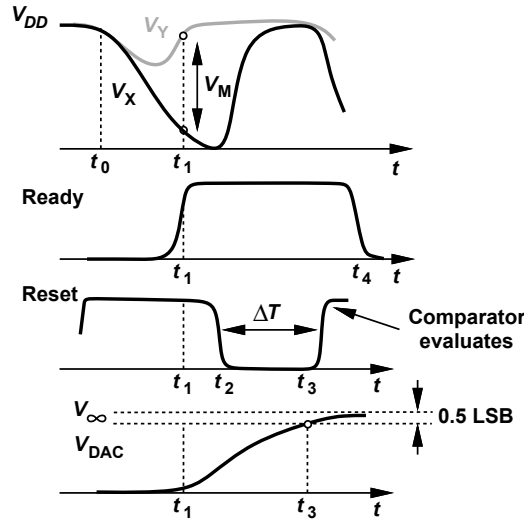


Figure 16.3 Asynchronous loop waveforms.

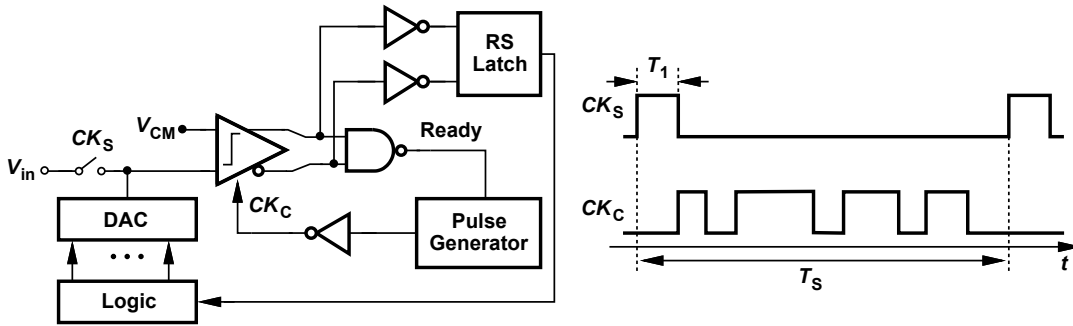


Figure 16.4 Clock waveforms for an asynchronous loop.

Solution

Not very accurately! The principal issue is that ΔT is generated by large-signal switching and Δt_S by small-signal settling. The ΔT block in Fig. 16.2(c) typically employs a chain of inverters, thereby suffering from nearly a twofold change in the delay with PVT variations. The DAC settling, on the other hand, depends on the small-signal on-resistance of switches (which does track PVT changes), the DAC capacitors, and the reference output impedance. For this reason, ΔT must be chosen *large* enough so as to exceed $t_S - (t_2 - t_1)$ under all extreme conditions.

Speed Improvement The fourth point we must consider is the speed improvement afforded by asynchronous operation. It is important to recognize that the “self-timed” nature of this method helps only with the comparator delay, t_{comp} , and not the DAC and logic response times. We examine the dependence of t_{comp} upon the comparator input difference, $V_{in1} - V_{in2}$. Note that $t_{comp} \approx t_1 - t_0$ in Fig. 16.3. We intuitively expect that t_{comp} falls as $|V_{in1} - V_{in2}|$ rises, but it reaches a technology-imposed lower bound beyond a certain level (Fig. 16.5). In other words, those SAR cycles in which $|V_{in} - V_{DAC}| < \Delta V_0$ take more time, and those in which $|V_{in} - V_{DAC}| > \Delta V_0$ do not. We also recall from the metastability analysis in Chapter 6 that the outputs of the StrongArm circuit first fall relatively linearly with time and then begin to depart regeneratively (Fig. 16.6).

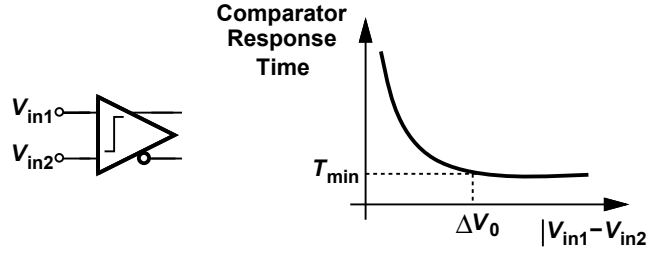


Figure 16.5 Comparator response time vs. input difference.

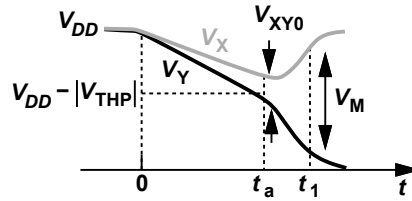


Figure 16.6 Response of a StrongArm comparator to a small input difference.

After t_a , regeneration gives

$$V_X - V_Y = V_{XY0} \exp \frac{t - t_a}{\tau_{reg}}, \quad (16.1)$$

where $V_{XY0} = V_X(t_a) - V_Y(t_a)$. Suppose $|V_X - V_Y|$ must reach a reasonable value, V_M . Thus, for $|V_{in1} - V_{in2}| < \Delta V_0$, the response time in Fig. 16.6 consists of t_a and another component equal to

$$t_1 - t_a = \tau_{reg} \ln \frac{V_M}{V_{XY0}}. \quad (16.2)$$

We have seen in Chapter 5 that $V_{XY0} = A_0(V_{in1} - V_{in2})$, where A_0 denotes the initial voltage gain. It follows that

$$t_1 - t_a = \tau_{reg} \ln \left| \frac{V_M}{A_0(V_{in1} - V_{in2})} \right|. \quad (16.3)$$

We conclude that the comparator response time consists of two components, namely, t_a in Fig. 16.6 and $t_1 - t_a$ in Eq. (16.3), the latter manifesting itself if $|V_{in} - V_{DAC}| < \Delta V_0$. Note that t_a also depends on $|V_{in} - V_{DAC}|$.

To determine the SAR speed improvement due to asynchronous operation, the analysis in [1] makes two assumptions: (1) the SAR loop response is dominated by the comparator delay, and (2) the comparator response is given by Eq. (16.3) for all SAR cycles. The best-case increase in the speed is by a factor of 2. In practice, however, the DAC and logic response times are not negligible and the comparator does not obey Eq. (16.3) in all of the cycles. We then expect a lesser improvement.

Metastability The most remarkable attribute of asynchronous operation is that it reduces the probability of metastable errors considerably. This is because the generation of Ready in Fig. 16.4 allows the comparator to automatically give itself sufficient time to make a decision. This point stands in contrast to the operation of synchronous SARs, in which each cycle time is fixed and metastability occurs more frequently. We intuitively expect a substantial drop in the error probability owing to its exponential dependence, $\exp(-T/\tau_{reg})$, upon the available time T in each SAR cycle. The reader is referred to [2] for a detailed analysis.

Clock Generation As explained in Chapter 15, an N -bit synchronous SAR ADC running at a rate of f_S requires a clock at Nf_S , which is difficult to produce in high-speed designs. Asynchronous topologies, on the other hand, confine the comparator clock generation to a “self-timed” local loop. We then surmise that the input clock need have only a frequency of f_S , and is thus simpler to generate.

In practice, however, we may still resort to a clock of frequency Nf_S . This is because the sampling operation requires only a small fraction of $T_S = 1/f_S$ for acquisition [Fig. 16.7(a)], dictating the frequency division approach described in Chapter 14 [Fig. 16.7(b)].

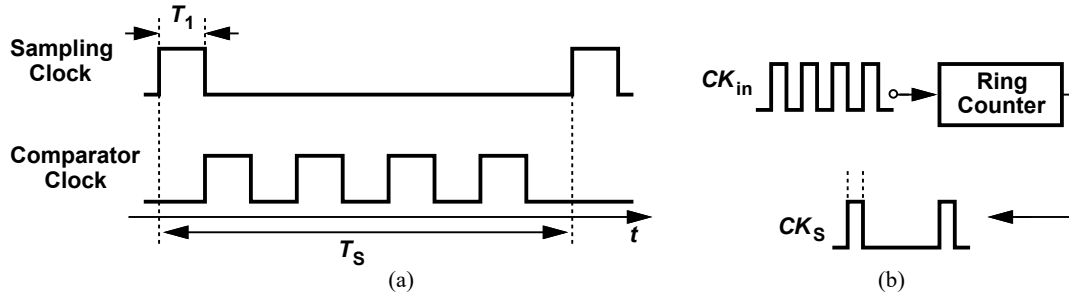


Figure 16.7 (a) Sampling and comparator clocks, and (b) generation of CK_S by means of a ring counter.

Example 16.2

A student proposes the circuit of Fig. 16.8 to generate the sampling clock from an input clock having a 50% duty cycle. Explain the drawbacks of this method.

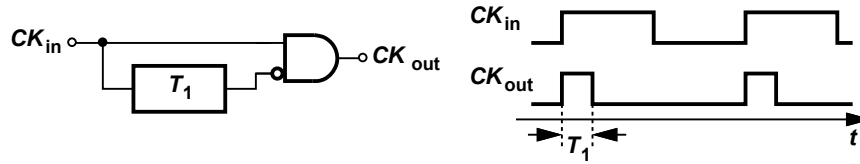


Figure 16.8 Use of a pulse generator for producing a clock.

Solution

The principal issue here is that the delay varies by as much as a factor of 2 across PVT corners. Thus, the time allocation between sampling and conversion departs from optimum. Another difficulty in high-speed, high-resolution designs is that the jitter introduced by the delay stage may be significant.

16.1.2 SAR ADCs with Multibit Quantizers

The speed of SAR ADCs can be improved through the use of multibit quantizers. The architectures studied thus far make a binary (1-bit) decision in each cycle and hence incorporate a single comparator. We surmise that a “finer” (more precise) decision reduces the necessary number of cycles. For example, a generic SAR ADC can digitize the difference between V_{in} and V_{DAC} with *two* bits of resolution, thereby requiring half as many cycles.

Let us ponder how this can be accomplished. In a 1-bit/cycle design, the loop begins with $V_{DAC} = V_{REF}/2$ [Fig. 16.9(a)]. But, rather than decide only whether $V_{in} > V_{REF}/2$ or $V_{in} < V_{REF}/2$, we wish to compare V_{in} to $V_{REF}/4$, $V_{REF}/2$, and $3V_{REF}/4$ in the first cycle. We thus require three comparators (a 2-bit quantizer) and three reference voltages [Fig. 16.9(b)]. As in a flash converter, the comparators are clocked simultaneously. How should we proceed in the next cycle? If we have determined that

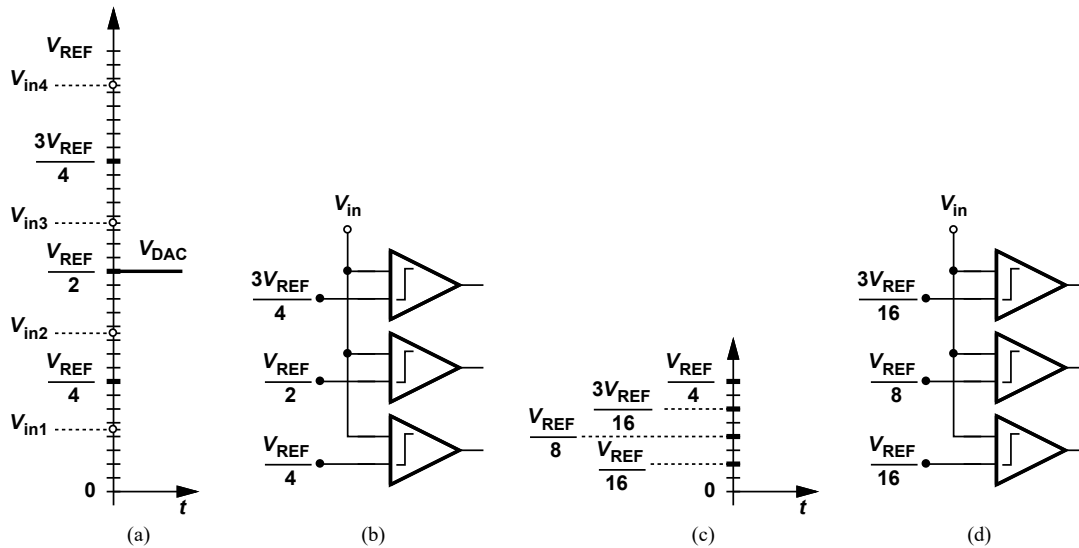


Figure 16.9 (a) Initial DAC voltage in a generic SAR loop, (b) 2-bit flash stage, (c) reference voltages in the second cycle, and (d) reconfiguration of the flash stage.

$0 < V_{in1} < V_{REF}/4$, we must now “zoom in,” subdivide this range, and compare V_{in} to $V_{REF}/16$, $V_{REF}/8$, and $3V_{REF}/16$ [Fig. 16.9(c)]. The second cycle then configures the 2-bit quantizer as shown in Fig. 16.9(d). For the other three input values in Fig. 16.9(a), the second-cycle references are chosen accordingly.

The foregoing thoughts suggest that three comparators can serve as a 2-bit quantizer in each cycle if their reference voltages are changed properly. One approach to realizing such a system is illustrated in Fig. 16.10 [3]. Here, based on the present decision of the 2-bit quantizer, the three DACs generate the reference voltages for the next decision.

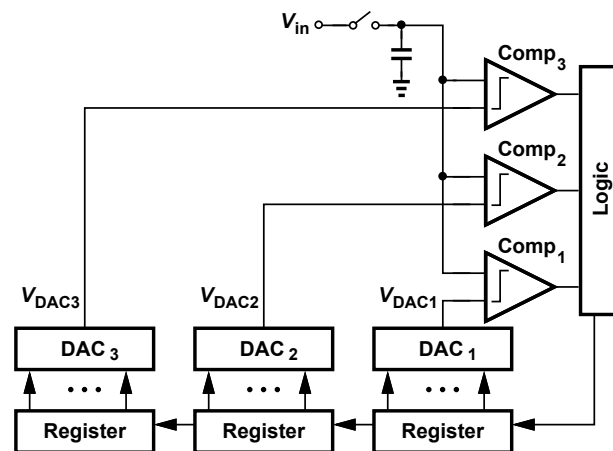


Figure 16.10 A 2-bit-per-cycle SAR loop.

Example 16.3

Suppose the ADC of Fig. 16.10 is designed for a resolution of 4 bits. Sketch the DAC voltages as a function of time for $V_{in1} = 2.2$ LSB, $V_{in2} = 8.3$ LSB, and $V_{in3} = 14.6$ LSB.

Solution

As shown in Fig. 16.11(a), the loop begins with $V_{DAC1} = 4$ LSB, $V_{DAC2} = 8$ LSB, and $V_{DAC3} = 12$ LSB.

For $V_{in1} = 2.2$ LSB, all three comparators generate logical ZEROs (a binary value of 00), indicating that $V_{in1} < 4$ LSB. This result commands the DACs to generate 1 LSB, 2 LSB, and 3 LSB, respectively. Now, the comparators are clocked again, yielding a thermometer code of 110 and hence a binary output equal to 11. The overall digital binary value is obtained by “concatenating” the results of the first and second cycles, emerging as 0011.

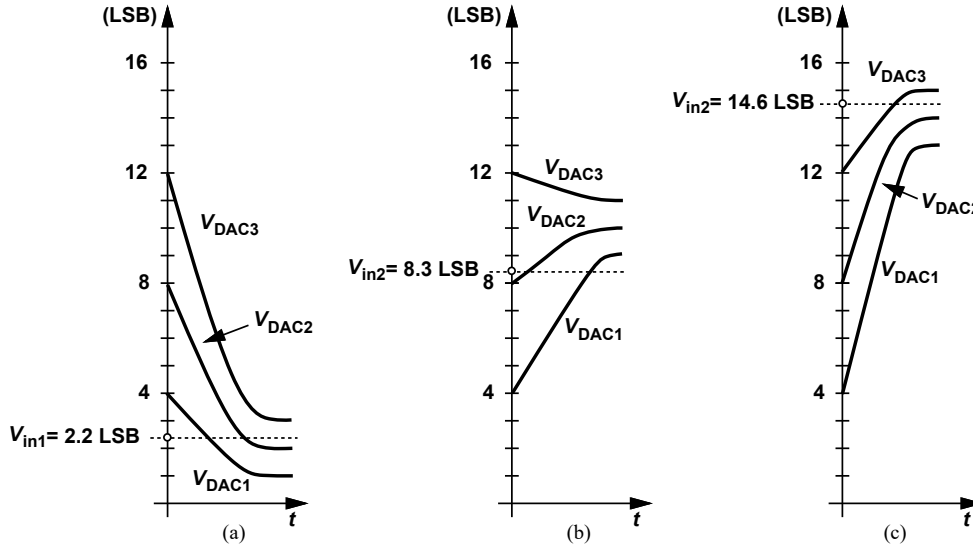


Figure 16.11 Response of 2-bit-per-cycle ADC to (a) $V_{in1} = 2.2$ LSB, (b) $V_{in2} = 8.3$ LSB, and (c) $V_{in3} = 14.6$ LSB.

For $V_{in2} = 8.3$ LSB, only Comp_1 and Comp_2 in Fig. 16.10 generate ONES in the first cycle, demanding that the DACs zoom onto the range between 8 LSB and 12 LSB. Thus, V_{DAC1} , V_{DAC2} , and V_{DAC3} settle to 9 LSB, 10 LSB, and 11 LSB, respectively [Fig. 16.11(b)]. When clocked again, the comparators generate ZEROs. The overall digital output is equal to 1000.

For $V_{in3} = 14.6$ LSB, we obtain the DAC trajectories shown in Fig. 16.11(c), where the loop zooms onto the range between 12 LSB and 16 LSB in the second cycle. The binary output is equal to 1110. We note that each DAC must provide 4 bits of resolution in this example.

The foregoing studies reveal that an N -bit SAR ADC employing a k -bit quantizer must include $2^k - 1$ N -bit DACs. The DACs present three issues: (1) they occupy a large area, (2) they increase the ADC's input capacitance proportionally, and (3) they also increase the logic complexity in the feedback path, raising the loop delay. The last issue implies that the overall conversion speed rises to a lesser extent than theoretically predicted. Let us delve into this point in the context of the 2-bit quantizer example depicted in Fig. 16.10 and with the aid of the waveforms shown in Fig. 16.11. We observe that the registers controlling DAC_1 , DAC_2 , and DAC_3 must provide binary contents equal to $4 \equiv 0100$, $8 \equiv 1000$, and $12 \equiv 1100$, respectively, in the first cycle. After the quantizer is clocked, the three DAC voltages must travel to values that depend on the thermometer code generated by the comparators, i.e., the DAC registers' contents must be updated accordingly. The situation becomes more complex if the SAR loop requires more than two cycles for conversion: the new value of the registers' contents depends on both the present thermometer code and the results of the *previous* cycles. In other words, a given thermometer code value may lead to different DAC outputs in different cycles. These observations suggest greater logic complexity than that of loops operating with a 1-bit quantizer.

Another difficulty facing the ADC of Fig. 16.10 relates to the offset of the comparators. While single-comparator SAR loops can tolerate this effect (Chapter 15), here an offset greater than 1 LSB distorts the thermometer code, introducing DNL and INL. Thus, offset cancellation is typically necessary.

Example 16.4

Suppose Comp_2 in Fig. 16.10 suffers from an offset, $V_{OS} = 1.1 \text{ LSB}$, such that it generates a high level only if $V_{in} - V_{DAC2} > 1.1 \text{ LSB}$. Sketch the DAC voltages versus time if $V_{in} = 8.3 \text{ LSB}$.

Solution

Since both Comp_2 and Comp_3 generate ZEROs in this case, and V_{in} is interpreted to be between 4 LSB and 8 LSB, leading to the waveforms shown in Fig. 16.12. At $t = t_a$, the comparators are clocked again, yielding high outputs and hence a binary value of 11. The overall digital output is equal to 0111, incurring an error of 1 LSB (why?). The reader can determine how much DNL such an offset introduces.

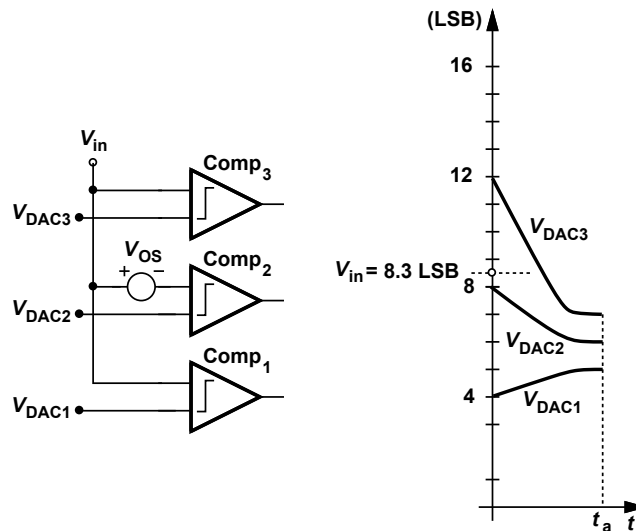


Figure 16.12 Effect of comparator offset on a 2-bit-per-cycle loop.

Considering the waveforms in Fig. 16.11, the reader may wonder whether the necessary DAC voltages can be generated by a single resistor ladder rather than by three capacitor DACs. Such an endeavor would reduce the complexity and the ADC input capacitance. This is indeed possible and illustrated in Fig. 16.13 for 2-bit/cycle operation [4]. Here, each comparator can sense any ladder tap voltage, i.e., the ladder and its switches

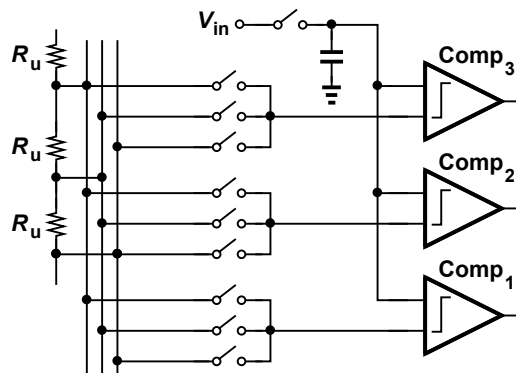


Figure 16.13 Use of a single resistor ladder to generate DAC voltages.

act as three DACs. In a 4-bit ADC, for example, Comp_3 , Comp_2 , and Comp_1 initially sense 12 LSB, 8 LSB, and 4 LSB, respectively. This approach entails three issues: (1) the ladder draws static power; (2) the time

constant at the DAC output nodes is governed by the ladder's Thevenin resistance; the on-resistance of the switches, and the load capacitance; yielding a slow response; and (3) in a fully-differential implementation, the number of switches attached to the ladder is doubled (why?), leading to complex routing.

The second issue mentioned above merits greater attention. For an N -bit ADC, the ladder in Fig. 16.13 consists of 2^N unit resistors, presenting a maximum Thevenin resistance of $(2^N/4)R_u$. To this value we must add the switch on-resistance, R_{on} . The load capacitance, on the other hand, is higher in this case than in a regular RDAC because the number of switches tied to the output node is tripled. The worst-case DAC time constant therefore emerges as

$$\tau_{max} = \left(\frac{2^N}{4} R_u + R_{on} \right) [3 \times 2^N C_{sw} + C_w + C_{comp}], \quad (16.4)$$

where C_{sw} , C_w , and C_{comp} denote the capacitances associated with one switch, the output wire, and the comparator input, respectively.

16.1.3 Hybrid Flash/SAR ADCs

We have seen in Chapter 13 that the flash architecture offers fast conversion, but suffers from higher complexity and power as the resolution increases. We thus surmise that a low-resolution flash stage can significantly improve the speed of a SAR ADC at a moderate cost. In this section, we explore this possibility.

Basic Architecture Figure 16.14(a) conceptually illustrates the idea. After the sampling is completed, the flash stage, also called the flash “sub-ADC,” quantizes the input, generating a digital output, D_F , that sets V_{DAC} to a coarse estimate of V_X [Fig. 16.14(b)]. For example, a 4-bit flash front end can set V_{DAC} to an integer multiple of $V_{REF}/16$. The SAR loop then begins with $|V_X - V_{DAC}| < V_{REF}/16$ and performs its binary search to reach the desired resolution. The flash and SAR stages perform “coarse” and “fine” conversions, respectively.

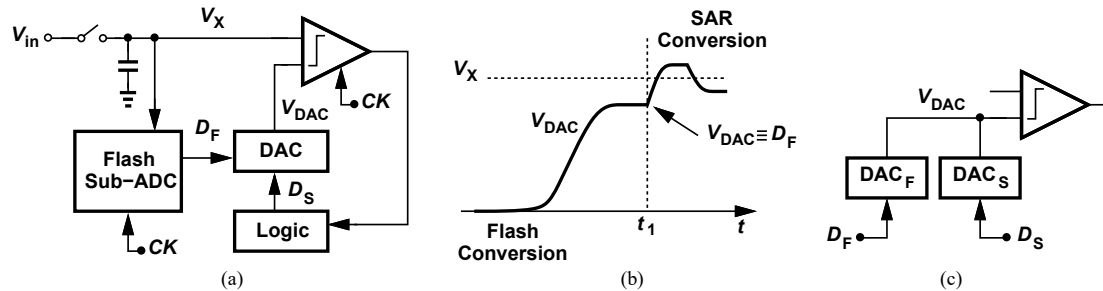


Figure 16.14 (a) Hybrid flash/SAR ADC, (b) DAC voltage waveform, and (c) illustration of two DACs driven by flash and SAR ADCs.

Note that the DAC receives the MSBs from the flash stage and the LSBs from the SAR logic. As depicted in Fig. 16.14(c), the flash and SAR data are applied to their dedicated DACs, with the output nodes shorted so that V_{DAC} represents the concatenation of D_F and D_S . We can view DAC_F and DAC_S as the “coarse” and “fine” sections of a single DAC. In fact, as seen below, we combine the two so that they can act as samplers, too.

Example 16.5

Do we prefer a thermometric or binary implementation for DAC_F in Fig. 16.14(c)?

Solution

We prefer a thermometric array for two reasons. First, as explained in Chapters 10 and 11, such a topology relaxes the matching requirements. Second, the flash stage naturally generates a thermometer code.

As an example, suppose a 10-bit ADC employs a 4-bit flash stage and a 6-bit SAR loop. Then, as illustrated in Fig. 16.15, the flash conversion divides the range into 16 segments, each 64 LSB wide, and decides in which segment V_{in} lies. For example, V_{in} can be between V_j and V_{j+1} , in which case DAC_F generates a voltage equal to V_j as the coarse estimate. The SAR now senses $V_{in1} - V_j$ and digitizes it further. Since V_{in1} can be anywhere between V_j and V_{j+1} , the SAR loop must have a full scale equal to 64 LSB.

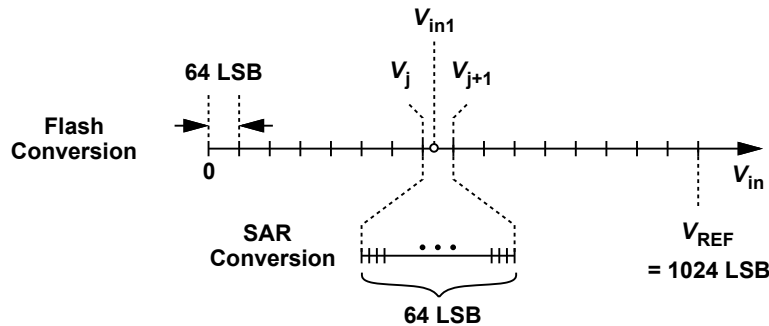


Figure 16.15 Voltage ranges corresponding to flash and SAR conversions.

It is instructive to sketch the timing diagram for the hybrid architecture of Fig. 16.14(a). As shown in Fig. 16.16, while the analog input is sampled, V_{DAC} is set to zero. In the hold mode, the flash stage is clocked and

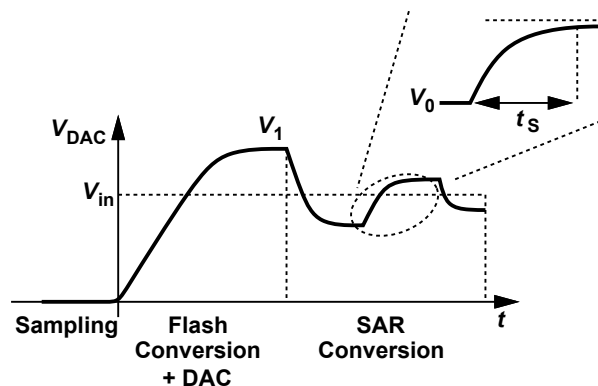


Figure 16.16 DAC waveform during flash and SAR conversions.

its DAC drives V_{DAC} to a coarse estimate of the input (V_1). Next, the SAR loop brings V_{DAC} closer to V_{in} . Interestingly, the DAC requires relatively short settling times during SAR cycles because V_{DAC} changes by only a small amount. Recall from Chapter 15 that the settling time is given by

$$t_S = \tau \ln \frac{V_\infty - V_0}{0.5 \text{ LSB}}, \quad (16.5)$$

suggesting that a 4-bit flash sub-ADC reduces t_S by about a factor of $\ln 16 \approx 2.8$. This speed advantage can be readily exploited by asynchronous operation. Thus, a flash stage followed by an asynchronous SAR loop proves optimum, as the former saves some cycles and shortens the DAC settling while the latter deals with the comparator response time.

As the reader will see, we expend a great deal on the analysis of this architecture because most of the concepts developed here prove useful in our multi-stage ADCs as well.

10-Bit Example Suppose we wish to develop a 10-bit SAR ADC based on top-plate sampling (Chapter 14). The standard loop would take nine cycles for one conversion. Let us instead begin with a flash conversion and resolve 4 bits. The SAR loop then receives this result and takes another five cycles to complete the process.

The overall conversion now requires six cycles. Figure 16.17 depicts the 10-bit ADC's details. Employing 15 comparators, the flash stage digitizes V_{in} , and its thermometer code, representing the four MSBs, controls the bottom plates of 15 unit capacitors in DAC_F . After V_{DAC} settles, the asynchronous SAR loop proceeds for five cycles and computes the six LSBs.

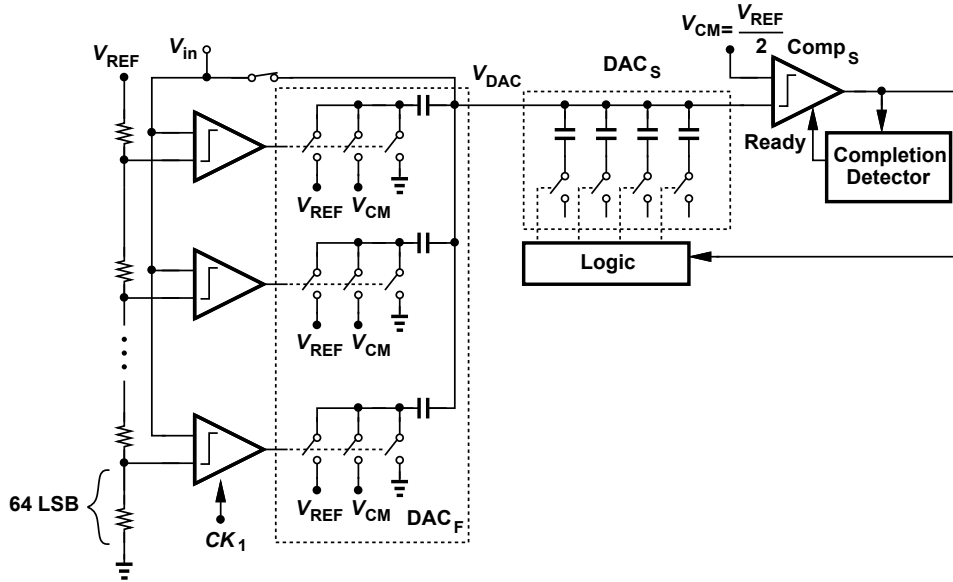


Figure 16.17 Detailed realization of a hybrid flash/SAR ADC.

Example 16.6

How does the overall DAC complexity in Fig. 16.17 compare to that of a standard top-plate-sampling SAR converter?

Solution

Recall from Chapter 14 that such a converter requires a 9-bit DAC for a resolution of 10 bits. Similarly, DAC_S in Fig. 16.17 is configured as a 5-bit array for 6-bit conversion. If realized in binary form, this DAC employs five capacitances equal to $16C_u$, $8C_u$, $4C_u$, $2C_u$, and C_u (Fig. 16.18).² DAC_F is then implemented by 15 capacitors equal to $32C_u$ (why?). The overall DAC consists of $16 \times 32 = 512$ units, the same number as that necessary for a standard 9-bit DAC.

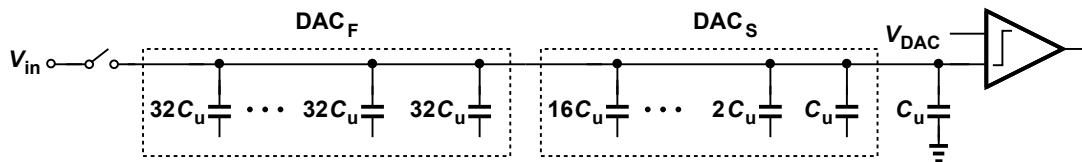


Figure 16.18 DAC implementation example for flash and SAR converters.

Let us revisit the DAC settling behavior of Fig. 16.16 for the 10-bit top-plate-sampling architecture of Fig. 16.17. In this case, V_{DAC} is initially equal to V_{in} , and the flash stage switches the bottom plates of the

²As explained in Chapter 14, a grounded C_u is also necessary so that the minimum change in V_{DAC} is equal to $V_{REF}/2^M$, where M is the desired resolution.

DAC_F capacitors from $V_{CM} = V_{REF}/2$ to V_{REF} or to zero. To study the operation, we consider several input values, namely, $V_{in1} = 60.5$ LSB, $V_{in2} = 520.4$ LSB, and $V_{in3} = 1020.6$ LSB, noting that the flash sub-ADC can resolve levels that are 64 LSB apart.

Before proceeding with the analysis, we make three remarks. First, V_{DAC} in Fig. 16.17 begins with a value equal to V_{in} . Second, after flash and SAR conversions are completed, V_{DAC} approaches $V_{CM} = 512$ LSB. Thus, the flash stage must *raise* V_{DAC} if $V_{in} < 512$ LSB and vice versa. Third, for this to occur, ONEs in the flash thermometer code must switch the left plates of their corresponding DAC capacitors from $V_{CM} = V_{REF}/2$ to zero. For example, if this code contains mostly ONEs (and hence V_{in} is near the full scale), then V_{DAC} must fall.

Suppose the thermometer code provides k ONEs and hence $15 - k$ ZEROs. As illustrated in Fig. 16.19, the left plates of k capacitors swing from $V_{REF}/2$ to zero, and the rest from $V_{REF}/2$ to V_{REF} . Applying

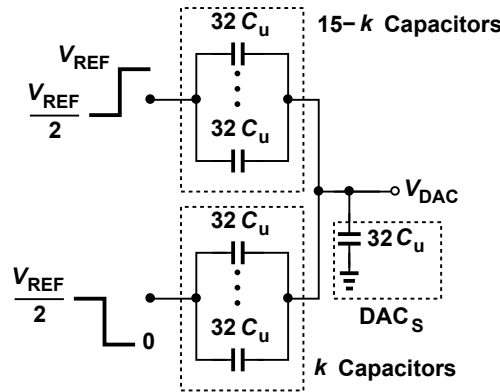


Figure 16.19 Equivalent circuit of the flash and SAR DACs.

superposition, we then write the voltage produced by DAC_F as

$$V_{DAC} = V_{in} - \frac{k \times 32C_u}{15 \times 32C_u + 32C_u} \frac{V_{REF}}{2} + \frac{(15 - k) \times 32C_u}{15 \times 32C_u + 32C_u} \frac{V_{REF}}{2} \quad (16.6)$$

$$= V_{in} + \frac{15 - 2k}{32} V_{REF}. \quad (16.7)$$

We now consider different input levels.

For $V_{in1} = 60.5$ LSB, the flash stage in Fig. 16.17 produces a zero thermometer code, requiring that all of the DAC_F capacitors' bottom plates swing from $V_{REF}/2$ to V_{REF} [Fig. 16.20(a)]. Consequently, V_{DAC} rises by $[15 \times 32C_u / (15 \times 32C_u + 32C_u)](V_{REF}/2) = (15/16)(V_{REF}/2) = 480$ LSB, reaching 540.5 LSB. Since this value exceeds $V_{CM} = V_{REF}/2$, the SAR comparator commands the loop to switch the bottom plate of the $16C_u$ capacitor in Fig. 16.18 from $V_{REF}/2$ to zero [Fig. 16.20(b)]. Now, V_{DAC} drops by $[16 / (16 \times 32)](V_{REF}/2) = 16$ LSB [Fig. 16.20(c)]. In the next three cycles, V_{DAC} falls by 8 LSB, 4 LSB, and 2 LSB. Then, the DAC settles for two more cycles, and the SAR comparator is clocked once more at the end.³ The reader is encouraged to find the digital output in this case.

We next consider the case of $V_{in2} = 520.4$ LSB, recognizing that eight of the flash comparators in Fig. 16.17 generate ONEs, and the other seven produce ZEROs. Figure 16.20(e) yields $V_{DAC} = V_{in} - 32$ LSB, which amounts to 488.4 LSB. The SAR loop then introduces changes equal to +16 LSB, +8 LSB, -4 LSB, +2 LSB, and +1 LSB in the DAC output.

For the third case, where $V_{in3} = 1020.6$ LSB, the flash stage switches all of the DAC_F bottom plates from $V_{REF}/2$ to zero, forcing V_{DAC} to drop by 480 LSB [Fig. 16.20(f)]. The subsequent binary search proceeds

³In an asynchronous SAR loop, some cycles take longer than others.

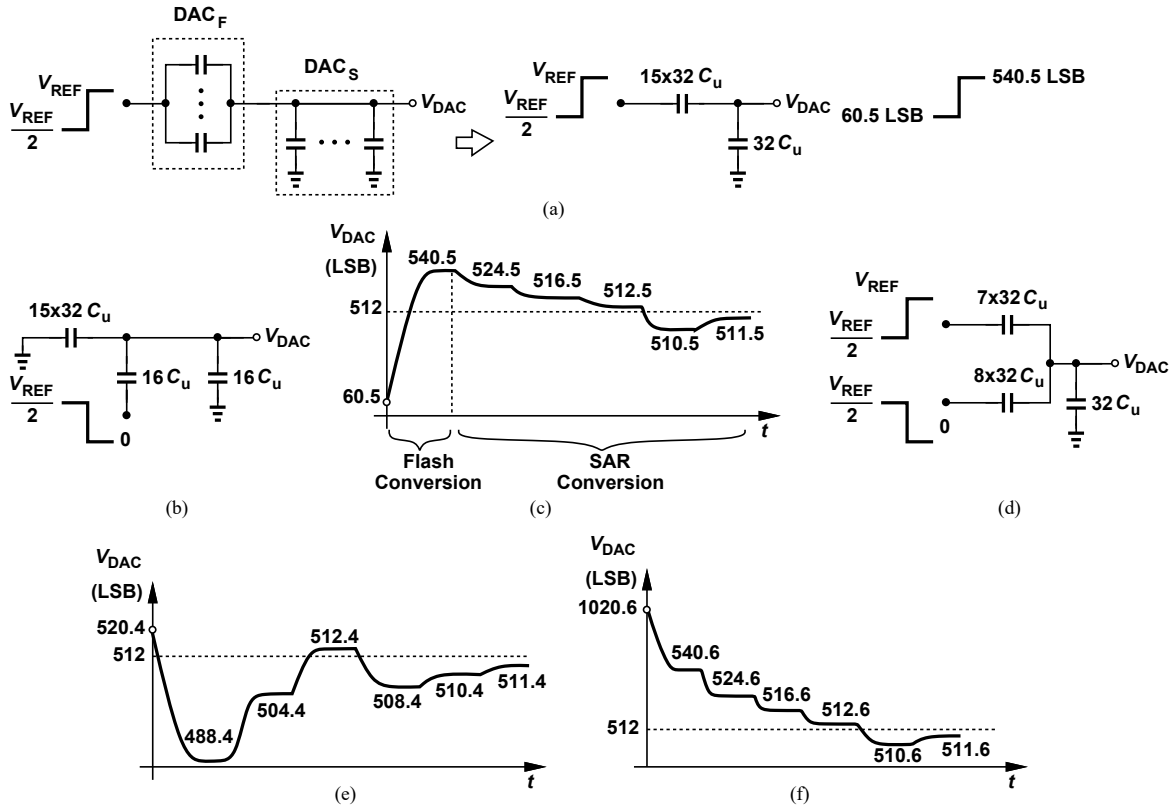


Figure 16.20 (a) DAC configurations for $V_{in1} = 60.5$ LSB, (b) simplified circuit, (c) DAC voltage vs. time, (d) equivalent circuit for $V_{in2} = 520.4$ LSB, (e) corresponding DAC voltage vs. time, and (f) DAC voltage for $V_{in3} = 1020.6$ LSB.

for five cycles, bringing V_{DAC} to 511.6 LSB. We note that, in all three cases, V_{DAC} settles to within 1 LSB of 512 LSB. Also, even though the final V_{DAC} values in different cases may be equal, the digital outputs are not because the SAR code is given by only the *jumps* in V_{DAC} .

Example 16.7

Repeat the above analysis for $V_{in4} = 192.5$ LSB.

Solution

With this input level, the three bottommost flash comparators in Fig. 16.17 generate ONEs and the rest, ZEROs [Fig. 16.21(a)]. The bottom plates accordingly switch to zero or V_{REF} [Fig. 16.21(b)], yielding a change of $+ [9 \times 32 C_u / (512 C_u)] (V_{REF} / 2) = +288$ LSB in V_{DAC} [Fig. 16.21(c)]. The SAR loop then increments V_{DAC} in binary steps up to a value of 511.5 LSB.

As mentioned earlier, the flash stage in Fig. 16.17 performs “coarse” conversion, narrowing down the input to a “subrange” of 64 LSB (Fig. 16.22). We call D_F the “coarse estimate.” The SAR loop then carries out “fine” conversion and provides the “fine estimate,” D_S . To obtain the overall ADC characteristic, we “stitch” copies of the fine characteristic: it is copied and shifted so that it begins at every coarse code transition.

Static Nonidealities The hybrid architecture of Fig. 16.17 assumes that the flash stage makes a correct decision before delivering the coarse estimate to the SAR loop. In practice, two static phenomena can invalidate this assumption.

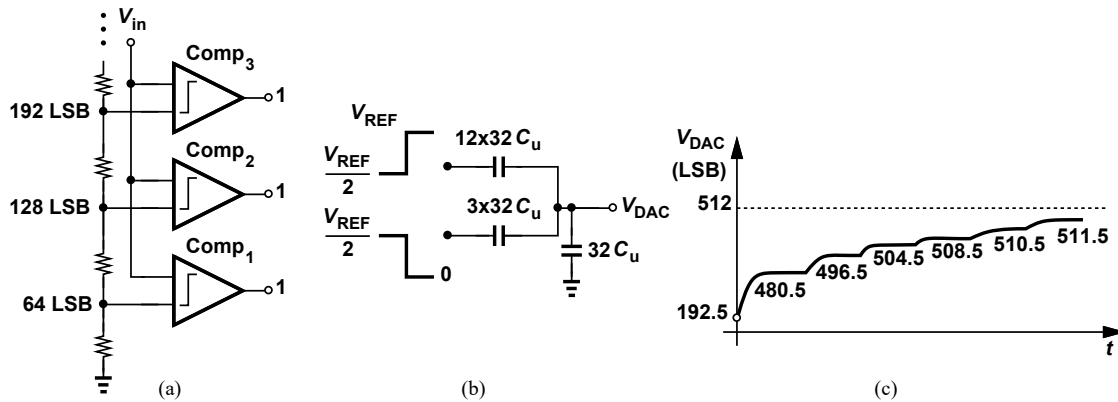


Figure 16.21 (a) Section of a flash stage, (b) simplified DAC operation, and (c) DAC voltage vs. time.

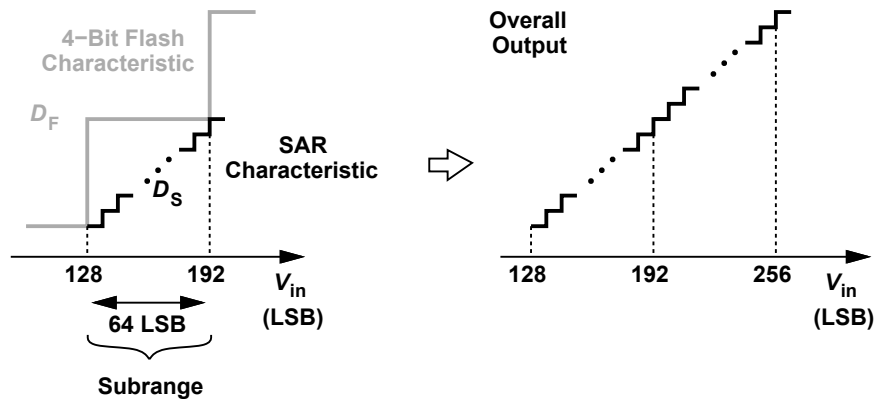


Figure 16.22 Illustration of obtaining the overall ADC characteristic.

First, let us assume that the offset of one of the flash comparators is greater than 1 LSB, e.g., greater than $V_{REF}/1024 \approx 1 \text{ mV}$ in a 10-bit system with $V_{REF} \approx 1 \text{ V}$. This is likely to occur as it is difficult to reduce the 3σ offset to less than 1 mV.⁴ Suppose, for example, $V_{OS} = 1.1 \text{ LSB}$ in Fig. 16.23(a), such that Comp_3 does not change its output unless V_{in} exceeds 193.1 LSB. Then, only Comp_1 and Comp_2 generate ONes for $V_{in} = 192.5 \text{ LSB}$. Consequently, only two capacitors in DAC_F switch from $V_{REF}/2$ to zero and the rest to V_{REF} [Fig. 16.23(b)], yielding a change of $+(11 \times 32/512)(V_{REF}/2) = +352 \text{ LSB}$ in V_{DAC} . As shown in Fig. 16.23(c), the SAR loop reduces V_{DAC} to 513.5 LSB. This trajectory and its final value should be compared to those in Fig. 16.21(c).

Example 16.8

A student reasons that the comparator offset in Fig. 16.23(a) translates to a DNL of 2 LSB because the final V_{DAC} values in Figs. 16.23(c) and 16.21(c) differ by this amount. Is the student's reasoning correct?

Solution

No, it is not. Even without an offset, the final DAC value still has a “margin” of 1 LSB. To determine the DNL, we must find the consecutive input values at which the digital output changes. We then ask, how much can

⁴Unless offset cancellation is used.

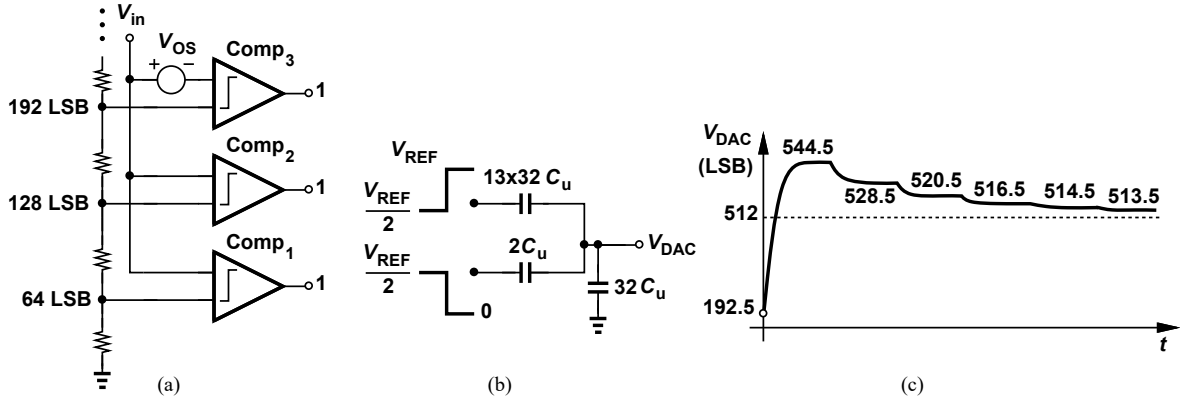


Figure 16.23 (a) Flash stage with comparator offset, (b) configuration of DACs, and (c) DAC voltage vs. time.

V_{in} depart from 192.5 LSB before the digital output changes? We observe in Fig. 16.23(a) that if V_{in} changes from 192.5 LSB to slightly below $192 \text{ LSB} + V_{OS} = 193.1 \text{ LSB}$, the flash stage digital output does not change, the shape of the trajectory remains the same, V_{DAC} settles to 514.1 LSB, and the SAR digital output is also constant. (Recall that the SAR code is given by the jumps in V_{DAC} rather than by its final value.) This is where one code transition is about to occur. To compute the lower code transition, we reduce V_{in} and seek a change in the SAR output. This takes place if V_{DAC} settles to slightly *below* 512 LSB in Fig. 16.23(c), requiring $V_{in} = 191 \text{ LSB}$. Thus, the DNL is equal to $193.1 \text{ LSB} - 191 \text{ LSB} - 1 \text{ LSB} = 1.1 \text{ LSB} = V_{OS}$. The key point here is that the flash stage requires high-precision comparators even though it provides a resolution of only 4 bits.

The above analysis can also be performed directly on the coarse and fine characteristics shown in Fig. 16.22. Depicted in Fig. 16.24(a), the former experiences a DNL error equal to V_{OS} . We must now superimpose the 6-bit SAR characteristic on the coarse one. As illustrated in Fig. 16.24(b), a gap appears between the end of one SAR characteristic, at $V_{in} = 192 \text{ LSB}$, and the beginning of the next, at $V_{in} = 192 \text{ LSB} + V_{OS}$. This gap signifies DNL. This method of manipulating characteristics proves crucial in the analysis of other ADC architectures as well (Chapter 18). The reader can repeat the study for the case of $V_{OS} = -1.1 \text{ LSB}$.

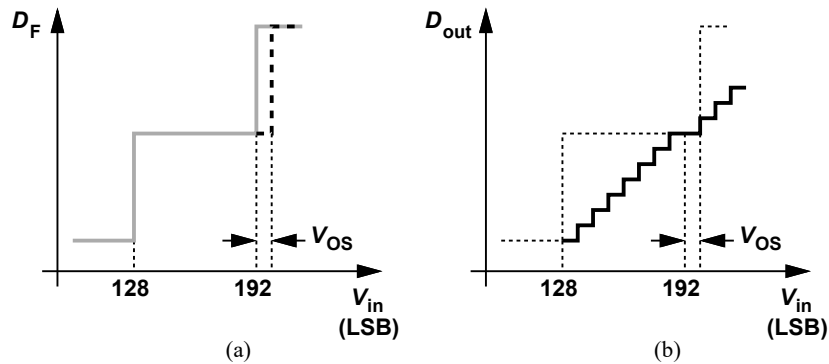


Figure 16.24 (a) Effect of offset on flash stage response, and (b) overall ADC characteristic.

Another static imperfection in the hybrid architecture of Fig. 16.17 arises from the offset of the SAR comparator, Comp_S . Recall from Chapter 15 that this offset shifts the characteristic of simple SAR loops [Fig. 16.25(a)], introducing no DNL or INL. Note that V_{FS1} is the nominal full scale of the SAR stage. In the hybrid ADC, on the other hand, this offset proves serious. To investigate the issue, let us construct the overall characteristic by stitching copies of that in Fig. 16.25(a). Depicted in Fig. 16.25(b), the result exhibits

a DNL error equal to V_{OS} at the coarse transition points—and a missing code if $V_{OS} > 1 \text{ LSB}$. Thus, the SAR comparator offset must also remain well below 1 LSB.

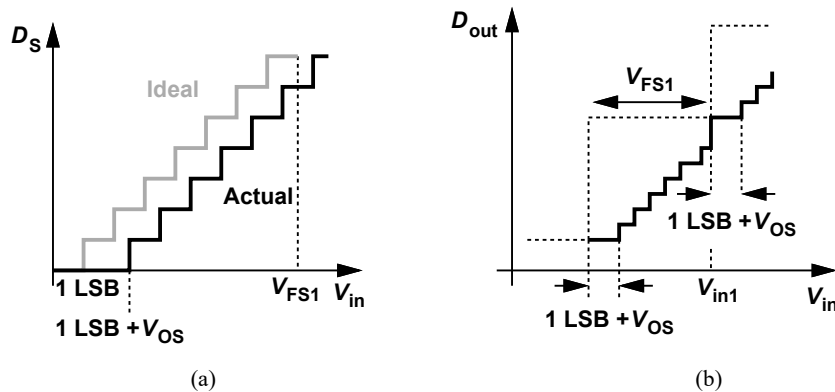


Figure 16.25 Effect of offset on (a) SAR stage, and (b) overall ADC characteristic.

Example 16.9

In Fig. 16.25(a), the SAR ADC generates a greater digital output if V_{in} is slightly higher than V_{FS1} , whereas in Fig. 16.25(b), it does not. Explain why.

Solution

In the hybrid architecture, as V_{in} exceeds V_{in1} , the flash converter increments its output, thereby presenting a small difference to the SAR loop. That is, for no input value does the SAR ADC receive a value greater than V_{FS1} .

The SAR comparator offset proves serious. It is possible to incorporate “overlap” and “redundancy” to alleviate this issue, as explained in Chapter 19.

Dynamic Nonidealities The flash and SAR decisions in Fig. 16.17 can become inconsistent due to dynamic errors, particularly, at the end of the sampling mode. Let us draw the input samplers of the comparators (Chapter 13) and the DAC capacitors as shown in Fig. 16.26(a). To sample V_{in} , we turn off S_a and S_b .⁵ How do we ensure that the voltages stored on C_a and on the DAC capacitors are equal? Figure 16.26(b) depicts

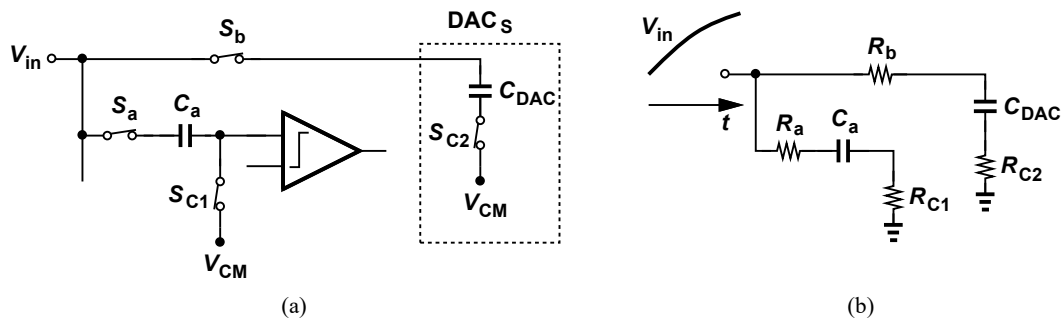


Figure 16.26 (a) Sampling operation of flash and SAR stages, and (b) equivalent circuit.

⁵As explained in Chapter 14, we prefer to release the bottom plates first so as to avoid input-dependent charge injection, but this is not possible in the top-plate-sampling architecture.

the simplified network in the sampling mode, modeling the switches by resistances and lumping the entire DAC_S into one capacitor and one switch. We observe that C_a and C_{DAC} sustain equal voltages if the two branches' time constants are equal:

$$(R_a + R_{c1})C_a = (R_b + R_{c2})C_{DAC}. \quad (16.8)$$

This is accomplished by properly scaling the comparator sampling devices with respect to those of the DAC. Specifically, denoting the aspect ratios of S_b and S_{c2} in Fig. 16.26(a) by $(W/L)_b$ and $(W/L)_{c2}$, respectively, and assuming that the DAC contains m units, we can select $(W/L)_a = (W/L)_b/m$, $C_a = C_{DAC}/m$, and $(W/L)_{c1} = (W/L)_{c2}/m$. If these relationships yield excessively large or small sampling devices for the comparator, further scaling may be necessary, e.g., C_a and its switches can be smaller or larger.

Despite proper scaling, the two sampling paths in Fig. 16.26(a) may suffer from random mismatches. This issue becomes critical for resolutions of 10 bits and higher.

Overlap and Redundancy In order to relax the budget for static and dynamic errors, we can create some “overlap” between the digitization ranges of the flash and SAR converters. Such a remedy requires “redundancy,” i.e., additional hardware or conversion time. As mentioned earlier, we defer this important concept to Chapter 19.

16.2 DAC Complexity Reduction

We have seen in Chapters 11 and 14 that capacitor DACs become complex for high resolutions. The capacitors, the switches, and the routing of the digital signals and supply rails lead to large area consumption and severe symmetry and coupling issues. We recognized in Chapter 14 that top-plate sampling lowers the necessary DAC resolution by 1 bit, halving the required area. Moreover, as described in Chapter 11, the complexity can be substantially reduced through the use of “splitting,” but the parasitics of the splitting capacitor limit the precision. In this section, we study other methods.

16.2.1 Monotonic Switching

Differential SAR ADCs incorporate two CDACs for sampling and processing the input. In our analysis of such ADCs in Chapter 14, we have assumed that the comparator decision controls both DACs in each cycle. Interestingly, if we avoid this condition, the DAC complexity can be halved.

Consider the SAR loop shown in Fig. 16.27(a), where the inputs are applied to the top plates while the bottom plates are tied to V_{REF} . Each DAC has a resolution of 4 bits. Suppose, for example, that $V_{in}^+ = 14.1$ LSB and $V_{in}^- = 1.8$ LSB, where $1 \text{ LSB} = V_{REF}/16$. When the circuit enters the hold mode and the comparator is clocked, the loop decides whether $V_{DAC}^+ > V_{DAC}^-$. In our previous studies, we reduced this difference in the next cycle by allowing V_{DAC}^+ to fall and V_{DAC}^- to rise. Alternatively, we can simply lower V_{DAC}^+ in subsequent cycles until it drops below V_{DAC}^- , then lower V_{DAC}^- until it drops below V_{DAC}^+ , etc. As illustrated in Fig. 16.27(b) [5], the bottom plate of the MSB capacitor in DAC^+ switches from V_{REF} to zero, yielding an 8-LSB downward jump in V_{DAC}^+ while DAC^- has no activity. The next two comparator decisions cause 4-LSB and 2-LSB drops in V_{DAC}^+ . At $t = t_a$, the comparator informs the loop that $V_{DAC}^+ < V_{DAC}^-$, demanding that V_{DAC}^- fall. But by how much? Since the last change in $|V_{DAC}^+ - V_{DAC}^-|$ was equal to 2 LSB, binary search prescribes that V_{DAC}^- change by 1 LSB. At $t = t_b$, the comparator is clocked for the last time. The binary output is equal to 11101. Note that the loop requires four DAC settling times and two 4-bit DACs for 5 bits of resolution.

Example 16.10

Repeat the foregoing analysis if $V_{in}^+ = 9.1$ LSB and $V_{in}^- = 6.9$ LSB.

Solution

After the comparator is clocked, V_{DAC}^+ falls by 8 LSB (Fig. 16.28). Next, V_{DAC}^- drops by 4 LSB, 2 LSB, and 1 LSB. The comparator is clocked for the last time at $t = t_1$.

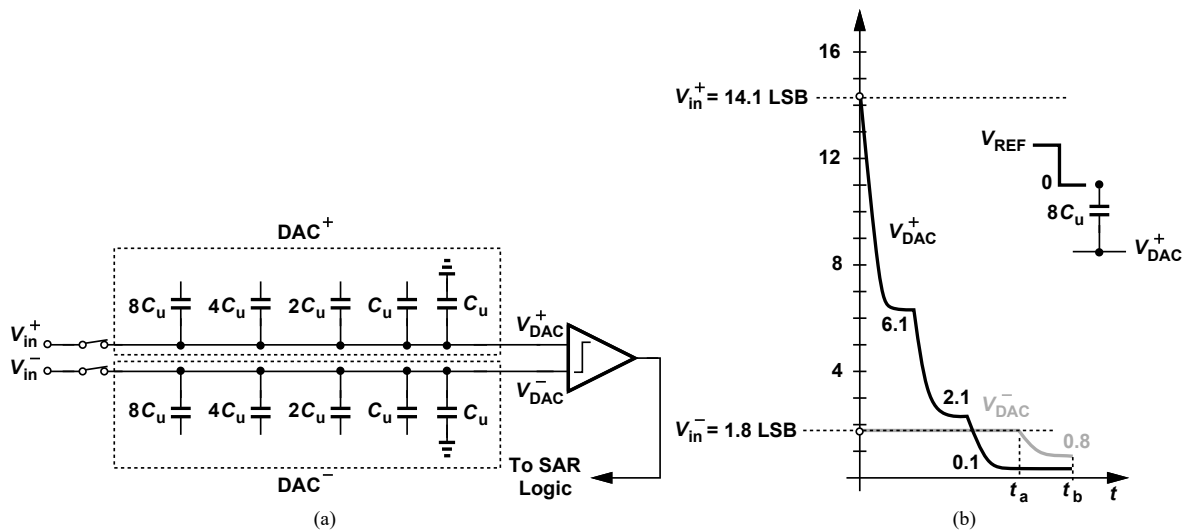


Figure 16.27 (a) SAR loop with top-plate sampling, and (b) DAC voltages vs. time.

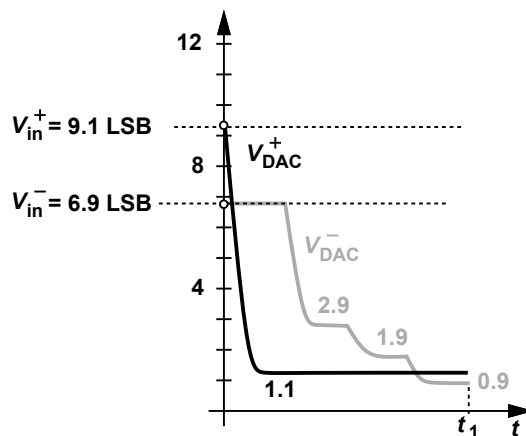


Figure 16.28 DAC voltages vs. time.

Called the “monotonic-switching” SAR ADC, the architecture of Fig. 16.27(a) presents advantages similar to those of the differential SAR loop studied in Chapter 14. Both allow a 1-bit reduction in the DAC resolution and both require $N - 1$ DAC settling times for an N -bit conversion. However, there are some important differences between the two. In the nonmonotonic switching structure, the comparator input CM level, $V_{CM,in}$, remains constant because V_{DAC}^+ and V_{DAC}^- vary differentially, whereas in the monotonic switching waveforms studied above, the CM level changes considerably. This behavior entails three issues: (a) the speed of the StrongArm comparator degrades at one extreme of the input CM level (Chapter 5); (b) the comparator offset also varies with $V_{CM,in}$ (Chapter 6); and (c) the junction capacitance of the input switches and nonlinear input capacitance of the comparator introduce distortion. From these perspectives, the monotonic-switching architecture is inferior.

Another difference between the two architectures relates to the logic necessary for the bottom-plate switches. The nonmonotonic loop requires that the bottom plates switch from $V_{CM} = V_{REF}/2$ to V_{REF} or zero, whereas monotonic switching conditionally swings the plates from V_{REF} to zero. The latter’s logic

must update the DAC^+ register or the DAC^- register depending on the comparator decision. The logic complexity levels are therefore comparable.

One advantage of monotonic switching concerns the DAC power consumption. As the capacitors charge and discharge, the switches in series with them burn power. In the simple branch shown in Fig. 16.29, as C_1 charges up from zero to V_1 and stores energy in the amount of $(1/2)C_1V_1^2$, the resistor dissipates an equal amount as well. This can be seen by proving that the energy drawn from V_1 is equal to $C_1V_1^2$. The monotonic-switching architecture lowers the DAC power consumption by about a factor of five [5] because it avoids upward transitions.

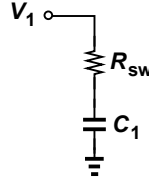


Figure 16.29 Equivalent circuit for energy calculation.

16.2.2 Three-Level Bottom-Plate Sampling

For resolutions of 12 bits and above, top-plate sampling may not provide the requisite linearity even with switch bootstrapping. For example, the design study in Chapter 4 suggests a third harmonic at -62 dB for a circuit running at 5 GHz. Thus, we turn to bottom-plate sampling for higher resolutions.

As a reminder from Chapter 14, we should clearly distinguish between top-plate and bottom-plate sampling. Shown in Fig. 16.30(a), the former is defined as a structure in which the switch sensing *the input* turns off first. Here, the input-dependent charge injected by S_1 introduces distortion. Bottom-plate sampling, on the other hand, is defined as an arrangement where the switch tied to a *constant* voltage turns off first [Fig. 16.30(b)]. In this case, the charge injected by S_2 onto C_1 is independent of V_{in} , creating only an offset (and hence a common-mode jump in differential configurations). It is shown in Chapter 15 that the charge injection of S_1 does not affect the voltage across C_1 . In SAR ADCs, we draw this circuit as shown in Fig. 16.30(c), noting that the “bottom” plate is drawn on top.

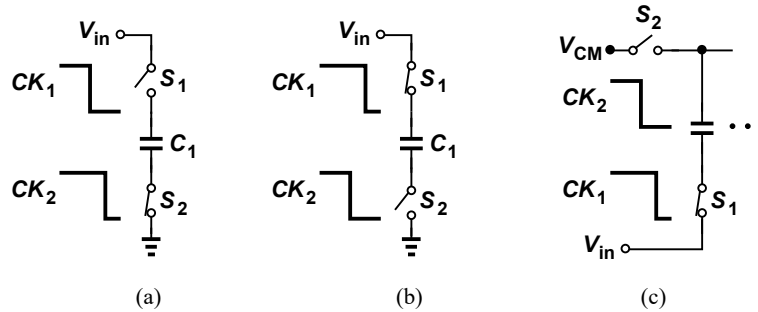


Figure 16.30 (a) Top-plate sampler, (b) bottom-plate sampler, and (c) use of the latter in a SAR ADC.

We wish to explore the possibility of increasing the resolution of bottom-plate-sampling DACs for a given number of unit capacitors. Our studies in Chapter 14 have assumed that the bottom plates can swing to only V_{REF} or zero. But what if we allow switching to $V_{REF}/2$ as well? The finer V_{DAC} jumps thus created can raise the resolution by 1 bit [6].

Consider the architecture depicted in Fig. 16.31(a) [6], where the bottom plates are initially connected to V_{in} and S_0 is on. We show that the 3-bit capacitor DAC provides 4 bits of resolution if these plates can swing to V_{REF} , $V_{REF}/2$, or zero. At the end of the sampling mode, S_0 turns off first and all of the bottom plates swing to $V_{REF}/2$. Thus, V_{DAC} changes from $V_{REF}/2$ by an amount equal to $V_{REF}/2 - V_{in}$, reaching $V_{REF} - V_{in}$ [Fig. 16.31(b)]. The comparator now determines whether V_{DAC} is greater or less than $V_{REF}/2$, in essence finding the polarity of $V_{REF} - V_{in} - V_{REF}/2 = V_{REF}/2 - V_{in}$. Based on this decision, the

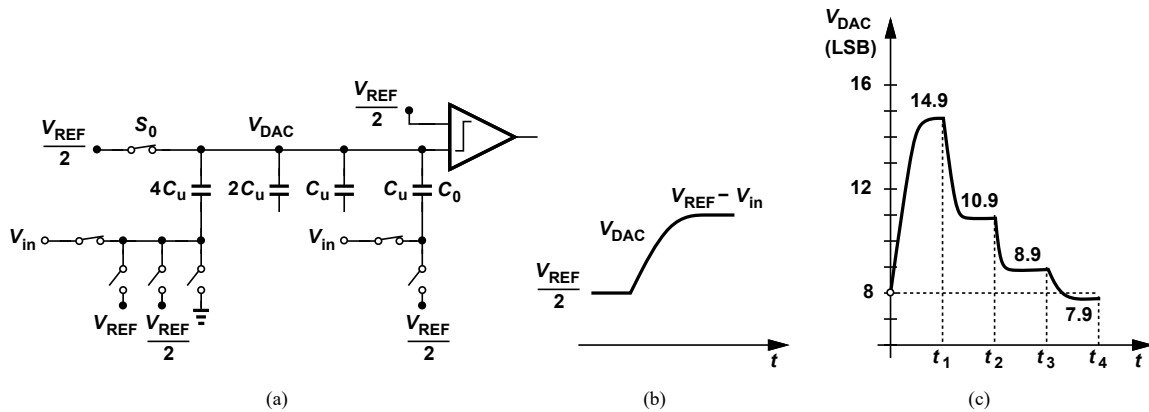


Figure 16.31 (a) Use of $V_{REF}/2$ switching to increase the resolution, (b) example of DAC switching, and (c) DAC voltage vs. time.

bottom plate of the MSB capacitor swings from $V_{REF}/2$ to zero or V_{REF} and changes V_{DAC} by $-V_{REF}/4$ or $+V_{REF}/4$, respectively. The comparator input difference, $V_{REF}/4 - V_{in}$ or $3V_{REF}/4 - V_{in}$, enables the next step in the binary search.

To examine the operation more closely, suppose V_{in} in Fig. 16.31(a) is equal to 1.1 LSB = $1.1V_{REF}/16$. Plotted in Fig. 16.31(c), V_{DAC} begins at 8 LSB and travels to $V_{REF} - V_{in} = 14.9$ LSB. At $t = t_1$, the comparator decides that $V_{DAC} > V_{REF}/2$, commanding the bottom plate of $4C_u$ to jump to zero. This is followed by two more decisions at t_2 and t_3 , bringing V_{DAC} to 7.9 LSB. At $t = t_4$, the comparator is clocked for the last time. We note that the DAC settles four times here.

Example 16.11

Repeat the foregoing analysis if $V_{in} = 7.2$ LSB.

Solution

As shown in Fig. 16.32, V_{DAC} first rises to $V_{REF} - V_{in} = 8.8$ LSB. At $t = t_1$, the comparator causes a 4-LSB fall in V_{DAC} . Next, V_{DAC} rises by 2 LSB and 1 LSB. The last comparator decision occurs at $t = t_4$.

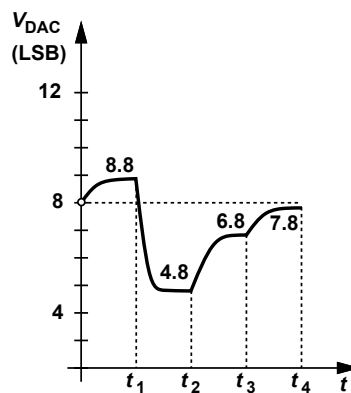


Figure 16.32 DAC voltage vs. time.

The three-level sampling technique depicted in Fig. 16.31(a) maintains the same input common-mode level for the comparator for different cycles, an advantage over the monotonic switching ADC of Fig. 16.27.

Nonetheless, three-level bottom-plate switching requires N DAC settling times for an N -bit conversion, a disadvantage.

We must answer two questions in regards to the circuit that generates $V_{REF}/2$. First, must $V_{REF}/2$ be very precise? To investigate this point, suppose the circuit delivers a voltage equal to $V_{REF}/2 + \Delta V$. Noting that, in Fig. 16.31(a), this value is applied to the comparator, to the left side of S_0 , and to the bottom plates, we obtain $V_{DAC} = 2(V_{REF}/2 + \Delta V) - V_{in} = V_{REF} - V_{in} + 2\Delta V$ in the first cycle (why?). The comparator then decides whether $V_{DAC} - (V_{REF}/2 + \Delta V) = V_{REF}/2 + \Delta V - V_{in}$ is positive or negative.

We may surmise that ΔV introduces an offset in the overall ADC characteristic, i.e., it can be simply subtracted from V_{in} . But that is not so! Let us consider the next SAR cycle. If the bottom plate of $4C_u$ jumps to zero [Fig. 16.33(a)], we have $V_{DAC} = V_{REF} + 2\Delta V - V_{in} - (V_{REF}/2 + \Delta V)/2 = 3V_{REF}/4 + 3\Delta V/2 - V_{in}$. The comparator then detects the polarity of $V_{DAC} - (V_{REF}/2 + \Delta V) = V_{REF}/4 + \Delta V/2 - V_{in}$. We conclude

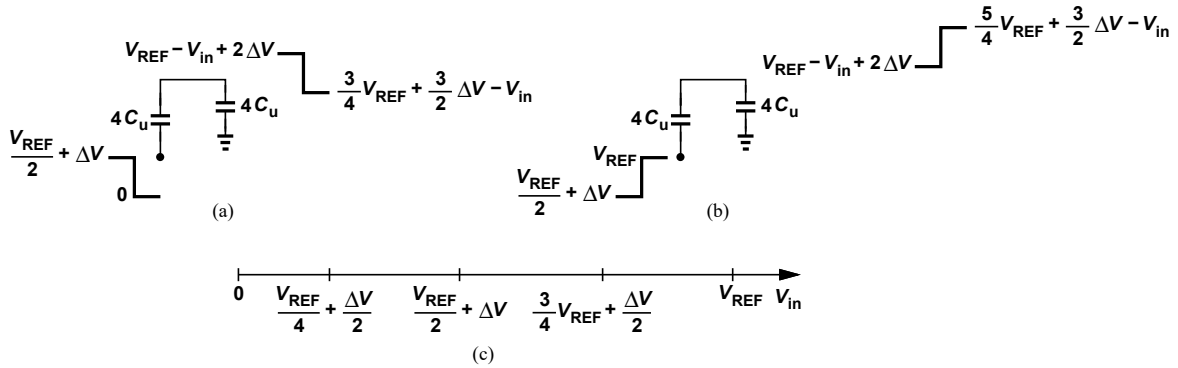


Figure 16.33 (a) Response of DAC to downward step, (b) response of DAC to upward step, and (c) resulting transition points.

that ΔV does not simply appear in series with V_{in} . Can we add ΔV to V_{REF} and hence view its effect as a gain error? No, we cannot. If the bottom plate of $4C_u$ in Fig. 16.31(a) swings to V_{REF} [Fig. 16.33(b)], we obtain $V_{DAC} = 2(V_{REF}/2 + \Delta V) - V_{in} + (V_{REF} - V_{REF}/2 - \Delta V)/2 = 5V_{REF}/4 + 3\Delta V/2 - V_{in}$. The comparator thus finds a difference equal to $3V_{REF}/4 + \Delta V/2 - V_{in}$. Figure 16.33(c) depicts the resulting decision thresholds, revealing that ΔV translates to nonlinearity. In other words, $V_{REF}/2$ must be generated with an error well below 1 LSB, e.g., by means of a precise resistive divider (which consumes power). The reader is encouraged to repeat this exercise for a differential topology.

The second question is, must the circuit generating $V_{REF}/2$ provide a low output impedance? In the first conversion cycle, all of the bottom plates switch to this voltage. As shown in Fig. 16.34(a), the $V_{REF}/2$ generator sees the entire DAC capacitance *in series* with the comparator input capacitance.⁶ It follows that this cycle benefits from fast settling due to the small net capacitance. In the next cycle, however, the MSB capacitor switches to zero or V_{REF} , introducing a time constant proportional to $2C_u$, the on-resistance of the switches in series with the capacitors, and the output impedances of both V_{REF} and $V_{REF}/2$ generators [Fig. 16.34(b)].

Example 16.12

Can an asynchronous SAR loop take advantage of the faster DAC settling in the foregoing ADC's first cycle?

Solution

No, it cannot. Recall from Section 16.1.1 that asynchronous operation proves useful only in terms of the comparator response. Such a loop must still allow a fixed delay equal to the worst-case DAC settling time.

⁶And the parasitics of S_0 .

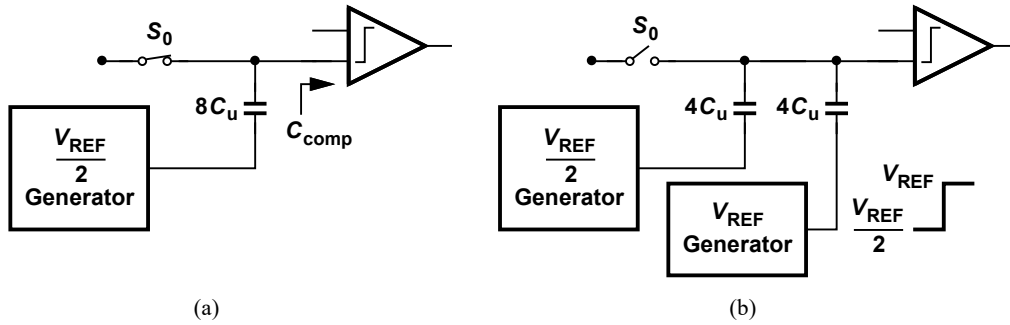


Figure 16.34 DAC with (a) $V_{REF}/2$ generator, and (b) both $V_{REF}/2$ and V_{REF} generators.

References

- [1]. S. Chen and R. Brodersen, "A 6-bit 600-MS/s 5.3-mW asynchronous ADC in 0.13-um CMOS," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 2669-2680, Nov. 2006.
- [2]. D. Bankman et al, "Understanding metastability in SAR ADCs: Part II: Asynchronous," *IEEE Solid-State Circuits Magazine*, vol. 11, pp. 16-32, Summer 2019.
- [3]. Z. Cao, S. Yan, and Y. Li, "A 32 mW 1.25 GS/s 6b 2b/step SAR ADC in 0.13 um CMOS," *IEEE Journal of Solid-State Circuits*, vol. 44, pp. 862-873, March 2009.
- [4]. H. Wei, et al, "An 8-b 400-MS/s 2-b-per-cycle SAR ADC with resistive DAC," *IEEE Journal of Solid-State Circuits*, vol. 47, pp. 2763-2772, Nov. 2012.
- [5]. C. Liu, et al, "A 10-bit 50-MS/s SAR ADC with a monotonic capacitor switching procedure," *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 731-740, April 2010.
- [6]. S. Cho, et al, "A 550-uW 10-b 40-MS/s SAR ADC with multistep addition-only digital error correction," *IEEE Journal of Solid-State Circuits*, vol. 46, pp. 1881-1892, Aug. 2011.

Problems

- 16.1. In Fig. 16.2(b), can we tie the inputs of the NAND gate to the RS latch outputs? Explain.
- 16.2. Considering that the arrangement in Fig. 16.2(c) can use either a NAND gate or a NOR gate, combine this circuit with that in Fig. 16.2(b) and simplify the logic. The ΔT block itself can provide inversion.
- 16.3. In Fig. 16.3, we double ΔT . Redraw the waveforms and explain whether the loop still operates properly.
- 16.4. Repeat the previous problem if we halve ΔT .
- 16.5. An engineer notices that the DAC in an asynchronous SAR loop is twice as slow as originally thought. How can the engineer adjust ΔT in Figs. 16.2(b) and 16.3?
- 16.6. Suppose the comparator in Fig. 16.4 becomes metastable in a given SAR cycle, i.e., its outputs take a long time to reach V_{DD} and ground. Redraw the waveforms in Fig. 16.3 for this case.
- 16.7. Determine the reference voltages in the third cycle in Fig. 16.9.
- 16.8. Suppose the middle comparator in Fig. 16.9(b) suffers from an offset, V_{OS} , such that its output changes if $V_{in} > V_{REF}/2 + V_{OS}$. If $V_{OS} = 2V_{REF}/16$, explain what happens in the second cycle.
- 16.9. Repeat the previous problem if the bottom comparator exhibits an offset with the same value.
- 16.10. Repeat Example 16.3 if $V_{in} = 4.2$ LSB.
- 16.11. Repeat Example 16.3 if $V_{in} = 7.3$ LSB.
- 16.12. Suppose the comparator offset in Fig. 16.12 is $V_{OS} = -1.1$ LSB. Repeat the analysis for this case.
- 16.13. A student claims that the capacitance of DAC_S in Fig. 16.14(c) creates a gain error for DAC_F . Is that correct? Explain. (Neglect parasitic capacitances.)
- 16.14. In Fig. 16.17, $V_{in} = 65.5$ LSB. Repeat the analyses in Figs. 16.19 and 16.20.
- 16.15. In Fig. 16.19, a parasitic capacitance equal to $0.1C_u$ appears from the output to ground. Calculate V_{DAC} and explain whether the SAR loop incurs an error.
- 16.16. Repeat the analysis in Fig. 16.23 if $V_{OS} = -1.1$ LSB.
- 16.17. Repeat the analysis in Fig. 16.27 if $V_{in}^+ = 5.9$ LSB and $V_{in}^- = 10.1$ LSB.
- 16.18. Suppose the comparator in Fig. 16.27(a) suffers from an offset, V_{OS} . Redraw the waveforms in Fig. 16.27(b) if $V_{OS} = +1.1$ LSB or $V_{OS} = -1.1$ LSB.
- 16.19. Repeat the analysis in Fig. 16.31 if $V_{in} = 6.8$ LSB.
- 16.20. Repeat the analysis in Fig. 16.33 for a differential topology.

Fundamentals of Pipelined ADCs

Pipelined ADCs are another popular class of converters that have been in use for half a century. As stand-alone designs or in combination with other architectures, these ADCs offer certain benefits in terms of speed, resolution, and complexity. For example, many of today's SAR converters also incorporate pipelining.

As with SAR operation, pipelining is also based on binary search. Nevertheless, owing to their “analog-intensive” nature, pipelined ADCs require more complex analyses than do SAR topologies. In this chapter, we begin with pipelining concepts and gradually develop the basic architecture. We next introduce two topologies called “non-flip-around” and “flip-around” circuits that form the core of each pipelined stage. We then compare SAR and pipelined architectures.

17.1 Binary Search Revisited

We invoked the concept of binary search in Chapter 14 to develop the SAR architecture. Let us consider the

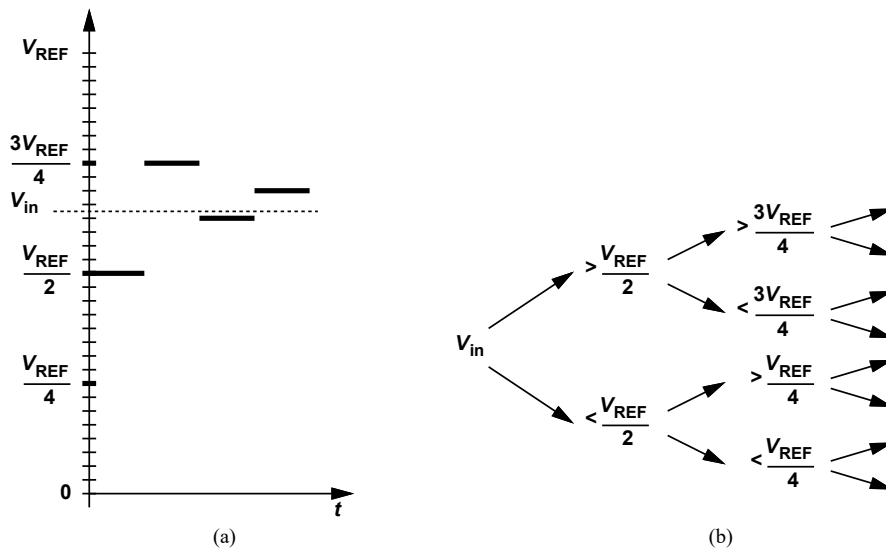


Figure 17.1 (a) Simple binary search, and (b) its decision tree.

idea from a different perspective. Suppose, as shown in Fig. 17.1(a), an ADC having an input range of zero to V_{REF} senses a voltage V_{in} . Binary search begins by nominating $V_{REF}/2$ as the best “analog estimate” of the input; a comparison then decides whether $V_{in} < V_{REF}/2$ or $V_{in} > V_{REF}/2$. We say we have resolved 1 bit thus far. If $V_{in} > V_{REF}/2$, the search identifies $3V_{REF}/4$ as a better approximation, etc. That is, the ADC

continues to compare V_{in} to the most recent analog estimate and directs the search according to the *polarity* of their difference. Figure 17.1(b) illustrates how the decision result in one cycle leads to comparison with the proper analog estimate in the next cycle.

In SAR ADCs, the analog estimates are successively generated by a single loop [Fig. 17.2(a)]. But that need not be the case in general. We can conceptually envision a converter consisting of cascaded stages [Fig. 17.2(b)], where Stage 1 computes the first analog estimate, Stage 2 the next one, etc. Our motivation in pursuing such an idea is that, perhaps, these stages can operate *faster* than a SAR loop.

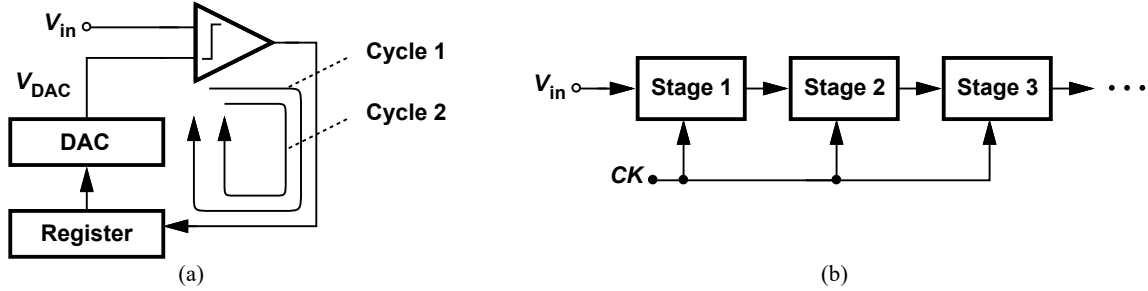


Figure 17.2 (a) SAR loop illustrating binary-search cycles, and (b) its pipelined counterpart.

In order to construct the stages in Fig. 17.2(b), we first ask, exactly what quantity is delivered by Stage 1 to Stage 2? Is it merely the analog estimate, e.g., $V_{REF}/2$? No, it cannot be! Stage 2 still needs V_{in} just as the second SAR cycle in Fig. 17.2(a) does. We must therefore ponder how the analog estimate and V_{in} should emerge from Stage 1.

If we follow the trajectories shown in Fig. 17.1(b), we can say that binary search first decides whether $V_{in} - V_{REF}/2 > 0$ or $V_{in} - V_{REF}/2 < 0$. Based on this result, it then considers $V_{in} - 3V_{REF}/4$ or $V_{in} - V_{REF}/4$ (Fig. 17.3). We surmise that Stage 1 in Fig. 17.2(b) can simply produce $V_{in} - V_{REF}/2$. Next, Stage 2 receives this quantity, detects its polarity, and generates $V_{in} - 3V_{REF}/4 = (V_{in} - V_{REF}/2) - V_{REF}/4$ or $V_{in} - V_{REF}/4 = (V_{in} - V_{REF}/2) + V_{REF}/4$. That is, Stage 2 need only add $-V_{REF}/4$ or $+V_{REF}/4$ to its input and deliver the result to Stage 3.

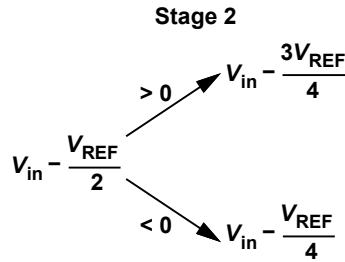


Figure 17.3 Second-stage decisions.

Example 17.1

Explain how Stage 3 in Fig. 17.2(b) operates.

Solution

If this stage receives $V_{in} - 3V_{REF}/4$ and decides that this difference is positive [Fig. 17.4(a)], then it must allow comparison of V_{in} to $7V_{REF}/8$, i.e., it must generate $V_{in} - 7V_{REF}/8 = (V_{in} - 3V_{REF}/4) - V_{REF}/8$ [Fig. 17.4(b)]. On the other hand, if $V_{in} - 3V_{REF}/4 < 0$, Stage 3 delivers $V_{in} - 5V_{REF}/8$, which amounts to $(V_{in} - 3V_{REF}/4) + V_{REF}/8$.

Similarly, if Stage 3 receives $V_{in} - V_{REF}/4$ and if this difference is positive, the output should be equal to $V_{in} - 3V_{REF}/8 = (V_{in} - V_{REF}/4) - V_{REF}/8$. For the case of $V_{in} - V_{REF}/4 < 0$, the output should emerge as $V_{in} - V_{REF}/8 = (V_{in} - V_{REF}/4) + V_{REF}/8$. In summary, Stage 3 simply adds $-V_{REF}/8$ or $+V_{REF}/8$ to its input.

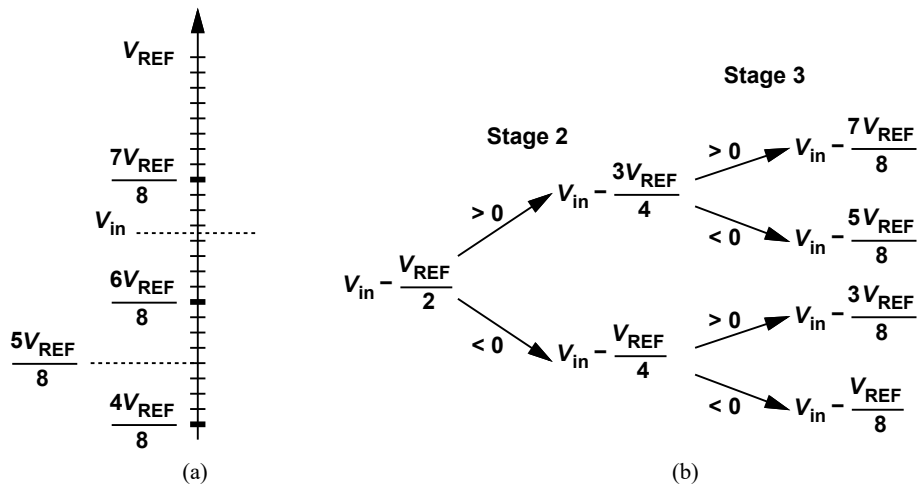


Figure 17.4 (a) Input range example, and (b) ADC decision tree.

The foregoing thoughts suggest that the stages in Fig. 17.2(b) require at least two functions, namely, comparison and addition (or subtraction). The n -th stage compares its input to zero, adds $+V_{REF}/2^n$ or $-V_{REF}/2^n$ to it, and provides the result to the next stage. But how is the digital output obtained? When Stage 2 detects the polarity of $V_{in} - V_{REF}/2$, it yields the MSB. Similarly, the decision made by Stage 3 signifies the next MSB, etc. It appears that resolving N bits requires $N + 1$ stages. We will show later that, in fact, N stages suffice.

We must make several remarks about the cascade shown in Fig. 17.2(b). First, in contrast to the SAR structure of Fig. 17.2(a), where the signal circulates around the same circuit, the cascade “unwraps” the loop and resolves each bit using a different block. We thus face a higher power consumption here, but will benefit from a higher speed. Second, as is the case in any ADC, we must include a sampling circuit—perhaps in Stage 1. Third, Fig. 17.4(b) implies that the stages in the cascade are different; i.e., each has its own design. We will revisit this point later.

17.2 Basic Pipelining

The cascaded stages shown in Fig. 17.2(b) digitize the analog input by processing it along with fractions of V_{REF} . Let us add a front-end sampler (Fig. 17.5) and ask for *how long* the sampled value must be held. We expect that the stages in the cascade take some time to resolve N bits, requiring a proportionally long hold time. Such an arrangement does not necessarily perform a faster conversion than a SAR loop does. In fact, the higher power consumption makes the cascade far less attractive.

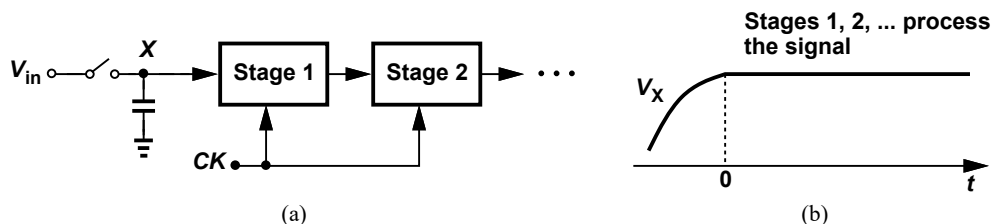


Figure 17.5 Sampling at a pipelined ADC input.

Fortunately, the performance can be dramatically improved through the use of “pipelining.” As an analogy, let us consider an assembly line in an automobile factory, where a long sequence of operations must be carried

out to build a vehicle. The line (which we call a “pipeline”) consists of a number of stations, each dedicated to a few operations. At any point in time, all of the stations “process” different vehicles “concurrently” (Fig. 17.6). That is, the rate at which a vehicle is built is limited by the time taken by only *one* station. We assume

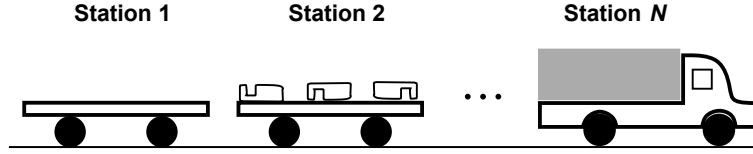


Figure 17.6 Analogy of a pipelined ADC to a vehicle assembly line.

all stations to require the same amount of time.

How can we apply this pipelining concept to the cascade shown in Fig. 17.2(b)? As evident from Fig. 17.4(b), the only information carried from one stage to the next is of the form $V_{in} \pm nV_{REF}/2^j$. We then surmise that this voltage can be *sampled* at the input of each stage so that the *preceding* stage need not hold its output any longer. This is similar to one station in Fig. 17.6 handing over the partially assembled vehicle to the next station. Let us see how this can be accomplished. As shown in Fig. 17.7(a), the front-end sampler enters the hold mode at $t = t_1$, providing a voltage equal to $V_{in}(t_1)$ to Stage 1. For example, we assume $V_{in}(t_1) > V_{REF}/2$. After this instant, Stage 1 begins to generate $V_{in}(t_1) - V_{REF}/2$ while the next sampler tracks the output of Stage 1 [Fig. 17.7(b)]. At $t = t_2$, this sampler goes into the hold mode and Stage 2 begins to process $V_{in}(t_1) - V_{REF}/2$ so as to produce either $V_{in}(t_1) - V_{REF}/4$ or $V_{in}(t_1) - 3V_{REF}/4$ [Fig. 17.7(c)]. Note that Stage 1 is now free and can begin to acquire the next value of V_{in} . Figure 17.7(d) summarizes the pipelined cascade’s operations, emphasizing that *different* input samples are processed concurrently by different stages.

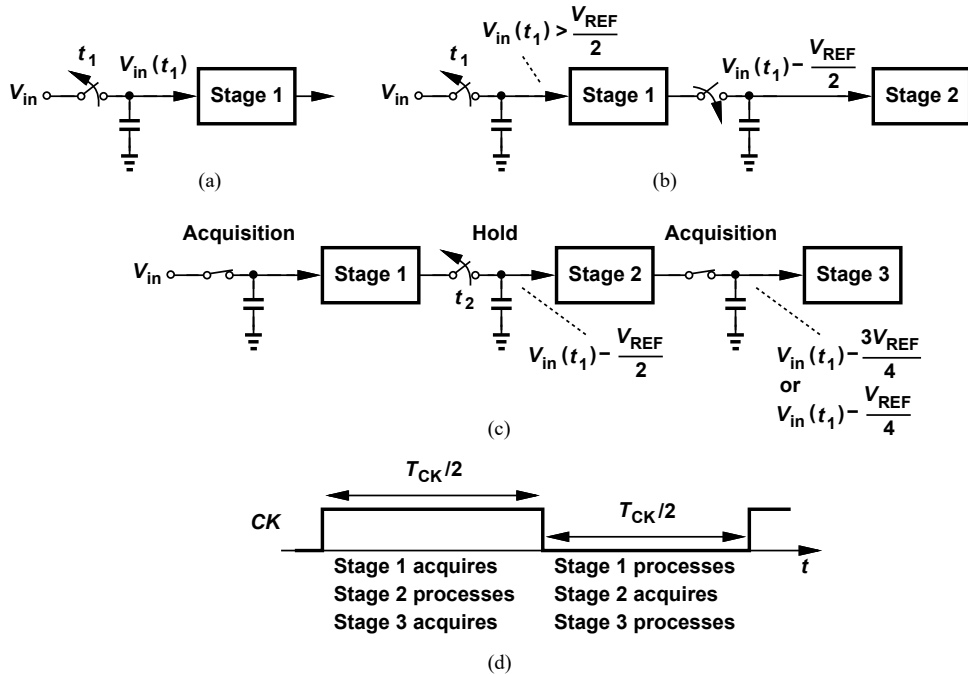


Figure 17.7 (a) First stage of a pipelined ADC, (b) voltages in two cascaded stages, (c) voltages in three cascaded stages, and (d) summary of operations.

The key point emerging from these thoughts is that the time interval $T_{CK}/2$ in Fig. 17.7(d) need only be long enough for one stage to acquire or to process—just as in the assembly line analogy. With N stages, therefore, a pipelined ADC resolves N bits at a conversion rate equal to $1/T_{CK}$, which can be considerably higher than that of SAR loops.

The abstract picture that we have thus far constructed for pipelined ADCs may raise many questions in the reader's mind. Indeed, we will need to spend several sections of this chapter to build a simple converter. The reader is urged to continue with patience.

An important observation that soon comes to our aid is that the polarity of $V_{in} - V_{REF}/2^j$ is the same as that of $2^j V_{in} - V_{REF}$. For example, Stage 1 in Fig. 17.7(b) can deliver $2V_{in} - V_{REF}$ rather than $V_{in} - V_{REF}/2$, i.e., it can *amplify* the input by a factor of 2 and subtract V_{REF} from it. But how should the second stage process $2V_{in} - V_{REF}$ so as to determine the sign of $V_{in} - 3V_{REF}/4$ or $V_{in} - V_{REF}/4$? After multiplying it by 2, this stage should subtract or add V_{REF} so as to yield, respectively, $4V_{in} - 3V_{REF} = 4(V_{in} - 3V_{REF}/4)$ or $4V_{in} - V_{REF} = 4(V_{in} - V_{REF}/4)$, which are simply amplified versions of their counterparts in Fig. 17.4(b). Figure 17.8 presents this modification.

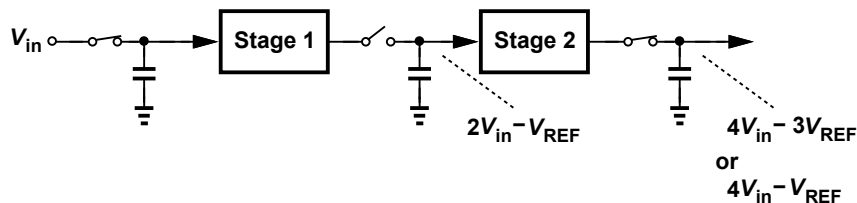


Figure 17.8 Residues in a pipelined ADC.

In summary, each stage in the pipeline performs four operations: it samples the output of the previous stage, it amplifies the result by a factor of 2, it adds or subtracts V_{REF} , and it produces a bit for the digital output.

This approach offers two benefits: (1) As the signal travels through the pipeline and is amplified, it becomes proportionally less susceptible to the noise and offset of the stages that it encounters. Thus, the stages deeper in the cascade can consume less power. (2) There is no need for providing fractions of V_{REF} to the pipeline. These points motivate us to devise a method of precise amplification by a factor of 2 (Section 17.4).

Fully-Differential Implementations Our thoughts have thus far assumed a single-ended ADC input voltage ranging from zero to V_{REF} . In a differential system, the input difference varies from $V_{REF}^- - V_{REF}^+ = -V_{REF}$ to $V_{REF}^+ - V_{REF}^- = +V_{REF}$. Binary search must therefore begin by comparing V_{in} to zero. This decision is followed by a comparison to $+V_{REF}/2$ or $-V_{REF}/2$, etc. [Fig. 17.9(a)]. Our amplifying and pipelining concept is then realized as shown in Fig. 17.9(b), where the polarity of $2V_{in} + V_{REF}$ or $2V_{in} - V_{REF}$ is detected. We return to this point in Section 17.5.

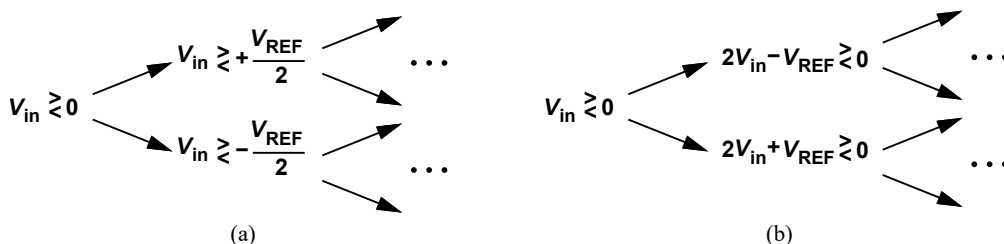


Figure 17.9 (a) Simple decision tree, and (b) its alternative form.

17.3 Brief Review of Op Amp Models

The design of pipelined ADCs relies on precision processing of analog signals and hence op amps. In this section, we review basic models and imperfections of these circuits. For op amp design principles, the reader is referred to [1].

In its simplest form, an op amp can be modeled by a voltage-dependent voltage source with an open-loop gain of A_0 and a zero output impedance (Fig. 17.10). In ADC design, A_0 falls in the range of 10^1 to 10^4 ,

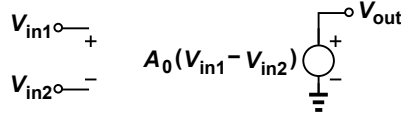


Figure 17.10 Simple op amp model.

depending on the necessary closed-loop precision. For example, consider the inverting amplifier shown in Fig. 17.11(a), where $V_{out}/V_{in} = -R_2/R_1$ if $A_0 \rightarrow \infty$. With $A_0 < \infty$, on the other hand, we turn to the equivalent circuit depicted in Fig. 17.11(b), recognizing that R_2 experiences the Miller effect. Thus, R_2 presents a resistance equal to $R_2/(1 + A_0)$ at X , yielding

$$V_X = \frac{\frac{R_2}{1 + A_0}}{\frac{R_2}{1 + A_0} + R_1} V_{in}, \quad (17.1)$$

and hence

$$V_{out} = -A_0 V_X \quad (17.2)$$

$$= \frac{-A_0 \frac{R_2}{1 + A_0}}{\frac{R_2}{1 + A_0} + R_1} V_{in}. \quad (17.3)$$

It follows that

$$\frac{V_{out}}{V_{in}} = \frac{-A_0 R_2}{R_2 + (1 + A_0) R_1} \quad (17.4)$$

$$= \frac{1}{1 + \frac{1}{A_0} (1 + \frac{R_2}{R_1})} \cdot \frac{-R_2}{R_1}. \quad (17.5)$$

The circuit therefore suffers from gain error, a serious effect in pipelined ADCs.

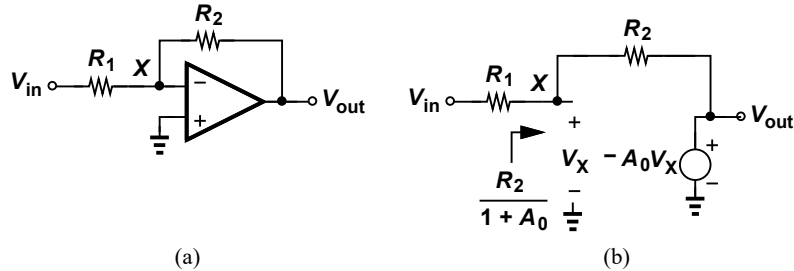


Figure 17.11 (a) Inverting amplifier, and (b) its model.

For a more realistic op amp model, we can add an output resistance, R_{out} , as shown in Fig. 17.12(a), arriving at a Thevenin equivalent. We generally assume $A_0 \gg 1$.

Since R_{out} is relatively large in typical CMOS op amps, we may opt for a Norton equivalent instead [Fig. 17.12(b)]. We call G_m the “transconductance” of the op amp. We have $V_{out} = G_m R_{out} (V_{in1} - V_{in2})$ and hence $A_0 = G_m R_{out}$. Note the direction of the current source in this model. (Choosing the opposite direction gives $A_0 = -G_m R_{out}$.)

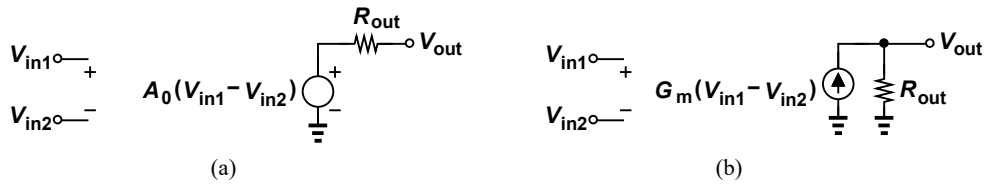


Figure 17.12 (a) Thevenin, and (b) Norton op amp models.

Example 17.2

Compute the input resistance in Fig. 17.13(a), assuming an open-loop output resistance of R_{out} for the op amp.

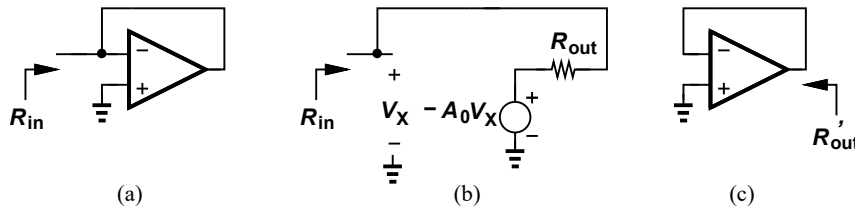


Figure 17.13 (a) Op amp in unity-gain feedback, (b) equivalent circuit, and (c) use of same model for output resistance calculation.

Solution

Replacing the op amp with its Thevenin equivalent, we arrive at Fig. 17.13(b) and recognize that R_{out} experiences the Miller effect. Thus,

$$R_{in} = \frac{R_{out}}{1 + A_0} \quad (17.6)$$

$$\approx \frac{R_{out}}{G_m R_{out}} \quad (17.7)$$

$$\approx \frac{1}{G_m}. \quad (17.8)$$

It follows that both R_{in} in Fig. 17.13(a) and R'_{out} in Fig. 17.13(c) are approximately equal to the inverse transconductance of the op amp.

The op amp output resistance proves particularly problematic if a *resistive* feedback network is used. As an example, let us return to the inverting amplifier of Fig. 17.11(a) and draw its equivalent circuit, including R_{out} (Fig. 17.14). Equation (17.6) still holds here if R_2 is replaced with $R_2 + R_{out}$. We also note that the current flowing through R_2 is equal to $(V_X + A_0 V_X)/(R_2 + R_{out})$.

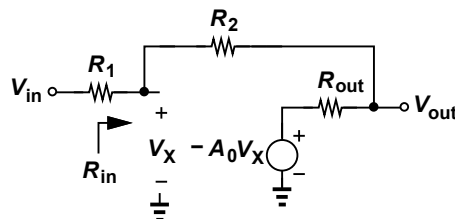


Figure 17.14 Inverting amplifier model including R_{out} .

A KVL gives

$$\frac{V_X + A_0 V_X}{R_2 + R_{out}} R_{out} - A_0 V_X = V_{out}. \quad (17.9)$$

It follows that

$$\frac{V_{out}}{V_{in}} = \frac{R_{out} - A_0 R_2}{R_2 + R_{out} + (1 + A_0) R_1} \quad (17.10)$$

$$= \frac{-1 + \frac{R_{out}}{A_0 R_2}}{1 + \frac{1}{A_0} \left(1 + \frac{R_2 + R_{out}}{R_1} \right)} \cdot \frac{-R_2}{R_1}. \quad (17.11)$$

In comparison to (17.5), (17.10) indicates that R_{out} introduces additional gain error. As explained later in this chapter, *capacitive* feedback networks avoid this issue.

We are also concerned with the finite bandwidth of op amps. We begin by simply assuming the gain in Fig. 17.10 follows a one-pole response:

$$V_{out} = \frac{A_0}{\frac{s}{\omega_0} + 1} (V_{in1} - V_{in2}), \quad (17.12)$$

where ω_0 denotes the open-loop -3 -dB bandwidth. An interesting and often undesirable effect here is the *inductive* behavior of closed-loop input and output impedances. The following example illustrates this point.

Example 17.3

Determine the input impedance of the circuit in Fig. 17.13(a) if the op amp is modeled by (17.12) and has a finite output resistance, R_{out} .

Solution

The equivalent circuit of Fig. 17.13(b) and Eq. (17.6) still apply here if A_0 is replaced with $A_0/(s/\omega_0 + 1)$. That is,

$$Z_{in} = \frac{R_{out}}{1 + \frac{A_0}{\frac{s}{\omega_0} + 1}} \quad (17.13)$$

$$= \frac{\frac{s}{\omega_0} + 1}{\frac{s}{\omega_0} + 1 + A_0} R_{out}. \quad (17.14)$$

Noting that the input impedance exhibits a zero, $\omega_z = -\omega_0$, and a pole, $\omega_p = -(1 + A_0)\omega_0$, we construct its magnitude as shown in Fig. 17.15. As expected, $|Z_{in}| \approx R_{out}/(1 + A_0)$ at low frequencies. But beyond $|\omega_z| = \omega_0$, $|Z_{in}|$ begins to rise, resembling an inductor.

In some analyses, we prefer to employ the op-amp Norton equivalent of Fig. 17.12(b), in which case G_m can follow a single-pole response [Fig. 17.16(a)]. Such a model proves useful for two-stage topologies. For example, we refer to the op amp structure in Fig. 17.16(b), where C_C serves as a Miller compensation

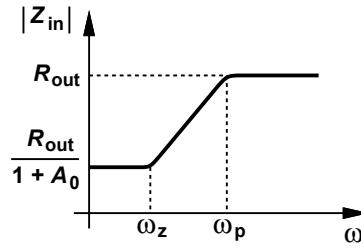


Figure 17.15 Behavior of input impedance vs. frequency.

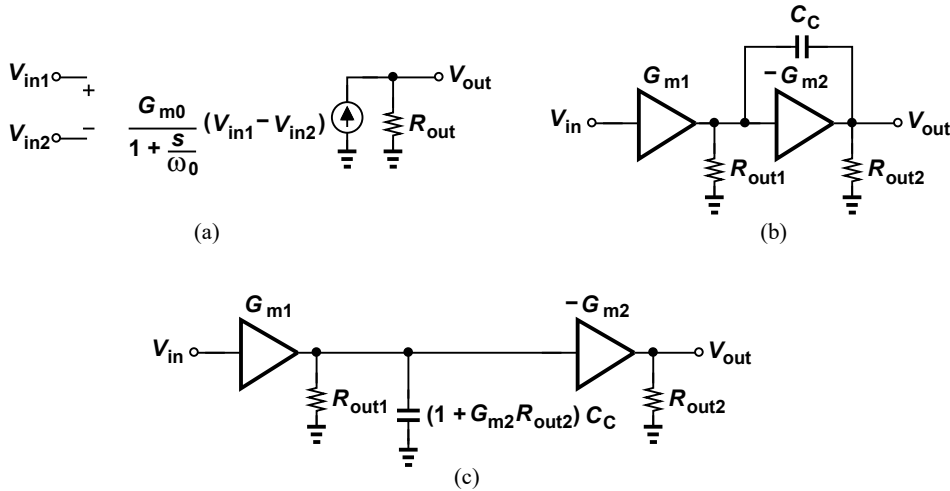


Figure 17.16 (a) One-pole op amp model, (b) two-stage op amp, and (c) its Miller approximation.

capacitor, and observe that the circuit can be approximated by that in Fig. 17.16(c). We then have

$$\frac{V_{out}}{V_{in}} = \frac{G_{m1} R_{out1} G_{m2} R_{out2}}{\frac{s}{\omega_0} + 1}, \quad (17.15)$$

where $\omega_0 = 1/[R_{out1}(1 + G_{m2} R_{out2})C_C]$. The overall op amp can be characterized by a one-pole transconductance,

$$G_m = \frac{G_{m1} R_{out1} G_{m2}}{\frac{s}{\omega_0} + 1}, \quad (17.16)$$

and an output resistance, R_{out2} . Note that G_{m0} in Fig. 17.16(a) is equal to $G_{m1} R_{out1} G_{m2}$. The similarity between the transfer functions expressed by Eq. (17.15) to Eq. (17.12) suggests that, for example, the behavior depicted in Fig. 17.15 also applies here if we simply replace R_{out} with R_{out2} and A_0 with $G_{m1} R_{out1} G_{m2} R_{out2}$. The reader is encouraged to solve the circuit without Miller's approximation and prove that $\omega_0 = 1/[(R_{out1} + R_{out1} G_{m2} R_{out2} + R_{out2})C_C]$.

17.4 Precision Multiply-by-2 Stages

We wish to construct a circuit that provides a voltage gain of 2. As explained later, the accuracy of this gain must be commensurate with the overall ADC resolution. For example, a 10-bit converter may require a gain error of less than 0.1%. This issue points to the use of feedback for precise gain definition.

17.4.1 Non-Flip-Around Amplifier

Let us begin with the simple topology shown in Fig. 17.17(a), where the closed-loop gain is approximately

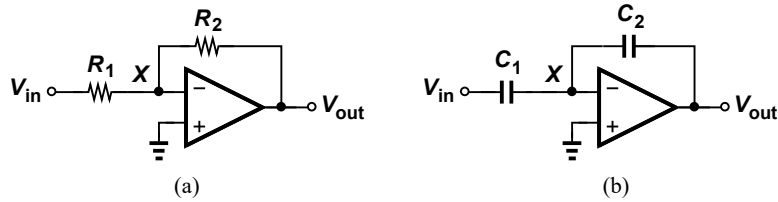


Figure 17.17 (a) Resistive, and (b) capacitive implementations of a multiply-by-2 stage.

equal to $-R_2/R_1$ if the open-loop gain of the op amp is high enough. The principal difficulty here is that the resistive feedback network loads the op amp, thus reducing its gain and hence raising the gain error. This effect proves serious because typical op amps exhibit a relatively *high* output resistance. We instead turn to the configuration depicted in Fig. 17.17(b), where the closed-loop gain is defined by capacitors C_1 and C_2 . We must study this circuit in detail as it exemplifies the topologies we frequently encounter in pipelined ADCs. In this journey, we answer questions such as (1) how does the circuit exactly operate? (2) why do the capacitors *not* reduce the gain of the op amp here? and (3) how is the initial condition on C_2 set?

Basic Operation In order to understand how the structure in Fig. 17.17(b) behaves, we first assume an ideal op amp and zero initial conditions on C_1 and C_2 and apply a step at the input (Fig. 17.18). Noting that node X is a virtual ground, we observe that C_1 receives positive charge in the amount of $V_1 C_1$ from V_{in} . What provides the negative charge on the right plate of C_1 ? This charge must come from the left plate of C_2 ,

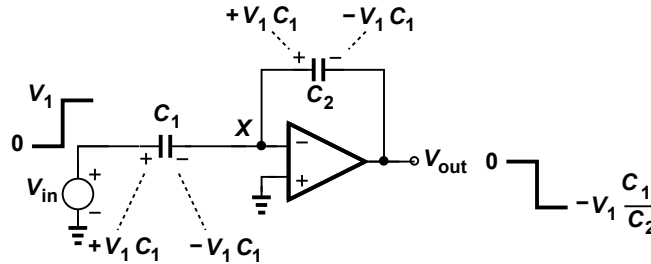


Figure 17.18 Charges in a multiply-by-2 stage.

thereby leaving positive charge of the same amount on that plate. It follows that C_2 sustains a voltage equal to $V_1 C_1 / C_2$, yielding

$$V_{out} = -\frac{C_1}{C_2} V_1. \quad (17.17)$$

We say the input voltage *change* is multiplied by $-(C_1/C_2)$. The closed-loop gain is equal to $-C_1/C_2$ and can be defined with high precision because capacitor mismatches are typically well below 1% (Chapter 11). For the pipelined stage in the previous section, we select $C_1 = 2C_2$ so as to realize a precise gain of 2.

Example 17.4

If V_{in} in Fig. 17.18 is a general function of time, determine V_{out} .

Solution

With a virtual ground at X , the current entering C_1 from V_{in} is equal to $C_1 dV_{in}/dt$. This current flows

through C_2 , generating a voltage given by

$$V_{C2} = \frac{1}{C_2} \int C_1 \frac{dV_{in}}{dt} dt \quad (17.18)$$

$$= \frac{C_1}{C_2} V_{in}, \quad (17.19)$$

where V_{C2} is defined as the voltage on the *left* plate of C_2 minus the voltage on the right plate. We have $V_{out} = -V_{C2} = -(C_1/C_2)V_{in}$.

Amplifier with Sampling Recall from Section 17.2 that pipelining requires each stage to sample and hold the output of the preceding stage. We thus ponder how the two-capacitor amplifier can be modified to this end. We begin by inserting two switches as shown in Fig. 17.19(a) so that V_{in} can be impressed across C_1 while they are on. But what exactly happens when S_1 and S_2 turn off and an input equal to V_{in0} is stored on C_1 ? Depicted in Fig. 17.19(b), the circuit now faces a “floating” input, with no charge transferred from C_1 to C_2 . To resolve this issue, we add another switch that pulls the left plate of C_1 to ground [Fig. 17.19(c)]. We can therefore say that V_A jumps from V_{in0} to zero, yielding $V_{out} = (C_1/C_2)V_{in0}$. We call the operation phases in Figs. 17.19(a) and (c) the “sampling” and “amplification” modes, respectively.

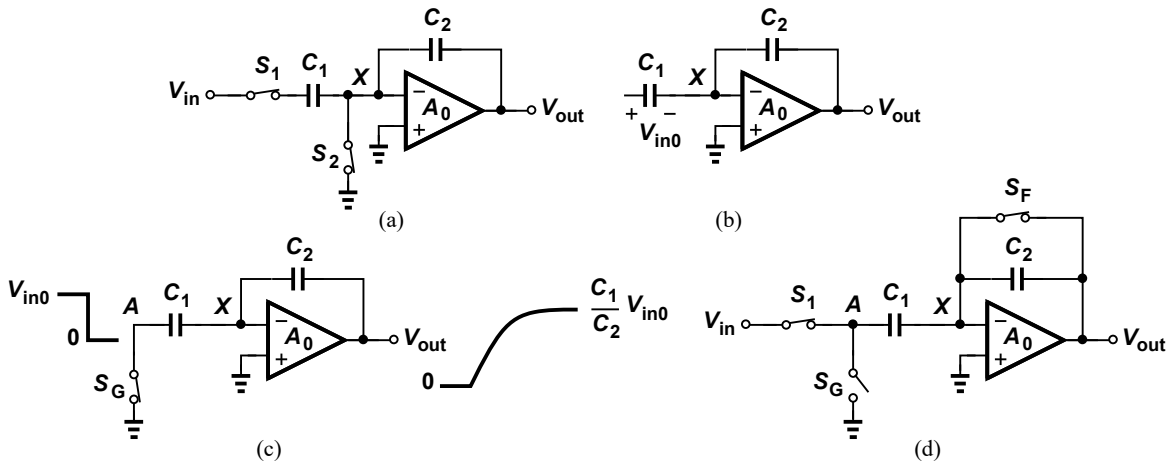


Figure 17.19 (a) Multiply-by-2 stage with switches, (b) circuit after S_1 and S_2 turn off, (c) circuit after S_G turns on, and (d) complete circuit.

The topology of Fig. 17.19(a) generates a zero output in the sampling mode if the initial condition on C_2 and the op amp offset are zero. With a finite offset, $V_{out} = -A_0V_{OS}$, introducing two issues. First, the op amp may saturate with such a large output and take longer to recover as the circuit enters the amplification mode. Second, C_2 carries a nonzero initial condition, corrupting V_{out} . To avoid these difficulties, we “reset” the op amp in the sampling mode by means of a feedback switch [Fig. 17.19(d)]; S_F is, in principle, no longer necessary. Configuring the op amp as a unity-gain buffer, S_F also removes any charge on C_2 from the previous amplification phase. The virtual ground at X still allows V_{in} to be impressed across C_1 .

It is interesting that the circuit of Fig. 17.19(c) is an *inverting* amplifier but it generates $V_{out} = (C_1/C_2)V_{in0}$ rather than $-(C_1/C_2)V_{in0}$. This is because we allow node A to jump from V_{in0} to zero.

For reasons that become clear in the next section, the topology of Fig. 17.19(d) is called a “non-flip-around amplifier.” It performs two of the three functions stipulated in Section 17.2 for a pipeline: it samples and multiplies the signal by a factor of 2. We must still determine how $2V_{in} - V_{REF}$ can be formed (Section 17.5).

Example 17.5

If V_{in} in Fig. 17.19(d) is a sinusoid, sketch the output waveform. Assume for now that S_1 and S_F turn on and off together. The clock duty cycle is 50%.

Solution

As shown in Fig. 17.20, $V_{out} = 0$ while S_1 and S_F are on. At $t = t_1$, these switches turn off and S_G turns on, causing V_A to jump from V_{in} to zero and V_{out} from zero to $(C_1/C_2)V_{in}$. This value is held until $t = t_2$, when the circuit returns to the sampling mode and V_{out} to zero. In the pipelined ADC environment of Fig. 17.8, the next stage can sense and sample V_{out} between t_1 and t_2 .

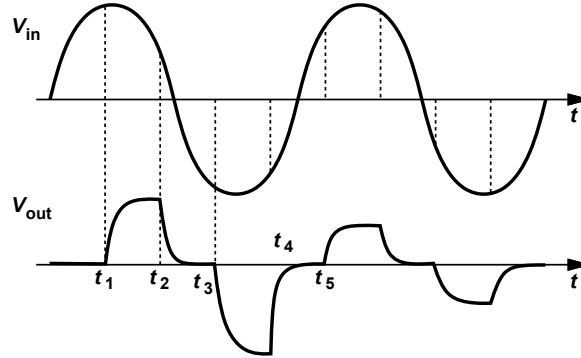


Figure 17.20 Input and output waveforms of a multiply-by-2 stage.

We have answered two of our earlier questions, namely, how the multiply-by-2 circuit in Fig. 17.18 operates and how the initial condition on C_2 is set. We now explain why, unlike resistors, capacitors do not reduce the gain of the op amp. We recognize that, after the amplifier settles, e.g., at $t = t_2$ in Fig. 17.20, C_1 and C_2 carry no current, acting as open circuits. From the equivalent circuit in Fig. 17.21, we note that the current through R_{out} falls to zero and hence V_{out} is independent of R_{out} . Thus, the op amp retains its open-loop gain regardless of its output resistance, a point of contrast to the results obtained in Section 17.3. This is a remarkable and extremely useful property of “switched-capacitor” circuits.

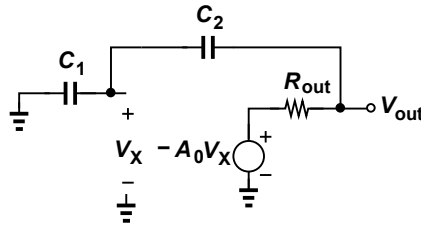


Figure 17.21 Equivalent circuit for output voltage calculation.

In practice, we prefer differential operation. Figure 17.22 shows how the non-flip-around circuit is modified for this purpose. Here, $C_1 = C_3$ and $C_2 = C_4$. In the amplification mode, we have $V_{out}^+ - V_{out}^- = -(C_1/C_2)(V_{in}^+ - V_{in}^-)$, where the plus sign accounts for the result observed in Fig. 17.19(c). For a gain of 2, we select $C_1 = 2C_2$ and $C_3 = 2C_4$.

17.4.2 Flip-Around Amplifier

Another configuration that provides both sampling and a precise gain of 2 is the “flip-around” amplifier. To develop this circuit, first consider the arrangement shown in Fig. 17.23(a), where *equal* capacitors C_1 and C_2

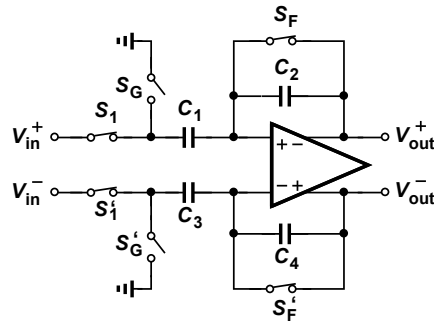


Figure 17.22 Fully-differential non-flip-around multiply-by-2 stage.

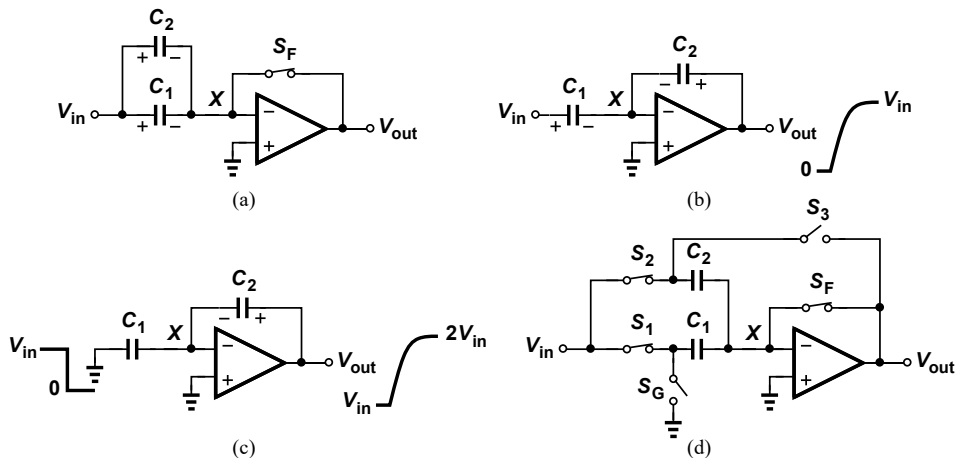


Figure 17.23 (a) Flip-around amplifier in the sampling mode, (b) output behavior after C_2 is flipped around the op amp, (c) output change after the input swings to zero, and (d) complete circuit.

sense V_{in} and S_F has reset the op amp, creating a virtual ground at X . Next, we turn S_F off and “flip” C_2 around the amplifier as illustrated in Fig. 17.23(b). As signified by the drawing, the *left* plate of C_2 in Fig. 17.23(a) switches to the op amp output in Fig. 17.23(b). Note that the total charge at node X cannot change after S_F turns off, i.e., any charge change on the right plate of C_1 must be supplied by C_2 . Assuming for now that V_{in} is constant, we observe that (1) the charge on C_1 has not changed and is still equal to $C_1 V_{in}$, (2) the charge on C_2 therefore remains constant and equal to $C_2 V_{in}$, and (3) V_{out} is equal to the voltage across C_2 , i.e., $V_{out} = V_{in}$. We can say the circuit has thus far amplified the input by a unity factor.

In the final operation phase, the left plate of C_1 fall to zero [Fig. 17.23(c)]. Acting as an inverting amplifier with a gain of $-C_1/C_2 = -1$, the circuit now inverts the input step and raises the output to $2V_{in}$. This charge change on C_2 comes from C_1 through the virtual-ground node. In contrast to this amplifier, the topology in Fig. 17.19(d) does not “flip” C_2 . Figure 17.23(d) depicts the complete structure of the flip-around amplifier.

Example 17.6

A student mistakenly selects $C_1 = 2C_2$ in Fig. 17.23(d). Explain how the circuit behaves.

Solution

The operation remains the same up to the time when the left plate of C_1 drops to zero [Fig. 17.23(c)]. This change is now amplified by a factor of $-C_1/C_2 = -2$ and added to $V_{out} = V_{in}$. The final output is then equal to $3V_{in}$.

The non-flip-around and flip-around amplifiers exhibit their own pros and cons; our study of their nonidealities in the next chapter will suggest that the latter is generally preferable. The nonidealities include the op amp's finite gain, bandwidth, output resistance, and offset voltage as well as the switch imperfections.

17.5 Multiplying DACs

The foregoing developments have partially realized a pipelined cell, providing sampling and amplification. The cascade in Fig. 17.8, however, demands a quantity of the form $2V_{in} - V_{REF}$ to be delivered from one stage to the next. In this section, we complete the functionality, arriving at a circuit block called the “multiplying DAC” (MDAC).

17.5.1 Non-Flip-Around MDAC

Let us return to Fig. 17.19(d) and note that V_A jumps from V_{in} to zero in the amplification mode, producing $V_{out} = (C_1/C_2)V_{in}$. What happens if A switches to $V_{REF}/2$ rather than to ground? As shown in Fig. 17.24(a), the change in V_A , namely, $V_{REF}/2 - V_{in}$, is amplified by a factor of $-(C_1/C_2) = -2$, yielding $V_{out} = 2V_{in} - V_{REF}$. Figure 17.24(b) depicts this simple modification. We now have a sampler/amplifier/subtractor circuit.

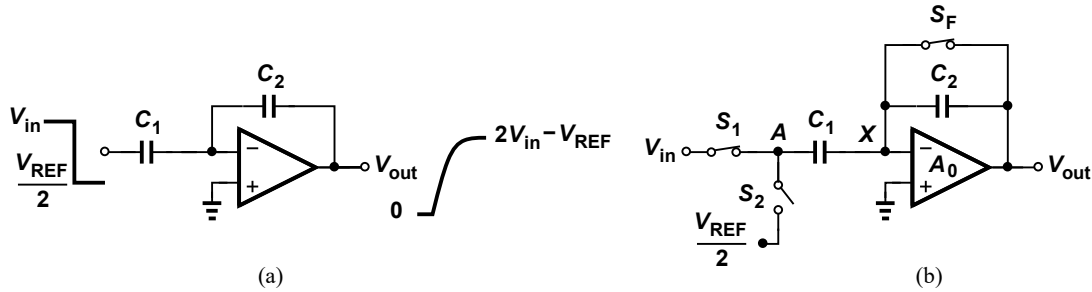


Figure 17.24 (a) Response of a non-flip-around circuit to an input step, and (b) addition of switches.

For differential operation, we must further modify the structure. In view of the procedure depicted in Fig. 17.9(b), we write

$$V_{out} = 2V_{in} + V_{REF} \quad \text{if } V_{in} < 0 \quad (17.20)$$

$$= 2V_{in} - V_{REF} \quad \text{if } V_{in} > 0, \quad (17.21)$$

where V_{out} and V_{in} denote differential quantities and $V_{REF} = V_{REF}^+ - V_{REF}^-$. These conditional operations naturally call for a comparator that determines whether $V_{in} < 0$ or $V_{in} > 0$ and directs the circuit accordingly. Shown in Fig. 17.25 is the realization. For simplicity, the topology is drawn in single-ended form even though

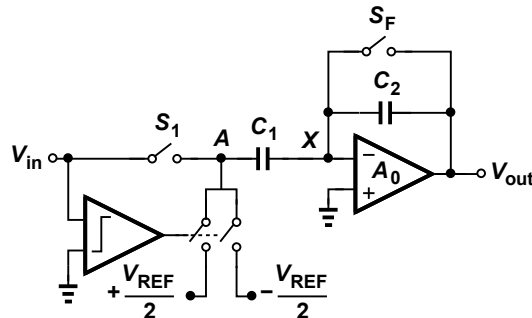


Figure 17.25 MDAC in amplification mode.

it is, in fact, differential. At the moment C_1 samples V_{in} , the comparator is clocked to detect its polarity and switches A to $+V_{REF}/2$ or $-V_{REF}/2$ in the amplification mode. We therefore have

$$V_{out} = -\frac{C_1}{C_2} \left(\pm \frac{V_{REF}}{2} - V_{in} \right) \quad (17.22)$$

$$= 2V_{in} \pm V_{REF}. \quad (17.23)$$

Note that the comparator output serves as the MSB of the ADC's digital output.

Example 17.7

Construct the complete differential counterpart of the circuit shown in Fig. 17.25.

Solution

The circuit requires V_{REF}^+ and V_{REF}^- ; we denote $V_{REF}^+ - V_{REF}^-$ by V_{REF} . Let us first generate $\pm V_{REF}/2 = \pm(V_{REF}^+ - V_{REF}^-)/2$. Illustrated in Fig. 17.26(a) is a ladder providing $V_1 - V_2 = +V_{REF}/2$. For $-V_{REF}/2$, we simply sense $V_2 - V_1$. Next, we construct the overall topology as depicted in Fig. 17.26(b), where, in the amplification mode, A switches to V_1 , and A' to V_2 if $V_{in}^+ > V_{in}^-$. We then have $V_{out} = -2[V_1 - V_{in}^+ - (V_2 - V_{in}^-)] = 2(V_{in}^+ - V_{in}^-) - 2(V_1 - V_2) = 2(V_{in}^+ - V_{in}^-) - V_{REF}$. Conversely, if $V_{in}^+ < V_{in}^-$, A and A' swing to V_2 and V_1 , respectively.

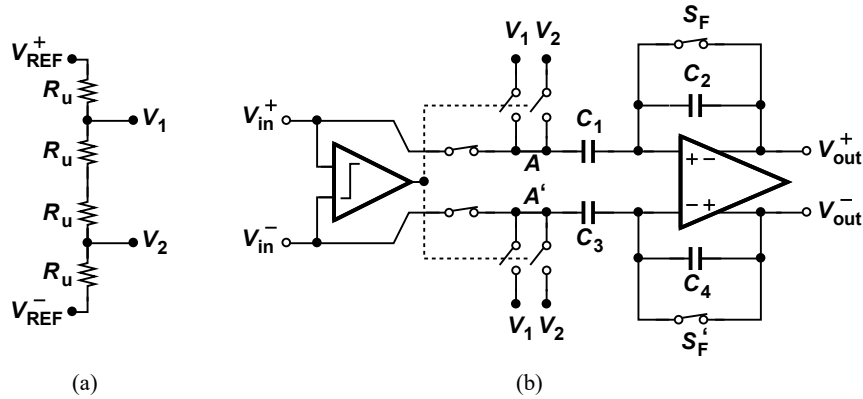


Figure 17.26 (a) Resistor ladder providing V_1 and V_2 , and (b) fully-differential MDAC.

The reader may notice that the MDAC does not need V_{REF}^+ and V_{REF}^- themselves for its operation. That is, we need only generate two voltages and simply call them $+V_{REF}/2$ and $-V_{REF}/2$. However, in other variants of pipelined ADCs, $+V_{REF}$ and $-V_{REF}$ are also necessary (Chapter 19).

It is instructive to summarize our findings. In the circuits of Figs. 17.25 and 17.26(b), the comparator acts as a 1-bit analog-to-digital converter. Next, the feedback amplifier receives the digital output of the comparator and accordingly adds or subtracts V_{REF} . We can view this latter action as 1-bit digital-to-analog conversion because an analog quantity ($+V_{REF}$ or $-V_{REF}$) is generated according to a digital signal (the comparator output). The circuit also multiplies V_{in} by a factor of 2. Thus, the overall topology is called a “multiplying DAC” even though it realizes A/D conversion and sampling as well. The MDAC plays a central role in pipelined ADC design and merits a closer look, as provided by the following examples.

Example 17.8

Suppose $V_{REF}^+ - V_{REF}^- = 32$ LSB in Fig. 17.26(b). Determine the amplifier output if $V_{in}^+ - V_{in}^- = +0.5$ LSB, $+31.5$ LSB, or -16.5 LSB (Fig. 17.27).

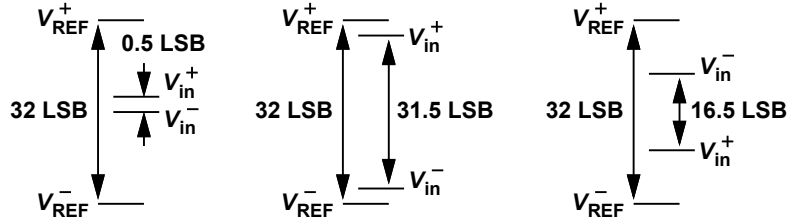


Figure 17.27 Examples of input levels.

Solution

For $V_{in}^+ - V_{in}^- = +0.5$ LSB, the comparator commands A and A' in Fig. 17.26(b) to switch to V_1 and V_2 , respectively, creating a differential voltage change equal to $(V_1 - V_2) - (V_{in}^+ - V_{in}^-)$, which amounts to $16 \text{ LSB} - 0.5 \text{ LSB} = +15.5 \text{ LSB}$ on the left plates of C_1 and C'_1 . This change is amplified by a factor of -2 , yielding $V_{out} = -31 \text{ LSB} = 2V_{in} - V_{REF}$. Similarly, for $V_{in}^+ - V_{in}^- = +31.5 \text{ LSB}$, we have $V_{out} = -2(16 \text{ LSB} - 31.5 \text{ LSB}) = +31 \text{ LSB}$.

If $V_{in}^+ - V_{in}^- = -16.5 \text{ LSB}$, the comparator produces a low logical level, swinging A to V_2 and A' to V_1 . The differential input change, $(V_2 - V_1) - (V_{in}^+ - V_{in}^-)$, is then equal to $+0.5 \text{ LSB}$ and hence $V_{out} = -1 \text{ LSB}$. This value is, of course, equal to $2V_{in} + V_{REF}$.

Example 17.9

Plot $V_{out} = V_{out}^+ - V_{out}^-$ as a function of $V_{in} = V_{in}^+ - V_{in}^-$ for the MDAC of Fig. 17.26(b).

Solution

Since the circuit is governed by Eqs. (17.20) and (17.21), we arrive at the behavior shown in Fig. 17.28, where $V_{REF} = V_{REF}^+ - V_{REF}^-$. The circuit multiplies V_{in} by 2 and, depending on the sign of V_{in} , adds or subtracts V_{REF} . We note that the output is close to the full scale if $V_{in}^+ - V_{in}^- \approx 0$, as observed in the previous example. Also, $V_{out} = 0$ for $|V_{in}^+ - V_{in}^-| = V_{REF}/2$. This input-output characteristic proves extremely useful in the analysis of pipelined ADCs.

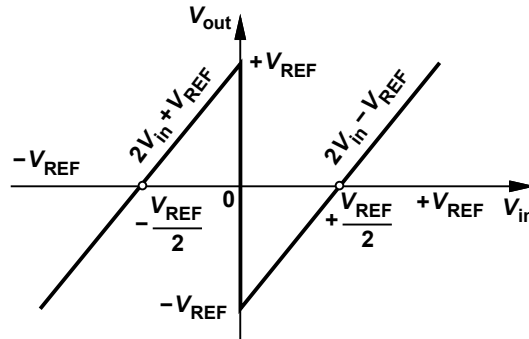


Figure 17.28 Input-output characteristic of differential MDAC.

17.5.2 Flip-Around MDAC

The concepts introduced above can also be applied to the flip-around sampler/amplifier described in Section 17.4.2. Let us begin with the basic topology shown in Fig. 17.29(a) and ask how it can perform subtraction if

$C_1 = C_2$. We seek the proper value of the reference, V_r . After sampling V_{in} on C_1 and C_2 , the circuit enters the amplification mode [Fig. 17.29(b)], generating $V_{out} = V_{in} - (C_1/C_2)(V_r - V_{in}) = 2V_{in} - V_r$. Thus, V_r is multiplied by a factor of -1 , a point of contrast to the behavior of the non-flip-around counterpart. In a single-ended design, we would select $V_r = V_{REF}$.

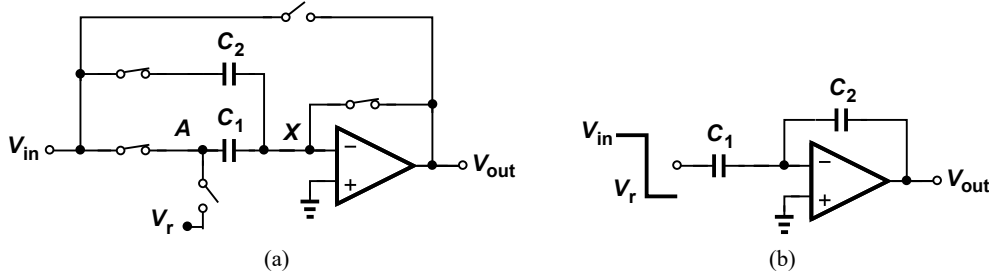


Figure 17.29 (a) Flip-around MDAC, and (b) its equivalent circuit.

Differential implementations must still follow Eqs. (17.20) and (17.21), demanding that we switch the left plates of C_1 and C_3 in Fig. 17.23(e) to V_{REF}^+ and V_{REF}^- . Figure 17.30 illustrates the resulting “flip-around MDAC.” Upon flipping around the amplifier, C_2 and C_4 deliver an output equal to $V_{in}^+ - V_{in}^-$. If

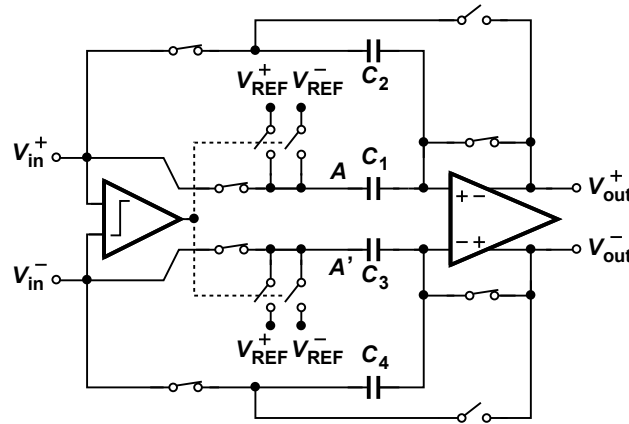


Figure 17.30 Fully-differential flip-around MDAC.

$V_{in}^+ > V_{in}^-$, the comparator swings A to V_{REF}^+ and A' to V_{REF}^- . We note that the input change, $(V_{REF}^+ - V_{in}^+) - (V_{REF}^- - V_{in}^-)$ is amplified by a factor of -1 and added to the component provided by C_2 and C_4 . It follows that $V_{out} = (V_{in}^+ - V_{in}^-) - (V_{REF}^+ - V_{in}^+) + (V_{REF}^- - V_{in}^-) = 2(V_{in}^+ - V_{in}^-) - (V_{REF}^+ - V_{REF}^-)$. Conversely, if $V_{in}^+ < V_{in}^-$, we have $V_{out} = 2(V_{in}^+ - V_{in}^-) + (V_{REF}^+ - V_{REF}^-)$.

An important advantage of the flip-around MDAC is that it does not require fractions of the reference voltage. The non-flip-around structure in Fig. 17.26, on the other hand, must deal with the settling time associated with the resistor ladder and the input capacitors.¹ Other advantages of the flip-around topology are described in Chapter 18. The two circuits studied thus far are called “1-bit-per-stage” MDACs.

An interesting difference between the two is that capacitor C_2 is reset in the former and must be charged by the op amp, whereas it is charged by V_{in} in the latter. We delve into this point in Problem 17.11.

Example 17.10

Suppose the MDACs of Figs. 17.26(b) and 17.30 are placed in two black boxes. How can we distinguish them?

¹Methods of generating $\pm V_{REF}/2$ without resistors exist, but they face other trade-offs.

Solution

Since both circuits follow Eqs. (17.20) and (17.21), their input-output characteristics are the same and similar to that in Fig. 17.28. But we observe that $C_1 = 2C_2$ in Fig. 17.26(b) whereas $C_1 = C_2$ in Fig. 17.30. We expect that this difference manifests itself in the *dynamic* behavior of the amplifiers, i.e., in their output settling times. We return to this point in the next chapter.

For the sake of simplicity, we will hereafter draw the two MDACs as shown in Fig. 17.31. For differential operation, we have $K = +1/2$ or $-1/2$ in Fig. 17.31(a) and $K = +1$ or -1 in Fig. 17.31(b). Note that $V_{in} = V_{in}^+ - V_{in}^-$ and $V_{REF} = V_{REF}^+ - V_{REF}^-$.

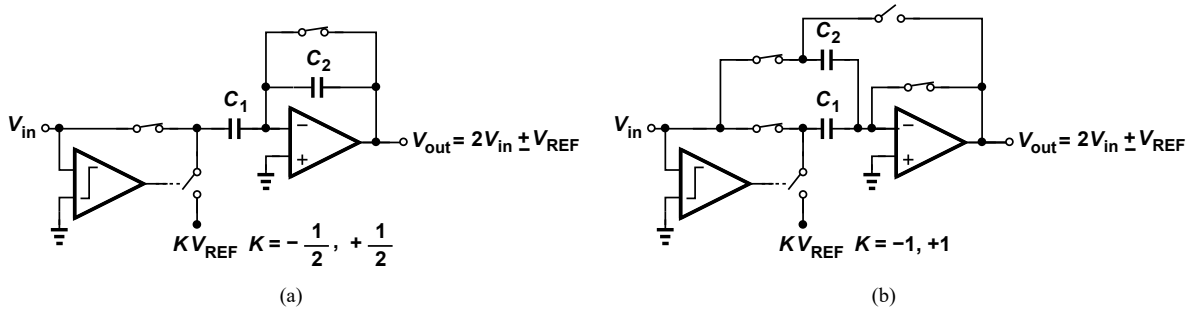


Figure 17.31 Simplified views of non-flip-around and flip-around MDACs.

It is common to view the MDAC as shown in Fig. 17.32, where a 1-bit “sub-ADC” (the comparator) and a 1-bit DAC produce an analog estimate of V_{in} , a subtractor computes the difference, and an amplifier multiplies the result by 2.

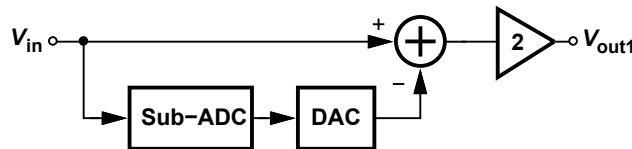


Figure 17.32 Compact view of MDAC.

17.6 Basic Pipelined ADC

Either of the MDACs developed in the previous sections can be cascaded to form a complete pipelined ADC. It is helpful to articulate the operation of each MDAC as follows: the circuit amplifies its input and adds V_{REF} to it if $V_{in} < 0$ or subtracts V_{REF} from it if $V_{in} > 0$. Let us employ the non-flip-around circuit to construct two stages. As shown in Fig. 17.33(a), while the first acquires the input, the second performs amplification. When the first begins to generate $2V_{in} \pm V_{REF}$ [Fig. 17.33(b)], the second samples this value. We study this cascade carefully, assuming $C_{11} = 2C_{12}$ and $C_{21} = 2C_{22}$.

As the cascade makes the transition from Fig. 17.33(a) to Fig. 17.33(b), a number of operations occur: the comparator in the first stage, Comp_1 , is clocked to decide whether $K = +1/2$ or $-1/2$, V_{out1} begins to travel toward $2V_{in} - V_{REF}$ or $2V_{in} + V_{REF}$, C_{21} is switched to V_{out1} , and the second op amp is reset. After V_{out1} settles and C_{21} acquires it with proper accuracy, the comparator in the second stage, Comp_2 , is clocked, and the cascade returns to Fig. 17.33(a).

We must now consider four cases. (1) The first stage generates $V_{out1} = 2V_{in} - V_{REF}$ and Comp_2 decides that this quantity is positive. Then, the second stage multiplies V_{out1} by 2 and subtracts V_{REF} from it, yielding $V_{out2} = 4V_{in} - 3V_{REF}$. When sensed by the comparator in the third stage, this quantity will reveal whether $V_{in} > 3V_{REF}/4$ or $V_{in} < 3V_{REF}/4$. (2) We have $V_{out1} = 2V_{in} - V_{REF}$ and this quantity is negative.

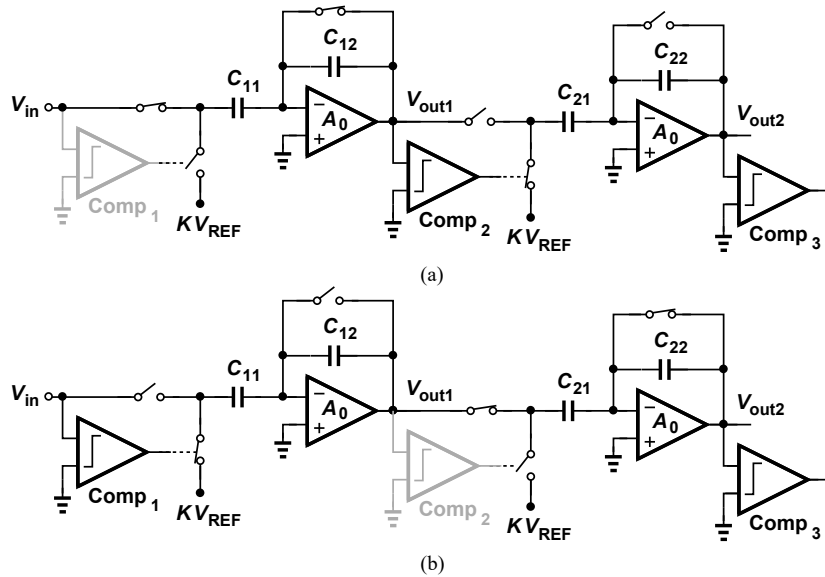


Figure 17.33 Pipelined chain using non-flip-around stages in two different clock phases.

Thus, $V_{out2} = 2V_{out1} + V_{REF} = 4V_{in} - V_{REF}$, making it possible for the third-stage comparator to decide whether $V_{in} > V_{REF}/4$ or $V_{in} < V_{REF}/4$. (3) We have $V_{out1} = 2V_{in} + V_{REF} > 0$ and hence $V_{out2} = 2V_{out1} - V_{REF} = 4V_{in} + V_{REF}$. The third stage then determines whether $V_{in} > -V_{REF}/4$ or $V_{in} < -V_{REF}/4$. (4) We have $V_{out1} = 2V_{in} + V_{REF} < 0$ and $V_{out2} = 2V_{out1} + V_{REF} = 4V_{in} + 3V_{REF}$, allowing the third-stage comparator to detect whether $V_{in} > -3V_{REF}/4$ or $V_{in} < -3V_{REF}/4$. These results are illustrated in Fig. 17.34 and expressed as

$$V_{out2} = 4V_{in} + 3V_{REF} \text{ if } -V_{REF} < V_{in} < -\frac{V_{REF}}{2} \quad (17.24)$$

$$= 4V_{in} + V_{REF} \text{ if } -\frac{V_{REF}}{2} < V_{in} < 0 \quad (17.25)$$

$$= 4V_{in} - V_{REF} \text{ if } 0 < V_{in} < +\frac{V_{REF}}{2} \quad (17.26)$$

$$= 4V_{in} - 3V_{REF} \text{ if } +\frac{V_{REF}}{2} < V_{in} < +V_{REF}. \quad (17.27)$$

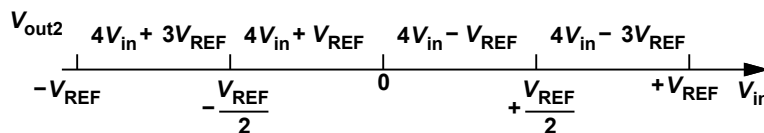


Figure 17.34 Dependence of V_{out2} upon V_{in} in different parts of the input range.

Example 17.11

Plot the outputs of Comp_1 and Comp_2 in Fig. 17.33 as a function of V_{in} for the input range from $-V_{REF}$ to $+V_{REF}$.

Solution

Comp_1 simply compares V_{in} to zero, exhibiting the behavior shown in Fig. 17.35(a). Comp_2 , on the other

hand, finds the sign of $V_{out2} = 2V_{in} - V_{REF}$ or $2V_{in} + V_{REF}$, producing a high output if $V_{in} > +V_{REF}/2$ or if $V_{in} > -V_{REF}/2$ [Fig. 17.35(b)]. We observe that Comp_1 and Comp_2 provide the first two MSBs of the digital output.

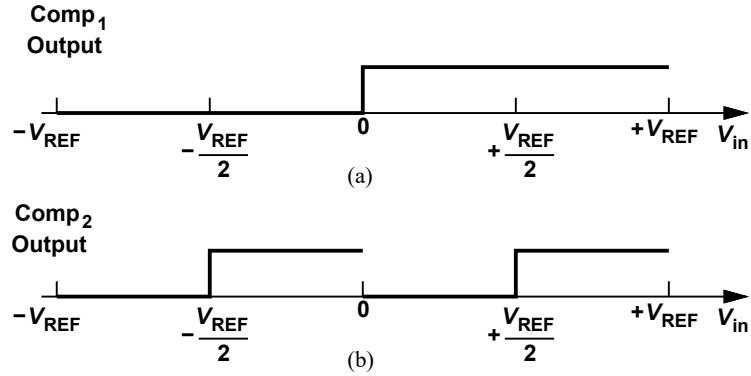


Figure 17.35 Output of (a) first comparator, and (b) second comparator vs. V_{in} .

How do we extend the cascade of Fig. 17.33 to an N -bit ADC? We recognize that $N - 1$ MDACs and one more comparator are necessary; the last stage merely determines the LSB and need not perform amplification and subtraction (Fig. 17.36). We predict that the op amps become the performance bottleneck here because of their gain, speed, voltage swing, and power trade-offs.

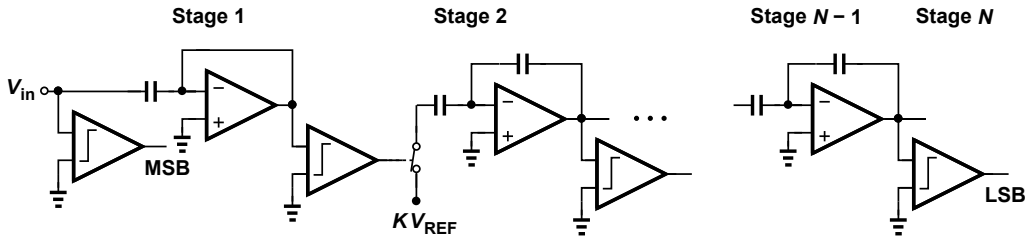


Figure 17.36 Compact representation of an N -bit ADC.

A similar approach can be taken to construct a cascade of flip-around MDACs. As shown in Fig. 17.37(a), the first stage acquires V_{in} while the second is in the amplification mode. Next, Comp_1 is clocked, C_{21} is flipped around the op amp, and C_{21} and C_{22} begin to acquire V_{out1} [Fig. 17.37(b)]. After V_{out1} settles and the acquisition is complete, Comp_2 is clocked and the circuit returns to Fig. 17.37(a) so that the second MDAC generates V_{out2} .

Example 17.12

We know that V_{out1} in Figs. 17.33 and 17.37 behaves as in Fig. 17.28. Repeat the analysis of Example 17.9 for V_{out2} .

Solution

From the four cases expressed by Eqs. (17.24)-(17.27) and from Fig. 17.34, we plot $4V_{in} + 3V_{REF}$, $4V_{in} + V_{REF}$, $4V_{in} - V_{REF}$, and $4V_{in} - 3V_{REF}$ in their corresponding intervals. Shown in Fig. 17.38, the result indicates that V_{out2} varies between $-V_{REF}$ and $+V_{REF}$ and is equal to zero at $V_{in} = -3V_{REF}/4$, $-V_{REF}/4$, $+V_{REF}/4$, and $+3V_{REF}/4$. Note that the slopes here are twice those in Fig. 17.28 owing to the amplification provided by the second stage.

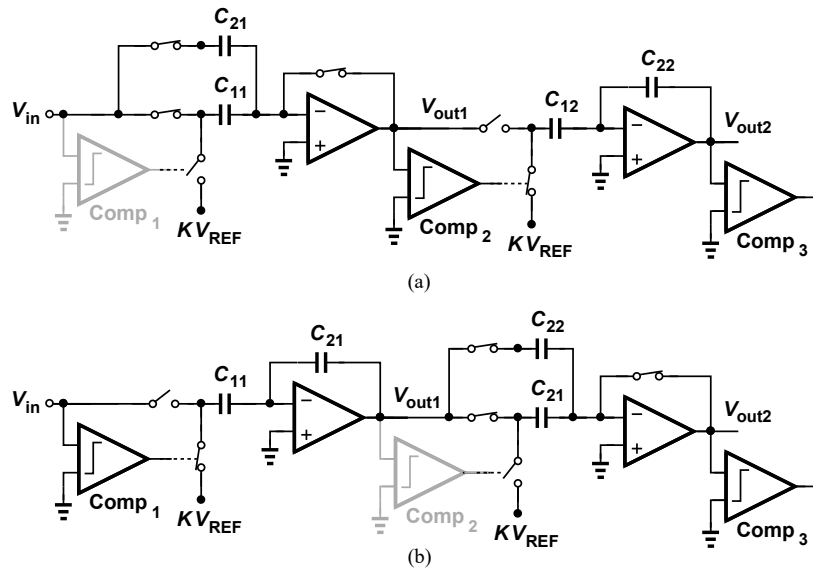


Figure 17.37 Pipelined chain using flip-around stages in two different clock phases.

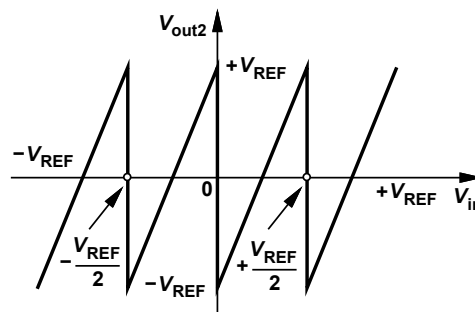


Figure 17.38 Input-output characteristic for the first two stages.

An important conclusion here is that the first stage in Figs. 17.33 or 17.37 amplifies V_{in} by a factor of 2 *without* requiring the op amp to accommodate a voltage swing equal to $2V_{in}$, which could reach $\pm 2V_{REF}$. The “look-ahead” action initiated by Comp_1 and realized by subtraction or addition of KV_{REF} guarantees that $-V_{REF} \leq V_{out1} \leq V_{REF}$. This attribute greatly eases the design of the op amp.

It is instructive to sketch the pipeline’s signals in the time domain. For example, suppose V_{in} in Fig. 17.37 is equal to $+10.7 \text{ LSB}$ and $V_{REF} = 16 \text{ LSB}$. We have $V_{out1} = 2V_{in} - V_{REF} = +5.4 \text{ LSB}$ (Fig. 17.39). The second stage receives this voltage, multiplies it by 2, and adds or subtracts V_{REF} so that the result is still bounded by $-V_{REF}$ and $+V_{REF}$. Thus, $V_{out2} = 2(+5.4 \text{ LSB}) - 16 \text{ LSB} = -5.2 \text{ LSB}$. Similarly, if we add a third stage to the pipeline, we have $V_{out3} = 2(-5.2 \text{ LSB}) + 16 \text{ LSB} = +5.6 \text{ LSB}$. Note that each output is reset for half of the clock cycle when the corresponding stage is in the sampling mode. The reader is encouraged to find the digital output resulting from these computations.

17.7 Concept of Residue

We have seen in Chapter 14 that binary search in a generic SAR ADC successively reduces the difference between the analog input, V_{in} , and the DAC output, V_{DAC} . We call $V_{in} - V_{DAC}$ the “residue.”

A similar concept applies to pipelined converters. Returning to Examples 17.9 and 17.11 and extending the idea to three stages, we plot their outputs as shown in Fig. 17.40, where $V_{out2} = 2V_{out1} \pm V_{REF}$ and

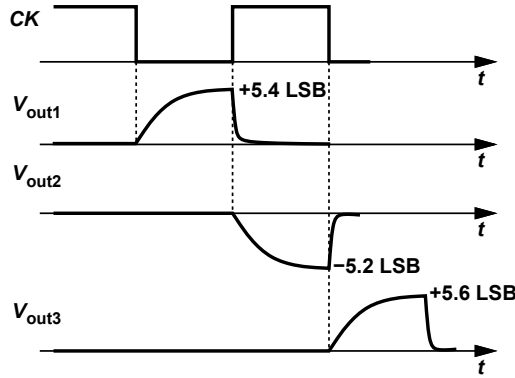


Figure 17.39 Waveforms at three outputs in a pipelined chain.

$V_{out3} = 2V_{out2} \pm V_{REF}$. We observe that the first comparator determines whether V_{out1} should be generated according to line L_1 or L_2 , the second comparator determines whether V_{out2} should be formed according to L_3 or L_4 or according to L_5 or L_6 , etc. Thus, for a given input, V_{in1} , the pipeline progressively narrows down the search “window.” We call V_{out1} the first residue, V_{out2} the second residue, etc. In this example, the third stage prescribes the line $V_{out3} = 8V_{in} - 7V_{REF}$ as the closest residue characteristic for V_{in1} .

Example 17.13

Plot the outputs of the three comparators for the residue characteristics shown in Fig. 17.40.

Solution

Comp₁ senses V_{in} and generates a high output level if $V_{in} > 0$. Comp₂ measures V_{out1} against zero and delivers a ONE as V_{in} exceeds $-V_{REF}/2$ or $+V_{REF}/2$ (Fig. 17.41). Similarly, Comp₃ detects the “zero crossings” of V_{out2} , yielding a high level as V_{in} exceeds $-3V_{REF}/4$, $-V_{REF}/4$, $+V_{REF}/4$, or $+3V_{REF}/4$.

The concept of residue plays a crucial role in the analysis of pipelined ADCs, especially when we study their imperfections. We expect that nonidealities distort the plots in Fig. 17.40. The following example gives a brief preview of our extensive analyses in the next chapter.

Example 17.14

The first-stage comparator in Fig. 17.37 suffers from an offset, $V_{OS} > 0$, generating a high logical level when $V_{in} > V_{OS}$. Plot the first residue for this case.

Solution

Equations (17.20) and (17.21) respectively emerge as

$$V_{out1} = 2V_{in} + V_{REF} \text{ if } V_{in} < V_{OS} \quad (17.28)$$

$$= 2V_{in} - V_{REF} \text{ if } V_{in} > V_{OS}. \quad (17.29)$$

Recognizing that the transition from $2V_{in} + V_{REF}$ to $2V_{in} - V_{REF}$ occurs at $V_{in} = V_{OS}$, we plot the characteristic as shown in Fig. 17.42. We also observe that two attributes of the characteristic do not change: points A and B, and the slopes of L_1 and L_2 (why?). But let us examine the behavior around points C and D. Since $V_{out1} = 2V_{in} + V_{REF}$ for $V_{in} < V_{OS}$, line L_1 continues to rise as V_{in} exceeds zero, reaching $2V_{OS} + V_{REF}$ at C. That is, the residue exhibits an “overshoot” or “overrange” of $2V_{OS}$. For $V_{in} > V_{OS}$,

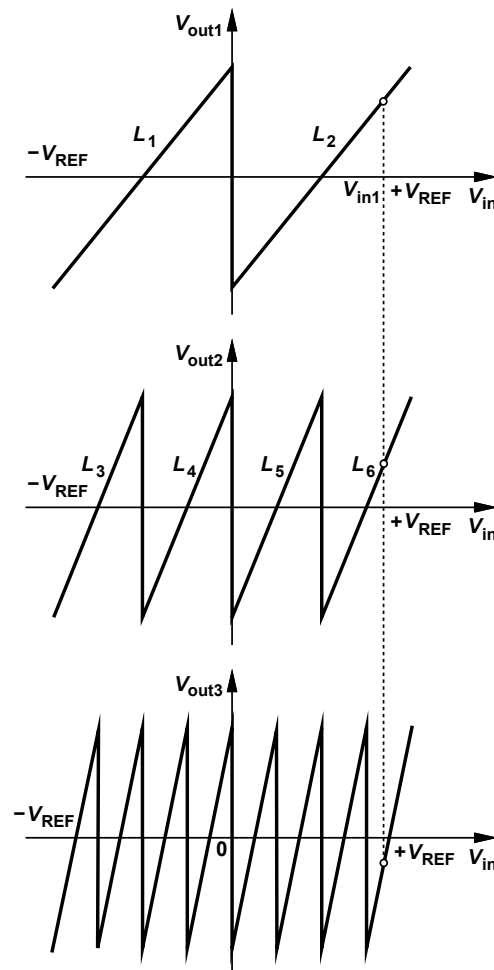


Figure 17.40 Residue plots for a three-stage pipeline.

the comparator forces $V_{out1} = 2V_{in} - V_{REF}$, yielding a value of $2V_{OS} - V_{REF}$ at D . We say the residue suffers from an “undershoot” or “underrange” of $2V_{OS}$. The reader is encouraged to repeat this analysis for $V_{OS} < 0$.

What are the implications of the nonidealities observed in Fig. 17.42? We expect that the overall ADC characteristic experiences DNL and INL. We return to this point in the next chapter.

Example 17.15

A student mistakenly selects $C_1 = 2C_2$ in the flip-around MDAC of Fig. 17.31(b). Plot the residue for this case.

Solution

Let us study the circuit in “slow motion.” Suppose the left plate of C_1 swings from V_{in} to zero [Fig. 17.43(a)] while C_2 is already flipped around the op amp and has generated $V_{out} = V_{in}$. The input change, $0 - V_{in}$, is multiplied by $-(C_1/C_2) = -2$ and added to the initial condition on C_2 , yielding $V_{out} = 3V_{in}$. Next, as shown in Fig. 17.43(b), we switch the left plate of C_1 to $\pm V_{REF}$ and obtain $V_{out} = 3V_{in} \mp (C_1/C_2)V_{REF}$,

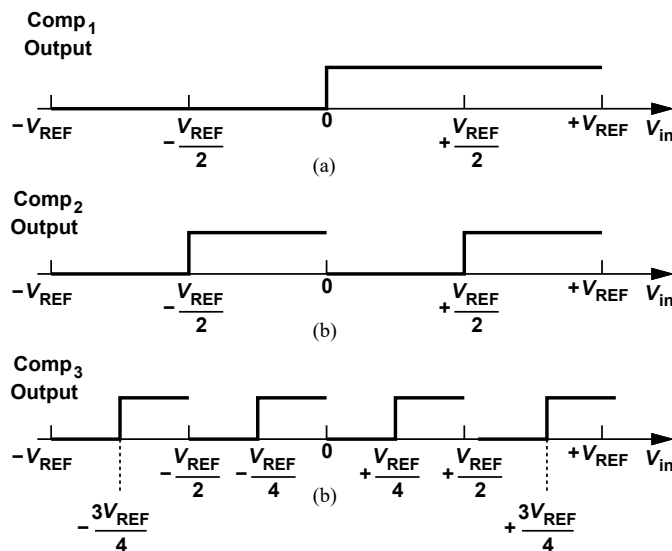


Figure 17.41 Outputs of comparators vs. V_{in} .

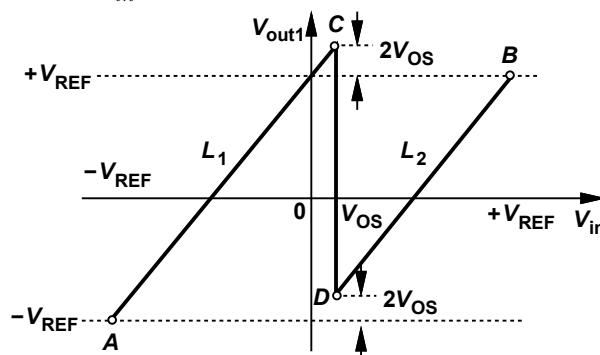


Figure 17.42 Behavior of residue in the presence of comparator offset.

which amounts to $3V_{in} \mp 2V_{REF}$. The residue is plotted in Fig. 17.43(c), exhibiting severe overshoot and undershoot around $V_{in} = 0$. In Problem 17.14, we repeat this analysis for the case of $C_2 = 2C_1$.

We now wish to examine the time necessary for a pipelined stage to complete its operations. As depicted conceptually in Fig. 17.44, the comparator is clocked, its output takes some time to rise and turn on a switch in series with the input capacitor, the residue, V_{res} , settles, and the input capacitor of the next stage acquires this value. It is only after this acquisition that the pipelined stage can begin to sample the next analog value. By comparison, a flash ADC can sense the next sample as soon as its comparators' decisions have been saved by RS latches. That is, pipelined ADCs are much slower than flash converters.

17.8 Multibit Pipelined Stages

The 1-bit-per-stage pipelines shown in Figs. 17.33 and 17.37 require $N - 1$ op amps for a resolution of N bits. Is it possible to resolve more than 1 bit in each stage so that the number of stages and hence the op-amp power consumption are reduced? Yes, it is.

It is instructive to predict what types of operations such a “multi-bit” stage requires. Recall that the MDACs studied previously (1) sample the input, V_{in} , (2) digitize it by means of a single comparator, (3) convert the

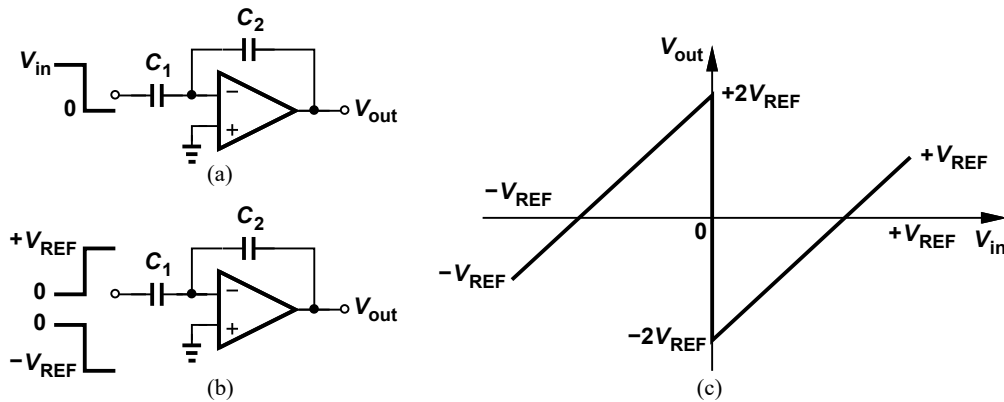


Figure 17.43 (a) Response to an input step, (b) response to the next step, and (c) resulting characteristic.

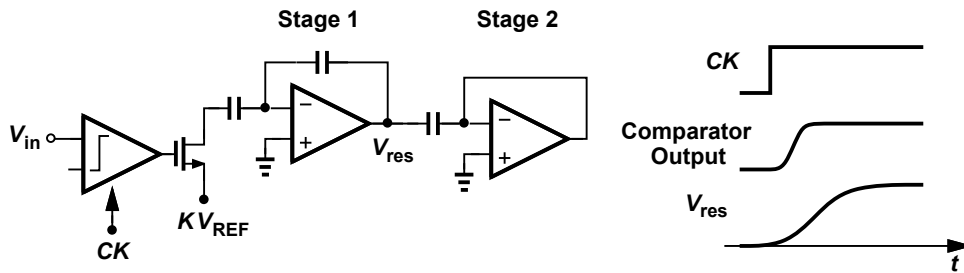


Figure 17.44 Time response of a pipelined stage.

digital output of the comparator to analog form, $\pm V_{REF}$, (4) subtract this result from V_{in} , and (5) amplify the residue. In general, therefore, a multi-bit pipelined stage emerges as shown in Fig. 17.45. The sub-ADC digitizes the input with some resolution, e.g., 4 bits, providing a “coarse” estimate. The DAC then delivers the analog version of this estimate to the subtractor, which generates the residue.

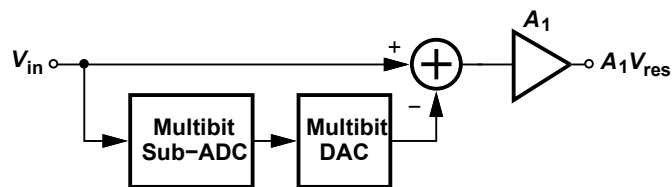


Figure 17.45 Multibit pipelined stage.

Suppose we wish to extend the MDACs of Fig. 17.31 to 2 bits of resolution. For simplicity, we begin with a single-ended design and employ three comparators sensing references equal to $V_{REF}/4$, $V_{REF}/2$, and $3V_{REF}/4$. Depicted in Fig. 17.46(a), the circuit is, of course, a simple flash ADC, giving a coarse estimate of the input. The digital output of this stage must now be converted to analog form, a task readily performed by a thermometric capacitor DAC [Fig. 17.46(b)]. Let us see how this cascade operates. If $0 < V_{in} < V_{REF}/4$, the flash thermometer code is equal to 000, the left plates of the DAC capacitors remain at zero, and $V_{DAC} = 0$. If $V_{REF}/4 < V_{in} < V_{REF}/2$, only Comp_1 generates a logical ONE, causing the left plate of its corresponding DAC capacitor to swing from zero to V_{REF} and yielding $V_{DAC} = V_{REF}/4$.

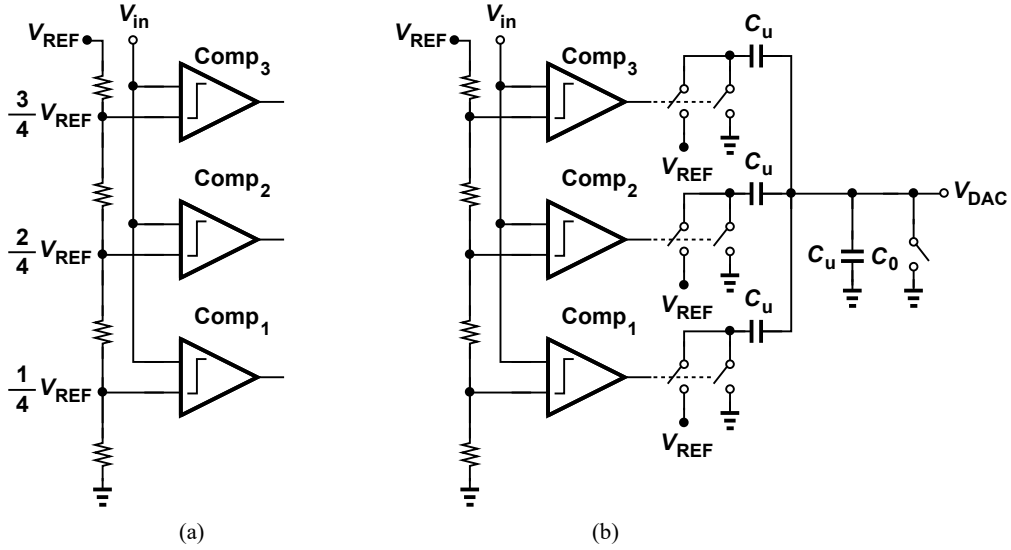


Figure 17.46 (a) Simple flash sub-ADC, and (b) multibit MDAC example.

It follows that

$$V_{DAC} = 0 \text{ if } 0 < V_{in} < \frac{V_{REF}}{4} \quad (17.30)$$

$$= \frac{V_{REF}}{4} \text{ if } \frac{V_{REF}}{4} < V_{in} < \frac{V_{REF}}{2} \quad (17.31)$$

$$= \frac{V_{REF}}{2} \text{ if } \frac{V_{REF}}{2} < V_{in} < \frac{3V_{REF}}{4} \quad (17.32)$$

$$= \frac{3V_{REF}}{4} \text{ if } \frac{3V_{REF}}{4} < V_{in} < V_{REF}. \quad (17.33)$$

We write $V_{DAC} = KV_{REF}/4$, where $K = 0, 1, 2, 3$.

We must now subtract V_{DAC} from V_{in} so as to produce the residue, V_{res} . We expect the residue to be of the form $V_{in} - KV_{REF}/4$, where $K = 0, 1, 2, 3$. We make two changes in the circuit [Fig. 17.47(a)]. First, we allow the left plates of all *four* unit capacitors to sense V_{in} in the sampling mode. Second, we switch the left plates of three units to zero or V_{REF} and that of the fourth unit only to zero. The circuit operates as follows. At the end of the sampling mode, the left plate of C_0 switches to zero. The other three left plates swing to zero or V_{REF} if their corresponding comparator produces a ZERO or a ONE, respectively. If $0 < V_{in} < V_{REF}/4$, then the flash stage delivers a thermometer code equal to 000, all left plates switch from V_{in} to zero, and $V_{res} = -V_{in}$ [Fig. 17.47(b)]. If $V_{REF}/4 < V_{in} < V_{REF}/2$, one comparator provides a high level, and hence the left plate of one unit capacitor swings to V_{REF} . As illustrated in Fig. 17.47(c), V_{res} falls to $-V_{in} + V_{REF}/4$ (why?). Except for a negative sign, this quantity is the residue of interest. Generally, if $KV_{REF}/4 < V_{in} < (K+1)V_{REF}/4$, and $K = 0, 1, 2, 3$, then $V_{res} = -V_{in} + KV_{REF}/4$.

Returning to the general realization of Fig. 17.45, we note that we have completed the structure except for the last amplification stage. For this purpose, we apply the residue obtained above to a virtual ground, as shown in Fig. 17.48. Here, $C_F = C_u$, and a voltage change of ΔV at the left plate of one DAC capacitor yields an output change equal to $-(C_u/C_F)\Delta V$ (why?). Thus, if K units switch from V_{in} to V_{REF} and the remaining from V_{in} to zero, we have $V_{out} = -K(V_{REF} - V_{in}) - (4 - K)(0 - V_{in}) = 4V_{in} - KV_{REF}$. The residue is therefore amplified by a factor of 4. This voltage gain is not unique and can be chosen less or greater.

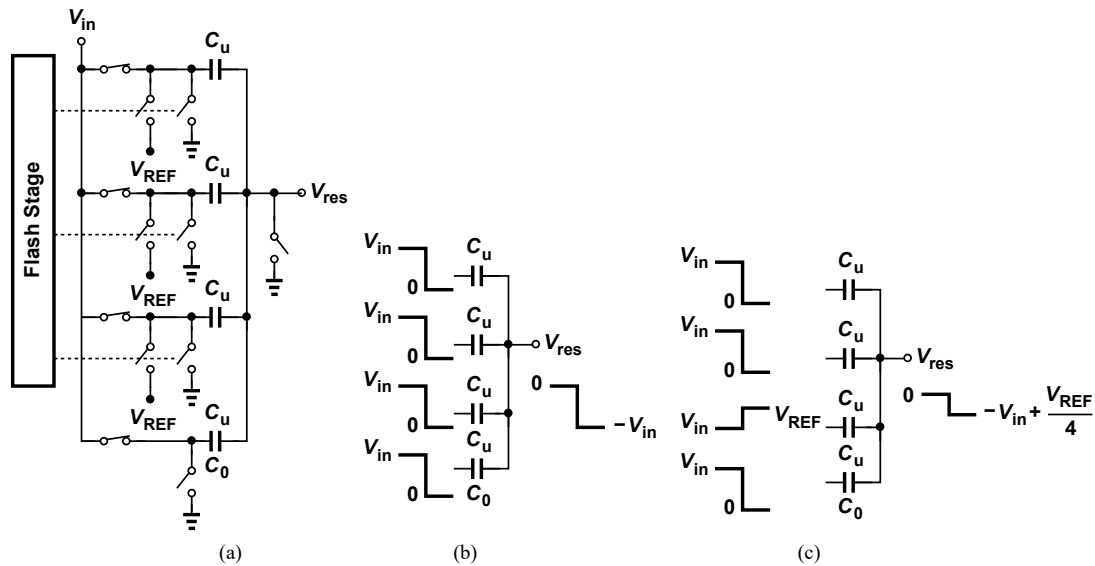


Figure 17.47 (a) Modified MDAC, (b) response to the initial change, and (c) response to the next change.

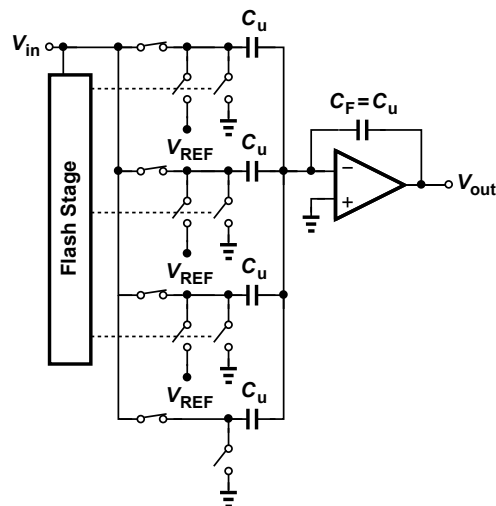


Figure 17.48 Multibit MDAC with residue amplification.

Example 17.16

If the analog input in Fig. 17.48 has a range of zero to V_{REF} , what is the range for V_{out} ?

Solution

If $0 < V_{in} < V_{REF}/4$, then $K = 0$ and $V_{out} = 4V_{in}$, ranging from zero to V_{REF} . Similarly, if $V_{REF}/4 < V_{in} < V_{REF}/2$, $V_{out} = 4V_{in} - V_{REF}/4$, etc. Figure 17.49 plots V_{out} as a function of V_{in} , revealing a range of zero to V_{REF} for the amplified residue. This point indicates that no fraction of the reference is necessary for the *next* pipelined stage if the gain is chosen equal to 4 in this case. On the other hand, if we select a lower gain, the op amp settles faster and operates with smaller output swings. The numerous trade-offs in pipelined ADCs are described in Chapter 18.

SAR	Pipelined
Conversion time = $N(t_{\text{comp}} + t_{\text{logic}}) + (N - 1)t_{\text{DAC}}$	Conversion time = $t_{\text{flash}} + t_{\text{MDAC}}$
No linear amplification $\Rightarrow$ Comparator noise an issue	Op amp gain, nonlinearity, swing, speed problematic
Needs a high-resolution, high-linearity DAC	Needs a high-linearity DAC
DAC kT/C noise an issue	MDAC kT/C noise an issue
Comparator noise an issue	Op amp noise an issue
V_{REF} settling "magnified" by a factor of $(N - 1)$	V_{REF} settling less critical

Table 17.1 Comparison of SAR and pipelined ADCs.

Another difference between the two converters relates to the settling of the reference, V_{REF} . In a SAR loop, V_{REF} must settle to the full precision in every SAR cycle. Thus, this settling time is multiplied by $N - 1$ for one conversion period. In a pipelined stage, the reference need only settle once. We conclude that the V_{REF} settling budget is much tighter in the former.

References

- [1]. B. Razavi, *Design of Analog CMOS Integrated Circuits*, Second Ed., McGraw-Hill, New York, 2017.

Problems

- 17.1. The two-stage op amp represented in Fig. 17.16(c) is configured as a unity-gain amplifier. Determine its input impedance and sketch the magnitude of this impedance as a function of frequency.
- 17.2. Suppose, in Fig. 17.2(b), $V_{in} = V_{REF}/8 + \epsilon$, where V_{REF} denotes the full scale and ϵ is a small positive voltage. Determine the output voltages of the three stages.
- 17.3. We assume that all of the capacitors in Fig. 17.7(b) begin with a zero initial voltage. If $V_{in} = 3V_{REF}/8 + \epsilon$, where V_{REF} denotes the full scale and ϵ is a small positive voltage, sketch the outputs of the three stages as a function of time.
- 17.4. In the multiply-by-2 stage of Fig. 17.19(c), we have mistakenly selected $C_2 = 2C_1$. Sketch V_{out} as a function of time.
- 17.5. Repeat the previous problem for the multiply-by-2 stage of Fig. 17.23(d).
- 17.6. In Fig. 17.24(b), $V_{in} = V_{REF}/2 + \Delta V$. Sketch V_{out} as a function of time if (a) $C_1 = 2C_2$, or (b) $C_1 = C_2$.
- 17.7. In Fig. 17.26(b), $V_{REF}^+ - V_{REF}^- = 32$ LSB. If $V_{in}^+ - V_{in}^- = -2.1$ LSB, explain how the comparator controls the four switches tied to V_1 and V_2 , and determine $V_{out}^+ - V_{out}^-$.
- 17.8. Suppose the capacitors C_1 and C_3 in Fig. 17.26(b) are 10% greater than their nominal value while C_2 and C_4 are accurate. Explain what happens to the plot shown in Fig. 17.28.
- 17.9. Repeat the previous problem for the flip-around MDAC of Fig. 17.30.
- 17.10. Compare the input capacitances of the two MDACs shown in Fig. 17.31 in the sampling mode. Explain how the switch resistances and the op-amp transconductances must be chosen so that the two circuits provide the same sampling bandwidth. (Recall that the op amp establishes an input impedance equal to the inverse of its transconductance at the virtual ground.)
- 17.11. Consider the MDACs shown in Fig. 17.31, noting that C_2 is reset in the non-flip-around topology but charged to V_{in} in the flip-around counterpart. A student claims that the latter settles faster because it does not require the op amp to charge C_2 . If V_{in} is near zero or near V_{REF} , sketch the output voltages of the two MDACs versus time and explain whether there is a speed advantage for one of the two.
- 17.12. In Fig. 17.33(a), $V_{in} = V_{REF}/8 + \Delta V$, where ΔV is a small voltage. Sketch V_{out1} and V_{out2} versus time and determine their final values.
- 17.13. Capacitor C_{11} in Fig. 17.33(a) is 10% greater than its nominal value while the other capacitors are accurate. Repeat the previous problem for this case.
- 17.14. In the flip-around MDAC of Fig. 17.31(b), C_2 is mistakenly chosen equal to $2C_1$. Construct the residue plot for this case.
- 17.15. In Fig. 17.46(b), plot V_{DAC} as a function of V_{in} as V_{in} varies from zero to V_{REF} .
- 17.16. Repeat the previous problem if C_0 is accidentally omitted.
- 17.17. Assuming that the flash stage in Fig. 17.47(a) is realized as in Fig. 17.46(a), determine V_{res} if $V_{in} = V_{REF}/4 + \Delta V$, where ΔV is a small voltage. Explain whether the left plates of the four unit capacitors are switched.
- 17.18. Repeat the previous problem if C_0 is accidentally omitted.
- 17.19. In Fig. 17.48, we select $C_F = 2C_u$. Plot the residue characteristic. Does this modification make the op-amp settling slower or faster?
- 17.20. Suppose each DAC unit capacitor in Fig. 17.48 exhibits a parasitic capacitance of C_p from its right plate to ground. If the op amp has an open-loop gain of A_0 , determine the residue for V_{in} near $V_{REF}/2$.
- 17.21. Consider the cascade shown in Fig. 17.50, assuming that each flash stage resolves 2 bits and the feedback capacitors around the op amps are equal to the DAC unit capacitors. (a) Draw the first pipelined stage in detail and plot its residue characteristic. Determine the residue gain and its maximum value. (b) Draw the second pipelined stage in detail and determine the factor J . Note that V_{REF}/J must be equal to the maximum residue value. (c) Plot the second-stage residue as a function of the first-stage residue and as a function of V_{in} .
- 17.22. Repeat the previous problem if the feedback capacitor around the first op amp is equal to twice the DAC unit capacitance in the first stage.

Nonidealities in Pipelined ADCs

Having developed the basic pipelined stage in the previous chapter, we are now ready to study its nonidealities and hence performance bounds. Due to the use of op amps and extensive analog signal processing, these ADCs exhibit numerous symptoms, requiring detailed analyses.

In this chapter, we assume a pipelined ADC consisting of cascaded 1-bit MDACs. The reader is encouraged to review the 1-bit MDACs and the concept of residue described in Chapter 17. We begin with a brief look at MDAC imperfections and then delve into the effect of comparator and op-amp offsets. Next, we study errors due to switch charge injection and op-amp nonidealities such as bandwidth and gain limitations and noise. Finally, we consider capacitor mismatch as well as reference bounce and the problem of timing mismatch. The detailed study in this chapter proves essential to the design of most multi-step ADCs.

18.1 General Considerations

18.1.1 MDAC Nonidealities

Consider the non-flip-around 1-bit MDAC shown in Fig. 18.1.¹ We identify the following imperfections in this circuit: the offset and delay of Comp_1 , the charge injection, clock feedthrough, and kT/C noise associated with switches and capacitors, the mismatch between C_1 and C_2 , the finite offset, gain, and speed of the op amp, the noise of the op amp, and the settling and noise of V_{REF} .

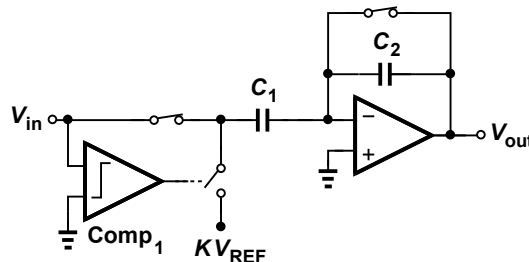


Figure 18.1 Non-flip-around MDAC.

We expect that some of these effects limit the static precision of the MDAC and quantify their impact by studying the residue. Other effects degrade the speed and must be examined in the time domain.

18.1.2 Residue Manipulations

In this chapter, we rely on residue plots extensively so as to translate each static imperfection to DNL, INL, or gain error. We should therefore be able to manipulate residues efficiently. The reader is urged to study this

¹The term “MDAC” sometimes refers to the multiply-by-2 circuit alone and sometimes to that circuit plus the comparator.

section patiently.

Suppose we are given an ideal 3-bit differential quantizer, which can be realized as a flash stage, a pipelined cascade, or any other ADC topology. The circuit provides the input-output characteristic shown in Fig. 18.2(a), where $-V_{REF} = V_{REF}^- - V_{REF}^+$ and $+V_{REF} = V_{REF}^+ - V_{REF}^-$. For example, $V_{REF}^- = 0.3$ V and $V_{REF}^+ = 0.7$ V, yielding an analog LSB size of $800 \text{ mV}/8 = 100 \text{ mV}$. We note that $D_{out,q}$ begins at 000 for $-4 \text{ LSB} < V_a < -3 \text{ LSB}$ and climbs to 111 as V_a exceeds $+3 \text{ LSB}$.

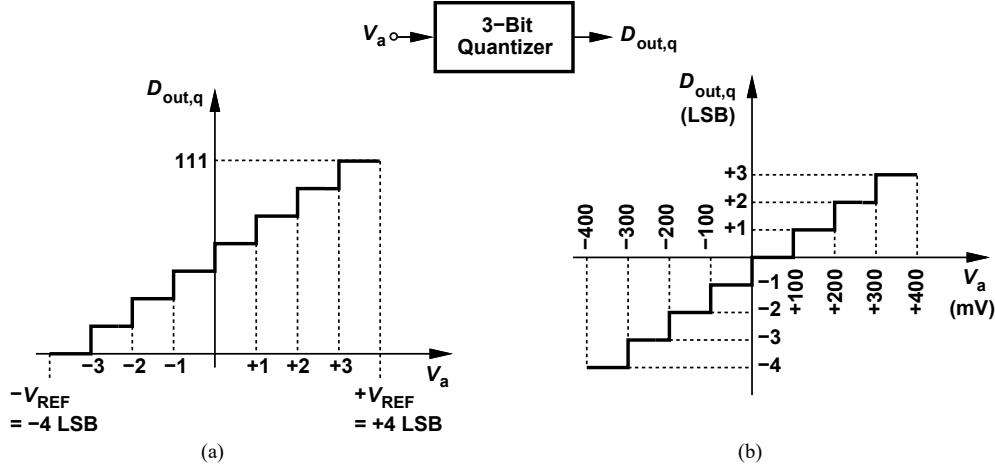


Figure 18.2 (a) A 3-bit quantizer and its characteristic, and (b) alternative view for ease of analysis.

Let us now imagine that the quantizer's digital output, $D_{out,q}$, can assume both negative and positive values. To represent negative quantities properly, we would need to resort to "2's complements," but we follow a conceptual line of thought and simply shift the characteristic down [Fig. 18.2(b)]. We represent the values of $D_{out,q}$ by -4 LSB , $\dots$, $+3 \text{ LSB}$ rather than by a binary word. The quantizer therefore maps $V_a = -399 \text{ mV}$ to $D_{out,q} = -4 \text{ LSB}$, $V_a = -99 \text{ mV}$ to $D_{out,q} = -1 \text{ LSB}$, etc.

In the next step, we precede the 3-bit quantizer by a 1-bit MDAC having a gain of 2 [Fig. 18.3(a)]. Here, V_a denotes the first residue and will be changed to V_{res1} in the next section. Our objective is to construct the input-output characteristic of this MDAC/quantizer cascade. We expect the MDAC to raise the overall resolution by 1 bit, yielding 16 output levels [Fig. 18.3(b)]. But we first plot the characteristics from V_{in} to V_a and from V_a to $D_{out,q}$. Shown in Fig. 18.3(c), the former is simply the MDAC's residue, given by

$$V_a = 2V_{in} + V_{REF} \text{ if } -400 \text{ mV} < V_{in} < 0, \quad (18.1)$$

$$V_a = 2V_{in} - V_{REF} \text{ if } 0 < V_{in} < +400 \text{ mV}, \quad (18.2)$$

and mapping $V_{in} = -400 \text{ mV}$ to $V_a = -400 \text{ mV}$, $V_{in} = -200 \text{ mV}$ to $V_a = 0$, etc. A key point here is that, as V_{in} varies from -400 mV to $+400 \text{ mV}$, V_a traverses the $[-400 \text{ mV} \text{ } +400 \text{ mV}]$ range *twice*. The characteristic from V_a to $D_{out,q}$ is already depicted in Fig. 18.2(b).

How do we plot $D_{out,q}$ as a function of V_{in} ? To avoid confusion in mapping V_{in} to $D_{out,q}$, we plot V_{in} versus V_{in} , V_a versus V_{in} , and $D_{out,q}$ versus V_a , as shown in Fig. 18.3(d). We then select a value for V_{in} , e.g., -399 mV (point A), noting that it yields $V_a = -398 \text{ mV}$ (point B) (why?), and hence $D_{out,q} = -4 \text{ LSB}$ (point C). For $V_{in} = -1 \text{ mV}$, we have $V_a = 2V_{in} + V_{REF} = +398 \text{ mV}$ and $D_{out,q} = +3 \text{ LSB}$. This characteristic repeats for $0 < V_{in} < +400 \text{ mV}$. In other words, $D_{out,q}$ resembles a "digitized" version of the residue.

Note that the MDAC gain of 2 amplifies the analog LSB by the same factor. That is, the LSB size sensed by the 3-bit quantizer's input is twice that at the MDAC input.

In the last step of our residue manipulations, we ask, how do we go from the nonmonotonic $D_{out,q}$ plot to the overall cascade characteristic? Recall that the MDAC's comparator generates a digital output, which serves as the MSB in this case. We must then "add" this output to $D_{out,q}$. From Chapter 17, we know that

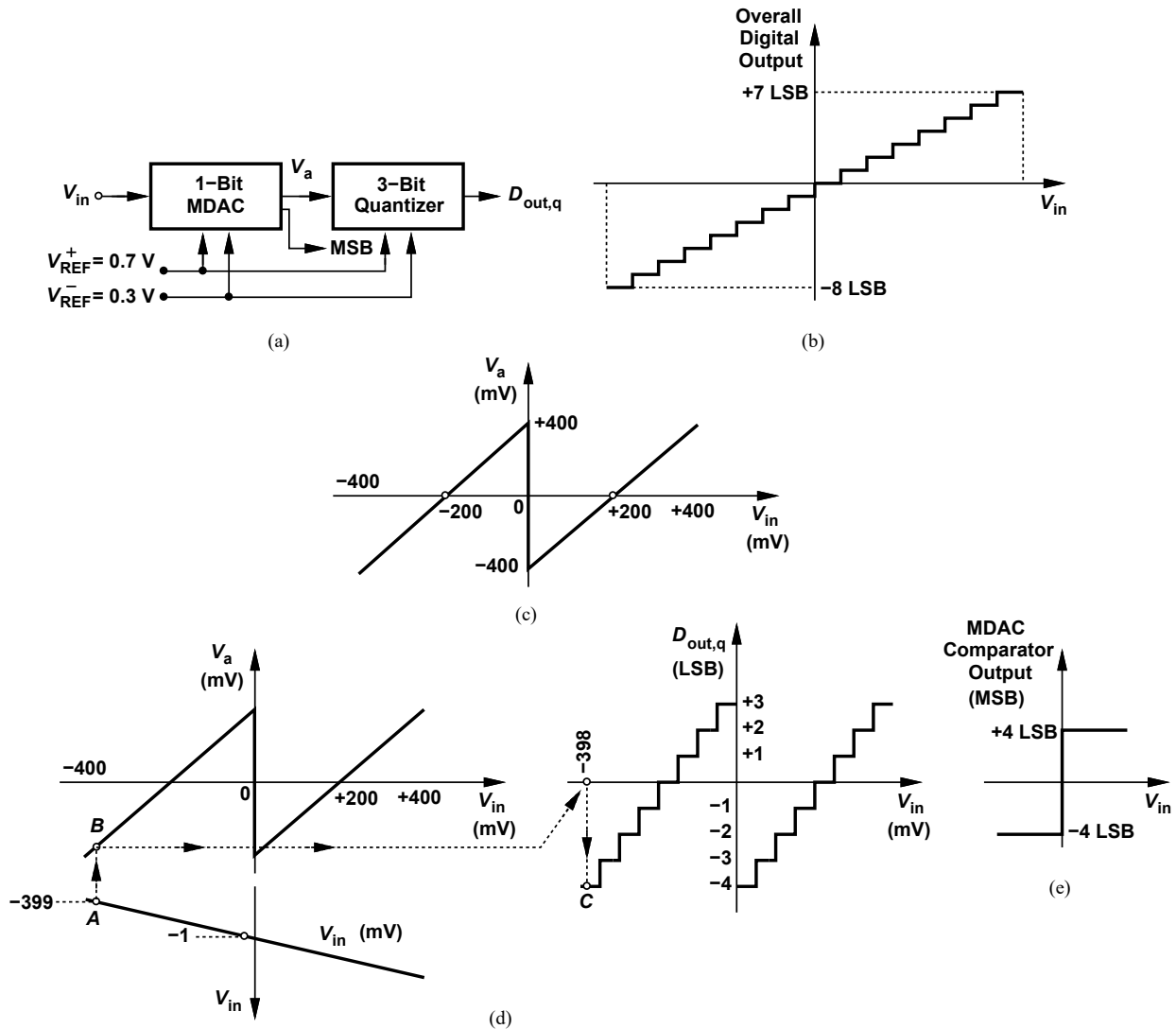


Figure 18.3 (a) Cascade of quantizers, (b) its characteristic, (c) residue plot, (d) mapping V_{in} to $D_{out,q}$, and (e) comparator output vs. V_{in} .

the MSB is low for $V_{in} < 0$ and high for $V_{in} > 0$. Acting as the fourth most significant bit,² the MSB has a weighting factor of 2^3 . In the spirit of our conceptual approach in Fig. 18.2(b), we also shift the MSB plot down by 4 LSB [Fig. 18.3(e)]. Adding this result to that in Fig. 18.3(d) then yields the characteristic shown in Fig. 18.3(b), where the vertical axis represents $MSB + D_{out,q}$. The 3-bit quantizer in our example corresponds to the stages that follow this MDAC. As another example, if a 10-bit pipelined ADC employs a 1-bit MDAC followed by other stages, we “digitize” the output of this MDAC with 9-bit resolution.

Summarized in Fig. 18.4 are our residue manipulations: we digitize the residue plot according to the total resolution of the stages that follow the first stage and shift the left section down and the right section up, arriving at the overall characteristic. Thus obtained, the result can reveal offset, gain error, DNL, and INL.

²By the fourth MSB, we mean, for example, D_1 in $D_4D_3D_2D_1D_0$.

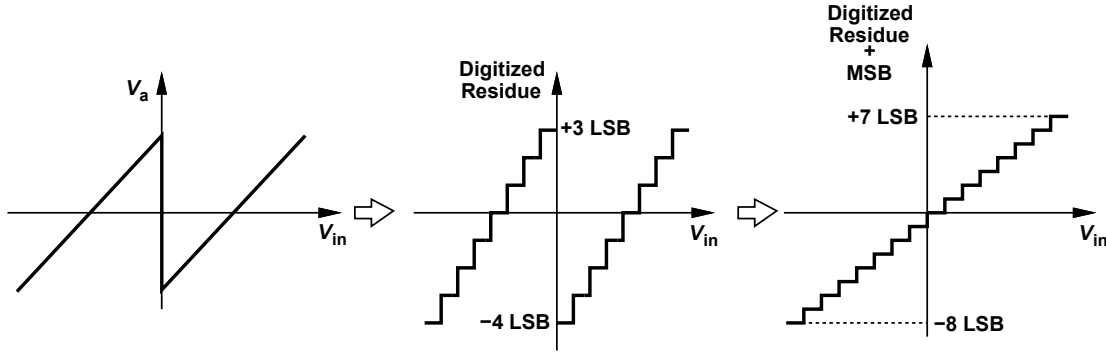


Figure 18.4 Summary of operations for finding the input-output characteristic.

18.2 Effect of Comparator Offset

18.2.1 Offset in the First Stage

Recall from Example 17.14 that the comparator offset, V_{OS} , in a 1-bit MDAC introduces overrange and underrange in the residue. Let us determine the resulting DNL and INL if such an MDAC serves as the first stage of a 4-bit ADC. We select $V_{REF}^- = 0.3$ V, $V_{REF}^+ = 0.7$ V, (1 LSB = 50 mV) and $V_{OS} = 90$ mV. We also conceptualize the overall ADC as shown in Fig. 18.5, where the last three stages are modeled by an ideal 3-bit quantizer, i.e., a 3-bit ADC.

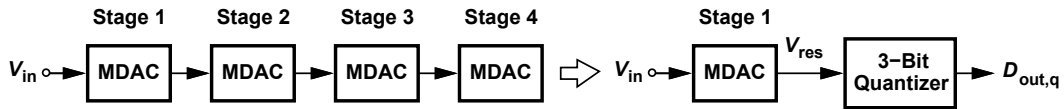


Figure 18.5 Conceptual reduction of pipelined stages.

Figure 18.6(a) plots the residue derived in Example 17.14. We assume this first stage is followed by an ideal 3-bit quantizer having a full-scale of $-V_{REF} = -400$ mV to $+V_{REF} = +400$ mV. What happens when the quantizer senses $V_{res} > +V_{REF}$? Viewing the quantizer as a flash stage for simplicity, we note that its output, $D_{out,q}$, saturates. Thus, the left section of the residue plot is digitized as shown in Fig. 18.6(b), yielding $D_{out,q} = +3$ LSB for -50 mV $< V_{in} < V_{OS}$. The right section, on the other hand, begins at $V_{res} = 2V_{in} - V_{REF} = 2V_{OS} - V_{REF} = -220$ mV. From Fig. 18.2(b), we note that this value maps to $D_{out,q} = -3$ LSB. It follows that $D_{out,q}$ begins from -3 LSB and reaches $+3$ LSB on this side of the plane.

The overall characteristic is formed by adding the MSB plot of Fig. 18.3(e) to this result, but we recognize that this plot is shifted horizontally by V_{OS} [Fig. 18.6(c)]. We thus arrive at the behavior depicted in Fig. 18.6(d). We observe three types of errors: (1) the code transition nominally occurring at $V_{in} = 0$ is shifted to $V_{in} = V_{OS} = 90$ mV, yielding DNL = 90 mV; (2) the digital output cannot take on a value of 0 LSB, suffering from a missing code because DNL > 1 LSB; and (3) viewing the INL as the integral of the DNL (Chapter 12), we obtain INL = 90 mV (why?). The circuit exhibits no gain error because the straight line passing through the characteristic's end points exhibits the nominal slope. The reader can show that a missing code is avoided if $|V_{OS}| < 1$ LSB.

The foregoing calculations illustrate the mathematical labor necessary for the analysis of pipelined ADC nonidealities. They also reveal that the MDAC comparator offset must remain well below 1 LSB. In Chapter 19, we introduce a simple modification that greatly relaxes this constraint.

18.2.2 Offset in Second Stage

We wish to quantify the effect of the comparator offset in the second stage of a pipelined ADC. We predict that the first residue amplification allows this offset to be twice as large as that in the first stage. We assume

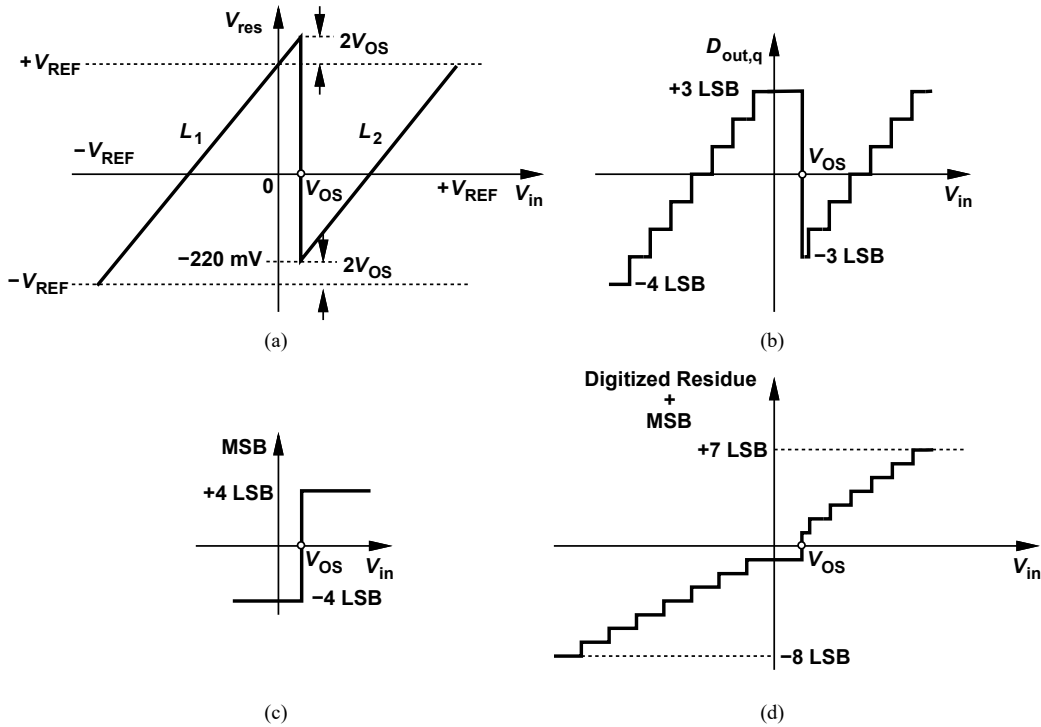


Figure 18.6 (a) Residue with comparator offset, (b) corresponding quantized behavior, (c) comparator output with offset, and (d) overall characteristic.

the same reference values as in the previous section. Our treatment applies to both flip-around and non-flip-around MDACs.

Consider the cascade shown in Fig. 18.7(a), where two 1-bit MDACs are followed by a 2-bit quantizer so as to yield an overall resolution of 4 bits. Note that $K = \pm 1/2$ for non-flip-around MDACs and $K = \pm 1$ for flip-around topologies. Recall from Chapter 17 that the residues generated by the two stages ideally behave as shown in Fig. 18.7(b), both varying from $-V_{REF}$ to $+V_{REF}$. If the first MDAC is ideal, we express lines L_1 and L_2 as

$$V_{res1} = 2V_{in} \pm V_{REF}, \quad (18.3)$$

where the choice of $+V_{REF}$ or $-V_{REF}$, i.e., the transition from L_1 to L_2 , is decided by Comp_1 . For V_{res2} , lines L_3 - L_6 are characterized by

$$V_{res2} = 2V_{res1} \pm V_{REF} \quad (18.4)$$

$$= (4V_{in} \pm 2V_{REF}) \pm V_{REF}. \quad (18.5)$$

The critical decisions made by Comp_2 occur where $V_{res1} \approx 0$, i.e., in the vicinity of $V_{in} = -200\text{ mV}$ and $V_{in} = +200\text{ mV}$. Thus, the V_{res2} transitions at $V_{in} = \pm 200\text{ mV}$ are susceptible to the offset of Comp_2 , but not the one at $V_{in} = 0$.

Ideal Case It is instructive to first construct the overall ADC characteristic when there is no offset. The 2-bit quantizer in Fig. 18.7 digitizes V_{res2} and exhibits the behavior shown in Fig. 18.8(a) as it senses an input ranging from -400 mV to $+400\text{ mV}$. We focus on line L_3 , but our results apply to all four. For $-400\text{ mV} < V_{res2} < -200\text{ mV}$, we have $D_{out,q} = -2\text{ LSB}$. This range corresponds to $-400\text{ mV} < V_{in} < -350\text{ mV}$ in Fig. 18.7(b) because we have $V_{res2} = 4V_{in} + 3V_{REF}$ and

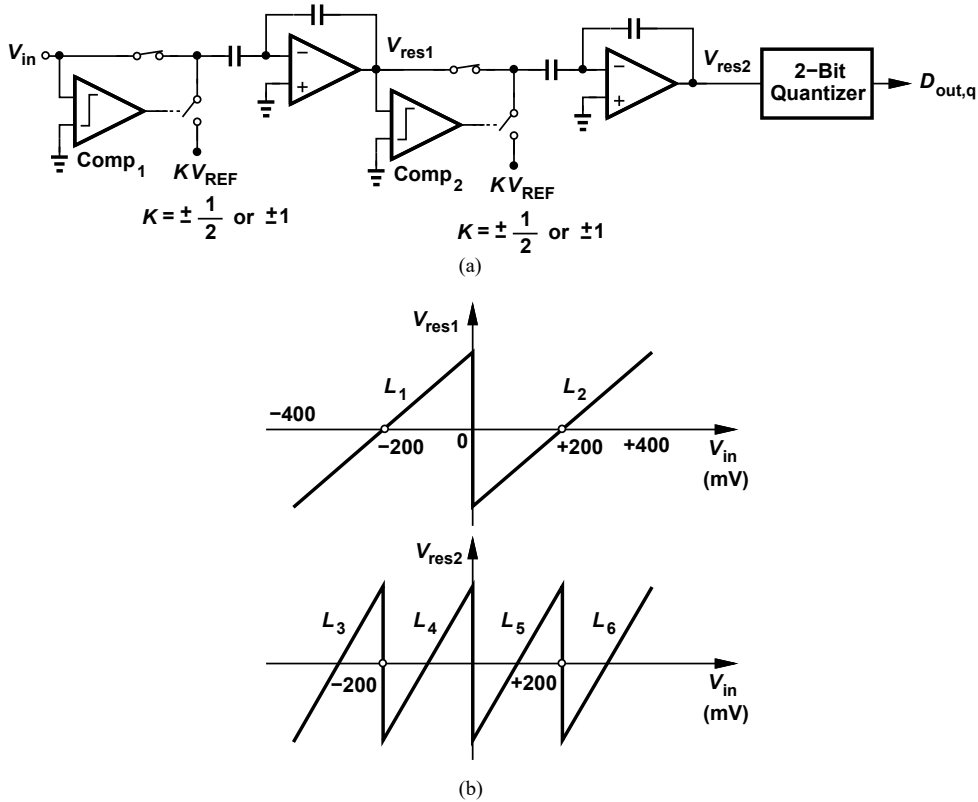


Figure 18.7 (a) Pipelined cascade, and (b) first and second residue plots.

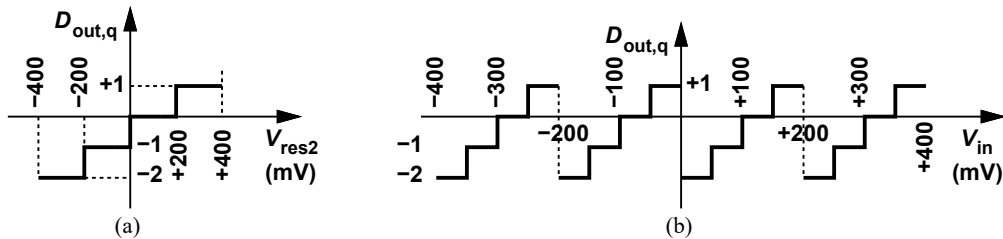


Figure 18.8 Quantizer output versus (a) the second residue, and (b) V_{in} .

hence $V_{in} = (V_{res2} - 3V_{REF})/4$. Similarly, for $-200 \text{ mV} < V_{res2} < 0$, we can equivalently write $-350 \text{ mV} < V_{in} < -300 \text{ mV}$, obtaining $D_{out,q} = -1 \text{ LSB}$. Subjecting (patiently) the entire V_{res2} plot in Fig. 18.7(b) to the quantizer's characteristic, we obtain the result depicted in Fig. 18.8(b).

In the next step, we must combine this digital output with those of Comp_1 and Comp_2 in Fig. 18.7(a). Drawing all three characteristics as in Fig. 18.9, we recognize that the output of Comp_1 serves as the MSB of the 4-bit ADC output. We thus apply a “weighing factor” to it so that it assumes values of -4 LSB or $+4 \text{ LSB}$ (why?). By the same token, the weighted output of Comp_2 swings between -2 LSB and $+2 \text{ LSB}$ so as to represent the next MSB. Summing the three plots yields the overall $D_{out}-V_{in}$ characteristic shown in Fig. 18.9.

Actual Case Let us assume an offset of $V_{OS} = +90 \text{ mV}$, i.e., Comp_2 in Fig. 18.7(a) changes its output only if V_{res1} exceeds this value. We expect that V_{res2} no longer makes a transition at $V_{in} = -200 \text{ mV}$. We proceed as follows: (a) assume $V_{res1} = 90 \text{ mV}$ and find the corresponding V_{in} , and (b) using the equa-

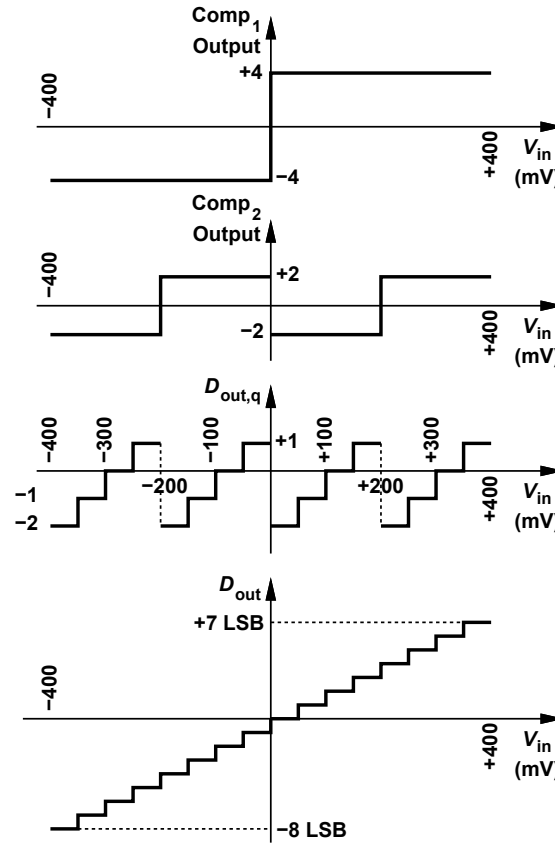


Figure 18.9 Comparator outputs, quantized residue, and overall characteristic.

tions for V_{res2} , determine at what value of V_{in} it makes a transition. From line L_1 in Fig. 18.10, we have $2V_{in} + V_{REF} = +90$ mV and hence $V_{in} = -155$ mV. We then note that line L_3 continues up to this input level before Comp_2 changes its output (Fig. 18.11). Thus, the transition is shifted from -200 mV to

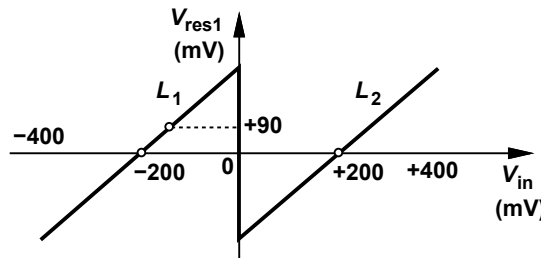


Figure 18.10 Residue plot example.

-155 mV, i.e., by $V_{OS}/2$. Expressing L_3 as $4V_{in} + 3V_{REF}$, we evaluate it at $V_{in} = -155$ mV, obtaining $V_{res2} = 580$ mV. That is, V_{res2} exhibits an overshoot of 180 mV. Similarly, line L_4 is written as $4V_{in} + V_{REF}$, taking on a value of -220 mV at the transition. The reader can readily show that the transition from L_5 to L_6 occurs at $2V_{in} - V_{REF} = 90$ mV and hence $V_{in} = +245$ mV and conclude that lines L_5 and L_6 suffer from the same errors. We recognize that the horizontal shift of both transitions is equal to $V_{OS}/2$, a result of the residue amplification by the first MDAC.

Let us now digitize V_{res2} by the (ideal) 2-bit quantizer in Fig. 18.7(a), noting its characteristic in Fig. 18.8(a). For -400 mV $< V_{res2} < -200$ mV (i.e., for -400 mV $< V_{in} < -350$ mV), $D_{out,q} = -2$ LSB and for -200 mV $< V_{res2} < 0$, $D_{out,q} = -1$ LSB (Fig. 18.12). Continuing the plot, we observe that, as V_{res2} exceeds $+200$ mV (i.e., as V_{in} passes -250 mV), $D_{out,q}$ crosses $+1$ LSB and remains constant up to

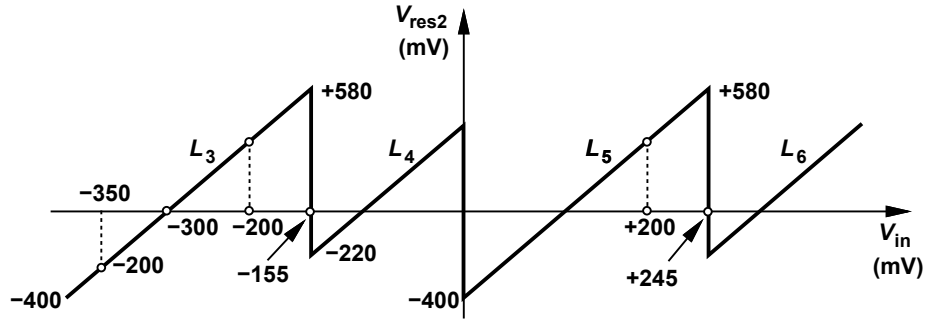


Figure 18.11 Second residue plot.

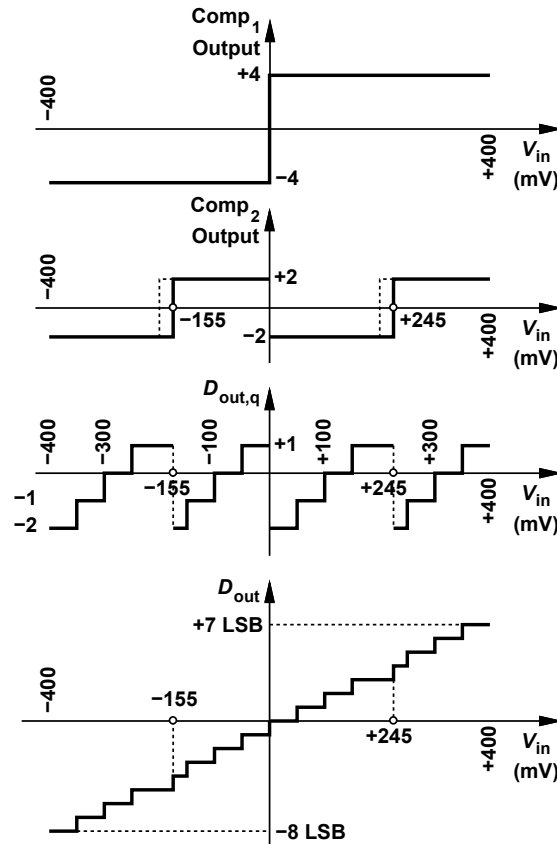


Figure 18.12 Effect of Comp2 offset on the characteristics of a pipelined ADC.

$V_{in} = -155$ mV. Beyond this input level, $V_{res2} = -220$ mV, yielding $D_{out,q} = -2$ LSB. The remainder of the plot is straightforward.

We must now combine this result with the outputs of Comp₁ and Comp₂ (Fig. 18.12) so as to obtain the overall 4-bit characteristic. We observe that the output of Comp₂ jumps up at $V_{in} = -155$ mV rather than at $V_{in} = -200$ mV. That is, for -250 mV $< V_{in} < -155$ mV, D_{out} remains at -5 LSB. At $V_{in} = -155$ mV, $D_{out,q}$ jumps down by 3 LSB, which occurs if the output of Comp₂ jumps up by 4 LSB, causing D_{out} to reach -4 LSB (Fig. 18.12). The remainder of the characteristic is plotted in a similar manner.

The result portrayed in Fig. 18.11 reveals a DNL of 45 mV $= V_{OS}/2$ because the transition points at $V_{in} = \pm 200$ mV are shifted by this amount. The INL is also equal to 45 mV (why?).

Our study here indicates that (1) the comparator offset in the first pipelined stage directly translates to

DNL and INL and must remain well below 1 LSB, (2) the comparator offset in the k -th stage is divided by 2^{k-1} , and (3) the 1-bit-per-stage pipelined architecture developed thus far is much less tolerant of comparator offsets than are SAR loops employing a single comparator.

18.3 Effect of Op-Amp Offset

In this section, we first learn that the flip-around MDAC cancels its op-amp offset, V_{OS} , but the non-flip-around topology does not. We then determine how the op-amp offset manifests itself in the residue and in the overall ADC characteristic.

Shown in Fig. 18.13(a) is the flip-around MDAC in the sampling mode. To find the effect of V_{OS} , we set the input to zero. Acting as a unity-gain stage, the op amp generates $V_{out} \approx V_{OS}$, and impresses this voltage on C_1 and C_2 . If $C_1 = C_2 = C$, each capacitor holds an amount of charge equal to CV_{OS} . In the

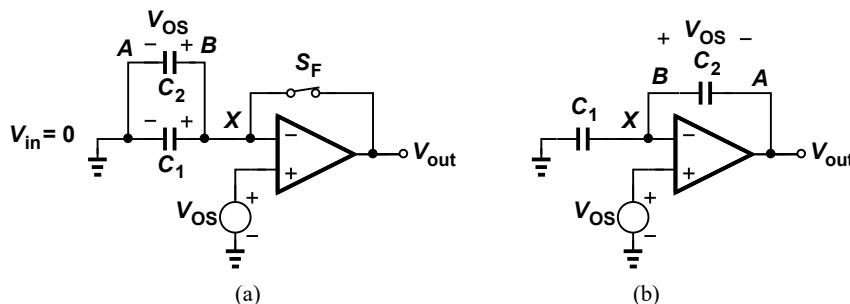


Figure 18.13 Effect of offset in (a) sampling, and (b) amplification modes in flip-around MDAC.

amplification mode [Fig. 18.13(b)], plate A of C_2 connects to the op-amp output. After the circuit settles, the high op-amp gain guarantees that $V_X \approx V_{OS}$, i.e., the charge on C_1 remains unchanged. Thus, the charge on C_2 must also be constant.³ Subtracting the voltage on C_2 from V_X , we obtain $V_{out} \approx 0$. Since with a zero input, the MDAC delivers a zero output, we say the offset of the op amp is canceled.

Let us repeat this analysis for the non-flip-around MDAC. Depicted in Fig. 18.14(a) in the sampling mode, the circuit forces V_{OS} across C_1 while C_2 is discharged. When the feedback switch turns off and the circuit enters the amplification mode [Fig. 18.14(b)], the op amp still requires that $V_X \approx V_{OS}$. That is, the charge on C_1 remains constant and so does that on C_2 . It follows that $V_{out} = V_X \approx V_{OS}$, revealing that the op amp offset directly adds to the MDAC output.

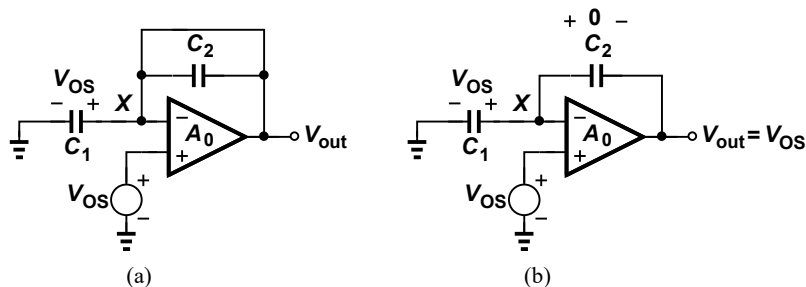


Figure 18.14 Effect of offset in (a) sampling, and (b) amplification modes in non-flip-around MDAC.

Next, we continue with the non-flip-around MDAC and write the residue produced by the first stage of a pipelined ADC as $V_{res1} = 2V_{in} \pm V_{REF} + V_{OS}$. The vertical shift leads to the plot shown in Fig. 18.15. In a manner similar to the previous sections, we digitize this characteristic by assuming that the MDAC is followed by an ideal 2-bit quantizer. As an example, we also assume $\pm V_{REF} = \pm 400$ mV and

³This is because any change in the charge on plate B must come from the right plate of C_1 .

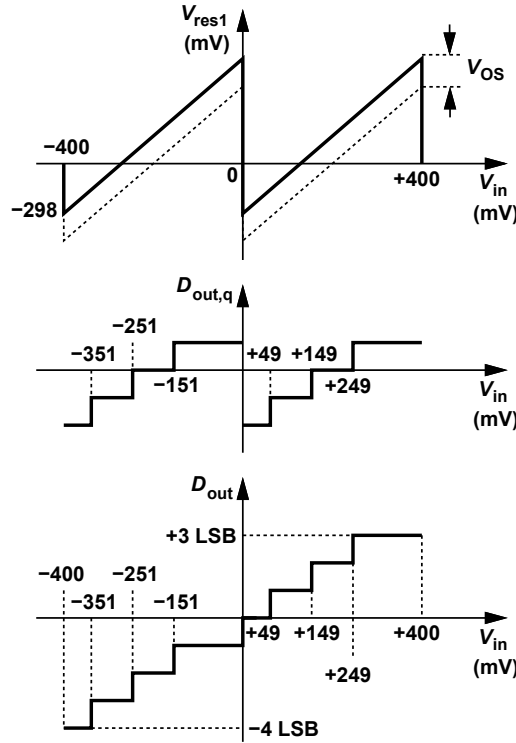


Figure 18.15 Effect of op-amp offset on the residue and digitized outputs.

$V_{OS} = +102$ mV (even though practical op amps exhibit much lower offsets). Since in Fig. 18.15,

$$V_{res1} = 2V_{in} + V_{REF} + V_{OS} = -298\text{mV} \quad (18.6)$$

at $V_{in} = -400$ mV, we note from the quantizer behavior in Fig. 18.8(a) that $D_{out,q} = -2$ LSB. For $D_{out,q}$ to change to -1 LSB, V_{res1} must cross -200 mV, which occurs at $V_{in} = -351$ mV (Fig. 18.15). The next transition of $D_{out,q}$ occurs at $V_{res1} = 0$, i.e., at $V_{in} = -251$ mV. Similarly, $D_{out,q}$ jumps to $+1$ LSB at $V_{res1} = +200$ mV and hence at $V_{in} = -151$ mV. For $V_{in} > 0$, the behavior of $D_{out,q}$ is identical. The reader is encouraged to compute the V_{in} values at which the transitions occur in this range.

In the last step of our analysis, we combine the comparator output with $D_{out,q}$ so as to derive the overall 3-bit ADC characteristic. The left and right sections of $D_{out,q}$ in Fig. 18.15 must be shifted down and up by 2 LSB, respectively. The result, D_{out} , suffers from a DNL of 51 mV ($= V_{OS}/2$) between $V_{in} = -400$ mV and $V_{in} = -300$ mV, and another DNL of 51 mV between $V_{in} = -200$ mV and $V_{in} = -100$ mV. The behavior repeats for $V_{in} > 0$. As expected, the overrange of V_{res1} translates to “saturated” outputs in the intervals $[-151 \text{ mV } 0]$ and $[+249 \text{ mV } +400 \text{ mV}]$.

18.4 Effect of Switch Charge Injection

As explained in Chapter 3, MOS switches release their channel charge as they turn off, a phenomenon that can lead to offset, gain error, and nonlinearity. The effect of charge injection is more complex in MDACs and is studied here. The following example proves crucial to our understanding.

Example 18.1

In the circuit of Fig. 18.16(a), V_{in} is constant and has a value of V_0 while C_2 holds no charge. At $t = 0$, S_1

turns off and at $t = t_1$, S_G turns on. Taking into account the channel charge of S_1 , plot V_{out} as a function of time.

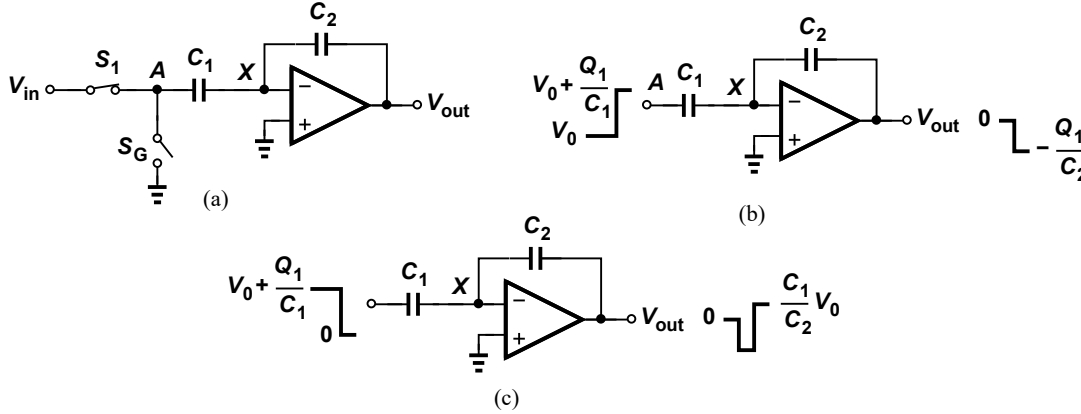


Figure 18.16 (a) Simple MDAC, (b) effect of charge injected by S_1 , and (c) removal of charge of S_1 by S_G .

Solution

Some of the channel charge of S_1 is injected onto node A when this switch turns off. Denoting this charge by Q_1 , we see that the charge on the left plate of C_1 changes from $C_1 V_0$ to $C_1 V_0 + Q_1$, yielding $V_A = V_0 + Q_1/C_1$ [Fig. 18.16(b)]. Amplified by a factor of $-C_1/C_2$, the input voltage change leads to $V_{out} = -(C_1/C_2)(Q_1/C_1) = -Q_1/C_2$. We can also predict this change by recognizing that, if $+Q_1$ lands on the left plate of C_1 , the right plate must draw $-Q_1$ from the left plate of C_2 because charge must be conserved at node X . Capacitor C_2 therefore develops a voltage of Q_1/C_2 and, with $V_X \approx 0$, an output equal to $-Q_1/C_2$.

When S_G turns on, V_A jumps from $V_0 + Q_1/C_1$ to zero, generating a change of $-(V_0 + Q_1/C_1)(-C_1/C_2)$ at the output [Fig. 18.16(c)]. That is, V_{out} goes from $-Q_1/C_2$ to

$$V_{out} = -\frac{Q_1}{C_2} + \frac{C_1}{C_2} \left(V_0 + \frac{Q_1}{C_1} \right) \quad (18.7)$$

$$= \frac{C_1}{C_2} V_0. \quad (18.8)$$

Interestingly, V_{out} is free from the charge injection of S_1 . This occurs because the amplifier simply responds to the input voltage *change*, i.e., the final input voltage minus the initial input voltage. As mentioned in Chapter 14, we call A a “driven” node.

18.4.1 Charge Injection in Non-Flip-Around MDAC

We begin with the non-flip-around MDAC and set its input to zero [Fig. 18.17(a)]. Suppose S_F turns off before S_1 does. The charge injected to the right is absorbed by the op amp, but that on the left arrives at X and must find its place. Since the op amp ensures that $V_X \approx 0$, C_1 cannot accept this charge, forcing it to land on the left plate of C_2 . This capacitor therefore sustains a voltage equal to Q_F/C_2 , yielding $V_{out} \approx -Q_F/C_2$. We say the channel charge of S_F introduces an offset in the residue; fortunately, this effect translates to a common-mode perturbation in fully-differential implementations (why?).

Let us now consider the charge injected by S_1 (while S_F is off) [Fig. 18.17(b)]. As explained in Example 18.1, this charge, Q_1 , arrives on the left plate of C_1 , causing its right plate to draw $-Q_1$ from the left plate of C_2 . The op amp still guarantees that $V_X \approx 0$ and hence $V_{out} = -Q_F/C_2 - Q_1/C_2$. We also have $V_A \approx Q_1/C_1$. But the amplification mode is not completed yet! Next, the left plate of C_1 is switched to

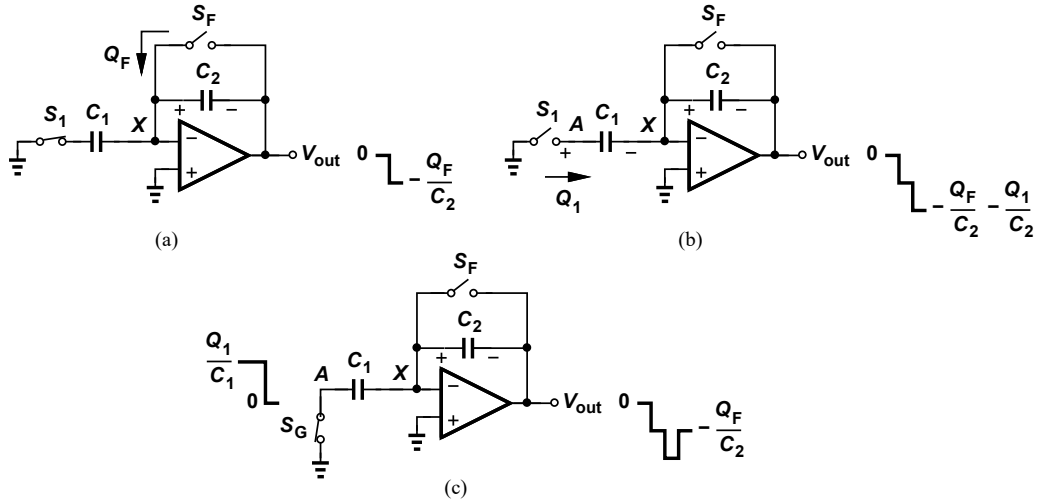


Figure 18.17 (a) Effect of charge injected by S_F , (b) effect of charge injected by S_1 , and (c) removal of charge when input drops to zero.

KV_{REF} . For simplicity, we assume $K = 0$ [Fig. 18.17(c)], and observe that V_A jumps from Q_1/C_1 to zero. Acting as an amplifier, the circuit then causes V_{out} to rise by $(Q_1/C_1)(C_1/C_2) = Q_1/C_2$. That is, $V_{out} = -Q_F/C_2$. The key point here is that V_{out} is independent of Q_1 .

The foregoing results can be summarized as follows. The charge injected by S_F onto X in Fig. 18.17(a) does not see a “resistive” or “low-impedance” path and hence remains on C_2 . Since this charge is independent of the input, it produces only an offset, which is suppressed in differential topologies. The channel charge of S_1 , on the other hand, depends on the input, leading to offset, gain error, and nonlinearity, but, because S_F turns off first, this charge has no effect on the circuit. We say this charge is unimportant because the left plate of C_1 switches from one “low-impedance” node (V_{in}) to another (ground or KV_{REF}).

Example 18.2

Explain how the MDAC of Fig. 18.17(a) behaves if S_1 turns off before S_F does.

Solution

Drawing the simplified circuit as in Fig. 18.18(a), we observe that Q_1 is deposited on C_1 while C_2 is reset. Capacitor C_1 now holds Q_1 but its charge can no longer change because its left plate remains floating. Note that $V_{out} = 0$. After S_F turns off, its charge, Q_F , arrives at X but cannot reside on C_1 [Fig. 18.18(b)]. Thus, as in the case studied above, Q_F simply lands on C_2 , yielding $V_{out} = -Q_F/C_2$.

When the left plate of C_1 swings to zero, the charge on C_1 is transferred to C_2 because $V_X \approx 0$. Carrying a total charge of $Q_1 + Q_F$, C_2 now generates $V_{out} = -(Q_1 + Q_F)/C_2$ [Fig. 18.18(c)]. The (input-dependent) charge of S_1 therefore manifests itself in the output, making this switching sequence inferior.

18.4.2 Charge Injection in Flip-Around MDAC

The effect of charge injection is more complex in this type of MDAC. We begin with the feedback switch, S_F [Fig. 18.19(a)]. As in the previous MDAC, we turn off S_F first. The charge, Q_F , lands on C_1 and C_2 . Sharing Q_F equally (because $C_1 = C_2$), these capacitors maintain the charge as the circuit enters the amplification mode. It is helpful to view the switching in “slow motion” [Fig. 18.19(b)], noting the *polarity* of the charge on C_1 and C_2 . After the transition is completed, $V_X \approx 0$, and the positive $Q_F/2$ charge on the right plate of C_1 travels to the left plate of C_2 , doubling the charge on this capacitor and producing $V_{out} = -Q_F/C_2$.

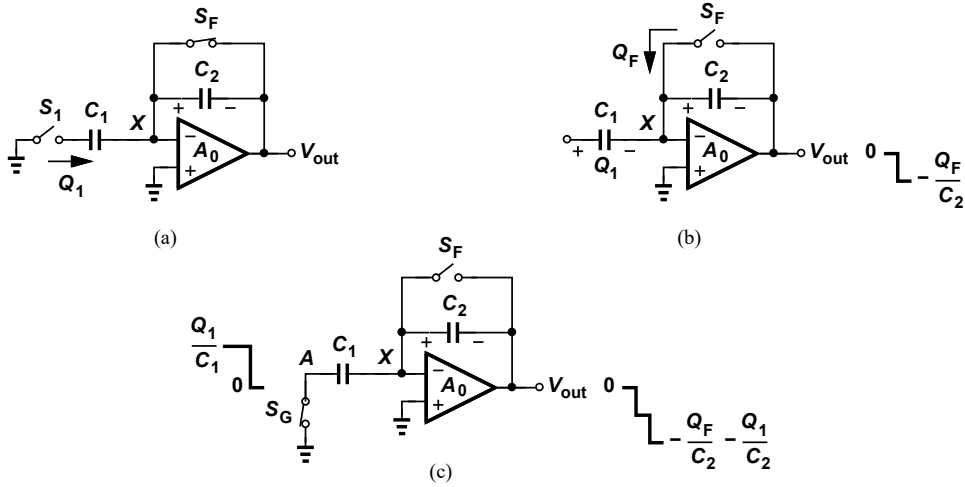


Figure 18.18 (a) Effect of charge injected by S_1 , (b) effect of charge injected by S_F , and (c) final result.

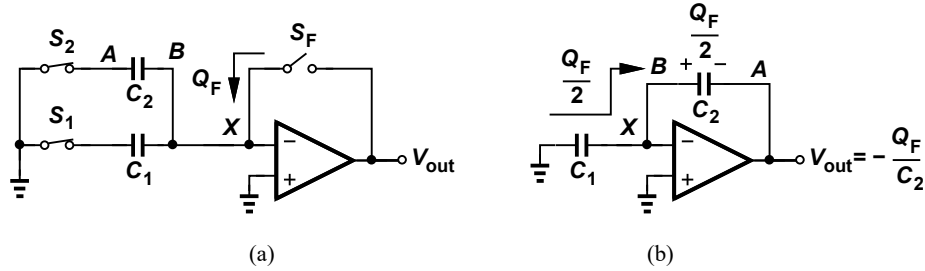


Figure 18.19 (a) Effect of charge injected by S_F , and (b) final result.

We must next consider the channel charge of S_1 and S_2 , assuming that S_F has turned off. These switches can be turned off simultaneously, but, for the sake of our study, we assume S_2 turns off first. As shown in Fig. 18.20(a), S_2 injects $+Q_2$ onto the left plate of C_2 , requiring that its right plate draw $-Q_2$ from the right plate of C_1 . When C_2 is flipped around the op amp [Fig. 18.20(b)], C_1 loses its charge to C_2 and generates $V_{out} = 0$. That is, the channel charge of S_2 does not affect the output. For the charge injected by S_1 [Fig. 18.20(c)], we note that the situation is similar to that in Example 18.1. The reader is encouraged to determine what happens if S_1 turns off before S_2 .

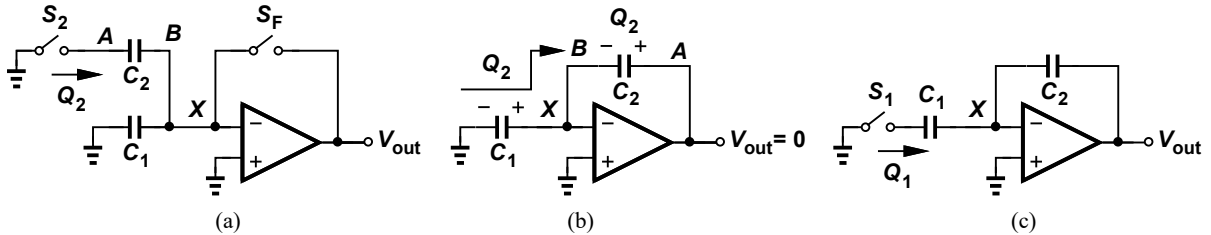


Figure 18.20 (a) Effect of charge injected by S_2 , (b) cancellation in amplification mode, and (c) effect of charge injected by S_1 .

In summary, our prescribed switching sequence guarantees that the input-dependent charge of the MDACs' sampling switches does not corrupt the output voltage. Only the channel charge of the feedback switch should be considered.

18.5 kT/C Noise

Our analysis of sampling circuits in Chapter 3 indicates that a capacitor inherits an rms noise voltage of $\sqrt{kT/C}$ when it samples a signal through a switch. In this section, we extend this result to MDACs. The charge injection studies in the previous section prove useful here as well.

18.5.1 Non-Flip-Around MDAC

Consider the switching sequence depicted in Fig. 18.17(a), noting that $V_X \approx 0$ and S_F freezes kT/C_2 across C_2 (why?). After S_F turns off, the thermal noise of S_1 plays no role just as its charge injection does not (see the previous section). When S_G turns on, its resistance contributes thermal noise, equal to kT/C_1 at A (because $V_X \approx 0$) and hence equal to $(kT/C_1)(C_1/C_2)^2$ at the output. The total output noise in the amplification mode is thus given by

$$V_{n,\text{rms}}^2 = \frac{kT}{C_2} + \frac{kT}{C_1} \left(\frac{C_1}{C_2} \right)^2 \quad (18.9)$$

$$= \left(1 + \frac{C_1}{C_2} \right) \frac{kT}{C_2} \quad (18.10)$$

$$= \frac{3kT}{C_2}. \quad (18.11)$$

18.5.2 Flip-Around MDAC

Referring to Fig. 18.19(a), we observe that each of the equal capacitors C_1 and C_2 carries an rms noise voltage of $\sqrt{kT/C_1}$ when S_F turns off. After the transition to Fig. 18.19(b), these uncorrelated voltages must be added properly:

$$V_{n,\text{rms}}^2 = \frac{2kT}{C_1}. \quad (18.12)$$

18.6 Effect of Op-Amp Nonidealities

We expect that op amps play a critical role in the performance of MDACs. In this section, we analyze the effect of the op amps' output impedance, gain, bandwidth, and noise, assuming a gain of 2 for MDACs.

18.6.1 Finite Output Impedance and Bandwidth

Sampling Mode Let us consider a non-flip-around MDAC in the sampling mode and model its op amp as shown in Fig. 18.21(a). We recognize that R_{out} experiences the Miller effect (why?), presenting a resistance

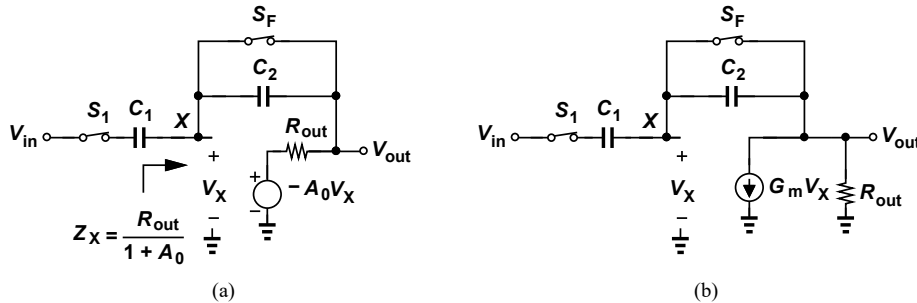


Figure 18.21 (a) Amplifier model including op amp output resistance, and (b) alternative model.

of $R_{out}/(1 + A_0)$ at the virtual ground node X . This value should preferably remain far below the on-resistance of S_1 so as not to degrade the acquisition speed. From another perspective, R_{out} does not allow

the op amp to maintain V_X constant if V_{in} changes with time. In order to appreciate the difficulty here, we turn to the Norton equivalent of the op amp [Fig. 18.21(b)] and write $|A_0| = G_m R_{out}$. It follows that $R_{out}/(1 + A_0) \approx 1/G_m$ because $G_m R_{out} \gg 1$. The op-amp transconductance must therefore be much greater than the inverse on-resistance of S_1 , a very stringent requirement. This analysis has assumed an infinite bandwidth for the op amp itself.

Example 18.3

Repeat the foregoing study for a flip-around MDAC.

Solution

In such an MDAC, *two* equal capacitors are tied to the input during the sampling mode. Equivalently, C_1 in Fig. 18.21 is doubled, i.e., the input capacitance is the same for both topologies. Appearing in series with both, the virtual-ground impedance, $1/G_m$, must still be as low as in the non-flip-around structure because $C_1 = 2C_2$ in Fig. 18.21. That is, the G_m requirements are similar.

It is possible to raise the transconductance of an op amp by increasing its open-loop gain. For example, the two-stage topology in Fig. 18.22(a) provides a gain of $A_0 = G_{m1} R_{O1} G_{m2} R_{O2}$. The virtual-ground impedance in Fig. 18.22(b) is therefore equal to $R_{O2}/(1 + A_0) \approx 1/(G_{m1} R_{O1} G_{m2})$ at low frequencies. Unfortunately, however, this arrangement generally requires frequency compensation and suffers from “internal”

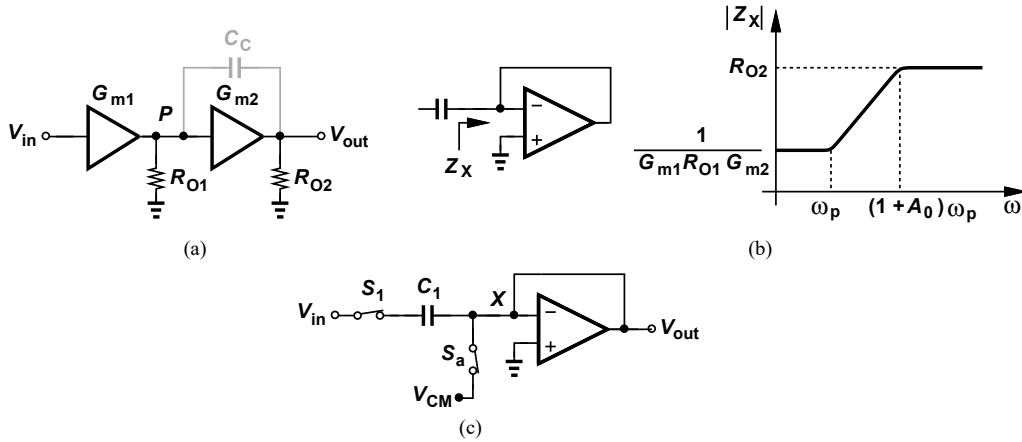


Figure 18.22 (a) Two-stage op amp model, (b) closed-loop input impedance, and (c) use of a CM switch.

bandwidth limitations. Specifically, the Miller multiplication of the compensation capacitor, C_c , establishes a dominant pole, ω_p , at node P given by⁴

$$\omega_p = \frac{1}{R_{O1}(1 + G_{m2}R_{O2})C_c}. \quad (18.13)$$

The op-amp transfer function can thus be approximated as

$$\frac{V_{out}}{V_{in}}(s) = \frac{G_{m1}R_{O1}G_{m2}R_{O2}}{R_{O1}(1 + G_{m2}R_{O2})C_c s + 1}. \quad (18.14)$$

We abbreviate this expression as $A(s) = A_0/(s/\omega_p + 1)$.

⁴Without Miller's approximation, this pole would be expressed as $1/[(R_{O1} + R_{O2} + G_{m2}R_{O1}R_{O2})C_c]$.

Let us return to Fig. 18.21(a) and write

$$Z_X = \frac{R_{out}}{1 + A(s)} \quad (18.15)$$

$$= \frac{R_{O2}}{1 + \frac{A_0}{\frac{s}{\omega_p} + 1}} \quad (18.16)$$

$$= \frac{R_{O2} \left(\frac{s}{\omega_p} + 1 \right)}{\frac{s}{\omega_p} + 1 + A_0}. \quad (18.17)$$

The virtual-ground impedance exhibits a zero at ω_p and a pole at $(1 + A_0)\omega_p$, behaving as shown in Fig. 18.22(b). If the MDAC input frequency in the sampling mode is greater than the *open-loop* pole frequency of the op amp, then $|Z_X|$ is higher than $1/(G_{m1}R_{O1}G_{m2})$ and can limit the acquisition speed. For this reason, we may short node X to a proper common-mode level by a switch [Fig. 18.22(c)].

Amplification Mode The settling time of the MDAC in the amplification mode generally becomes the speed bottleneck, thereby playing a central role in the performance of the overall ADC. The MDAC developments in Chapter 17 have assumed an infinitely fast op amp. We now study the settling behavior. The analysis is fairly long but the results are important. We use the Thevenin and Norton equivalents of the op amp to gain interesting insights.

Consider the MDAC shown in Fig. 18.23(a). We wish to determine the step response of the circuit. Assuming a one-stage op amp for simplicity and drawing the equivalent in Fig. 18.23(b), we surmise that the transfer function is of first order because C_1 and C_2 simply appear in series. The circuit can therefore be

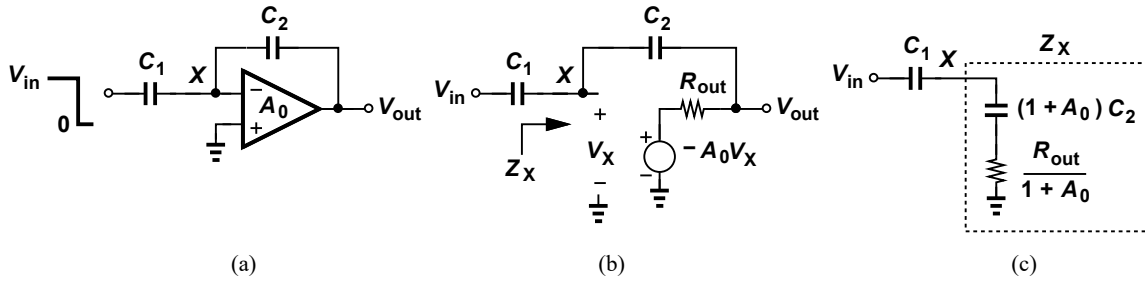


Figure 18.23 (a) Amplifier with a step input, (b) equivalent circuit, and (c) simplified model.

characterized by a single time constant, τ . Since both C_2 and R_{out} experience the Miller effect, we express Z_X as

$$Z_X = \frac{1}{1 + A_0} \left(\frac{1}{C_2 s} + R_{out} \right). \quad (18.18)$$

From the simplified structure in Fig. 18.23(c), we have

$$\tau = \frac{C_1(1 + A_0)C_2}{C_1 + (1 + A_0)C_2} \cdot \frac{R_{out}}{1 + A_0} \quad (18.19)$$

$$= \frac{R_{out}C_1}{\frac{C_1}{C_2} + 1 + A_0}. \quad (18.20)$$

Typically, $(1 + A_0)C_2 \gg C_1$ and hence

$$\tau \approx \frac{R_{out}C_1}{1 + A_0}. \quad (18.21)$$

Interestingly, the time constant is relatively independent of C_2 and can be obtained by simply writing $Z_X \approx R_{out}/(1 + A_0)$.

In reality, the MDAC incurs two other capacitances. As shown in Fig. 18.24(a), the op amp input capacitance, C_{in} , and the sampling capacitor of the second pipelined stage, C_b , must also be taken into account.

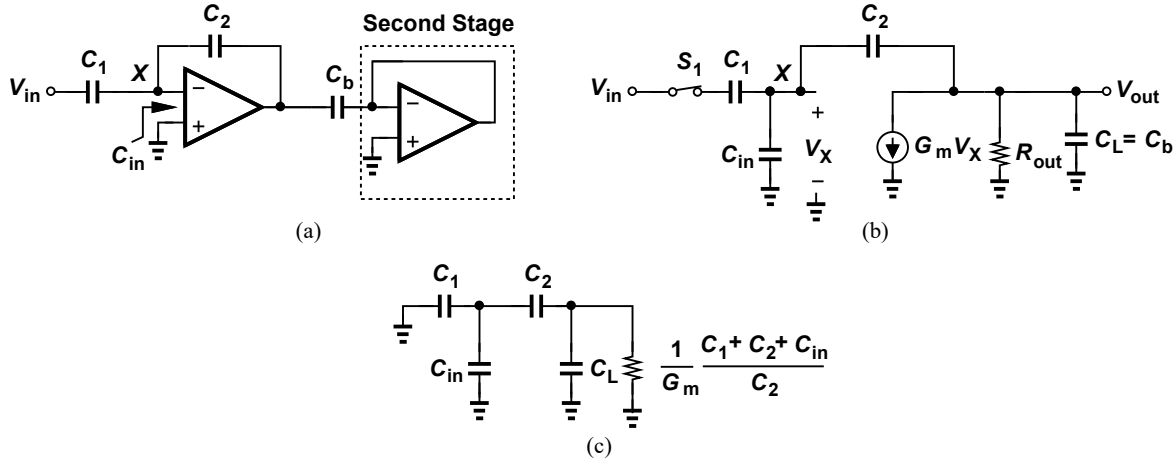


Figure 18.24 (a) Pipelined cascade, (b) equivalent circuit, and (c) simplified model.

Using the Norton equivalent of the op amp, we arrive at the small-signal model depicted in Fig. 18.24(b). A KCL at X yields $(V_{in} - V_X)C_1s + (V_{out} - V_X)C_2s = V_XC_{in}s$ and hence

$$V_X = \frac{V_{in}C_1 + V_{out}C_2}{C_{in} + C_1 + C_2}. \quad (18.22)$$

A KCL at the output node leads to

$$\left(V_{out} - \frac{V_{in}C_1 + V_{out}C_2}{C_{in} + C_1 + C_2} \right) C_2s + V_{out}C_Ls + \frac{V_{out}}{R_{out}} + G_m \frac{V_{in}C_1 + V_{out}C_2}{C_{in} + C_1 + C_2} = 0. \quad (18.23)$$

It follows that

$$\frac{V_{out}}{V_{in}}(s) = \frac{-C_1(G_m - C_2s)R_{out}}{R_{out}[(C_{in} + C_1)C_2 + (C_{in} + C_1)C_L + C_2C_L]s + C_{in} + C_1 + C_2 + G_mR_{out}C_2}. \quad (18.24)$$

The circuit is still of first order. If $G_mR_{out}C_2 \gg C_{in} + C_1 + C_2$, the time constant (the inverse of the pole frequency) emerges as

$$\tau = \frac{(C_{in} + C_1)C_2 + (C_{in} + C_1)C_L + C_2C_L}{G_mC_2}. \quad (18.25)$$

This expression can be rewritten as

$$\tau = \left(\frac{C_1 + C_2 + C_{in}}{C_2G_m} \right) \left[C_L + \frac{C_2(C_{in} + C_1)}{C_2 + C_{in} + C_1} \right]. \quad (18.26)$$

Let us construct an equivalent circuit for this time constant, noting that the first factor is the resistance and the second, the capacitance. Shown in Fig. 18.24(c), the result can be interpreted as follows. The equivalent resistance is given by $(C_1 + C_2 + C_{in})/(C_2 G_m)$, as if the op amp transconductance were *reduced* by the feedback factor, $C_2/(C_1 + C_2 + C_{in})$. The equivalent capacitance is given by C_L in parallel with the series combination of C_2 and $C_1 + C_{in}$.

Example 18.4

In the circuit of Fig. 18.24(b), $V_{in} = V_1 u(t)$. Compute the output response.

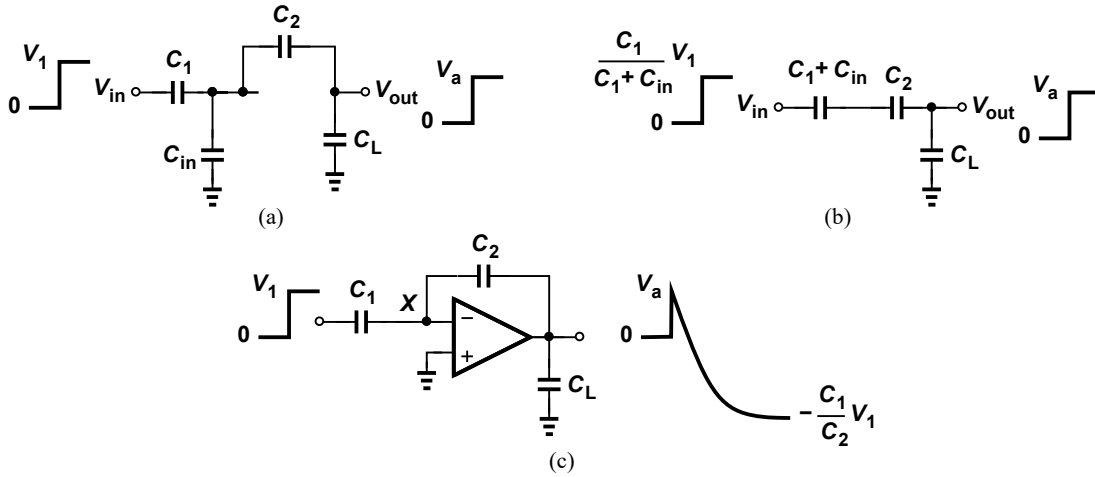


Figure 18.25 (a) Equivalent circuit for step response at $t = 0^+$, (b) simplified model, and (c) overall response.

Solution

The initial step transition travels only through the capacitors [Fig. 18.25(a)] (why?).⁵ The Thevenin equivalent of V_{in} , C_1 , and C_{in} yields a step of height $V_1 C_1 / (C_1 + C_{in})$ and an output capacitance of $C_1 + C_{in}$ [Fig. 18.25(b)]. The output therefore jumps from zero to

$$V_a = \frac{(C_1 + C_{in})C_2}{(C_1 + C_{in})C_2 + (C_1 + C_{in} + C_2)C_L} V_1 \frac{C_1}{C_1 + C_{in}}. \quad (18.27)$$

Next, the op amp initiates its own action, pushing V_{out} toward a value given by the closed-loop gain times V_1 . Approximating this gain by $-C_1/C_2$, we write the final value of V_{out} as $(-C_1/C_2)V_1$ [Fig. 18.25(c)]. It follows that

$$V_{out}(t) = V_{out}(\infty) + [V_{out}(0^+) - V_{out}(\infty)] \exp \frac{-t}{\tau} \quad (18.28)$$

$$\approx -\frac{C_1}{C_2} V_1 + \left(V_a + \frac{C_1}{C_2} V_1 \right) \exp \frac{-t}{\tau}, \quad (18.29)$$

where τ is given by (18.26). We observe that the initial jump to V_a *increases* the settling time, a byproduct of the MDAC's right-half-plane zero.

⁵A system generating an output jump in response to an input step exhibits at least one zero in its transfer function, as is the case in Eq. (18.24).

We should make two remarks at this point. First, both the loop gain and the load capacitance of the MDAC change from the sampling mode to the amplification mode, and hence so does its phase margin. In the former mode, the feedback factor is unity, suggesting that the phase margin is lower. For this reason, the op amp is typically compensated for unity-gain feedback.

Second, the op amp model in Fig. 18.24(b) is not accurate for two-stage topologies. The “internal” pole of a two-stage op amp inevitably creates a second-order response, requiring a more elaborate analysis.

18.6.2 Finite Op-Amp Gain

Recall from Chapter 17 that MDACs ideally amplify the input voltage by a factor of 2. The finite gain of the op amp creates an error in this factor, introducing DNL and INL. Let us begin with some basic examples.

Example 18.5

A 3-bit differential ADC (quantizer) with $\pm V_{REF} = \pm 400$ mV experiences a gain error of -10% . Sketch the ADC’s characteristic. Does the ADC suffer from nonlinearity?

Solution

In the ideal situation, the analog LSB is equal to 100 mV. A gain error of -10% means that the slope is 10% less. Thus, the LSB is *larger* by 10% while the input range is still limited to ± 400 mV. Depicted in Fig. 18.26(a) is this characteristic.

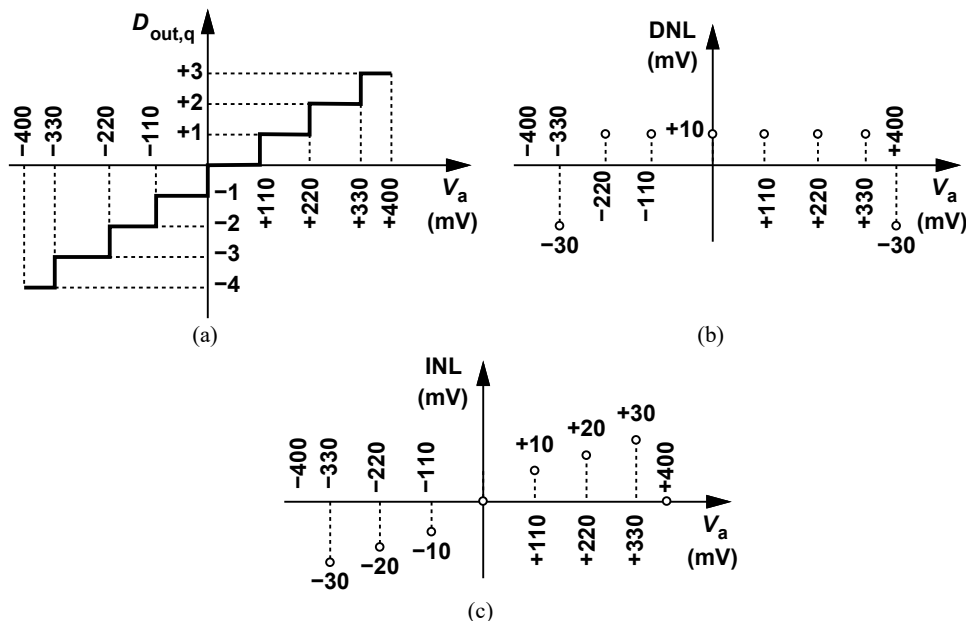


Figure 18.26 (a) Effect of gain error, (b) resulting DNL, and (c) resulting INL.

If we simply define the LSB to be 110 mV (rather than 100 mV), then the ADC remains linear because all of its transition points lie on a straight line. Nevertheless, to improve our analytical skills, let us assume 1 LSB = 100 mV and find the DNL and INL. The DNL between -400 mV and -330 mV ($D_{out,q} = -4$ LSB) is computed as -330 mV $- (-400$ mV) $- 100$ mV = -30 mV. The other values are obtained in a similar manner, leading to the profile shown in Fig. 18.26(b). Integrating these values produces the INL plot depicted in Fig. 18.26(c).⁶

⁶Recall from Chapter 12 that the DNL and INL can be plotted versus the output code rather than versus V_{in} . The general behavior is still the same.

What happens if the gain error is -40% ? An LSB of 140 mV implies that the code transitions at $V_{in} = \pm 300$ mV are shifted to ± 420 mV, respectively, falling outside the input range and prohibiting digital output values of -4 LSB and $+3$ LSB. Thus, even though $DNL < 1$ LSB, missing codes can occur, but only at the ends of the characteristic.

Example 18.6

A non-flip-around MDAC is followed by an ideal 3-bit quantizer [Fig. 18.27(a)]. Assuming $K = \pm 1/2$, $\pm V_{REF} = \pm 400$ mV, and a gain error of -10% for the MDAC, (a) plot the residue and the quantizer output as a function of the input, V_{in} , and (b) construct the overall ADC characteristic and its DNL profile.

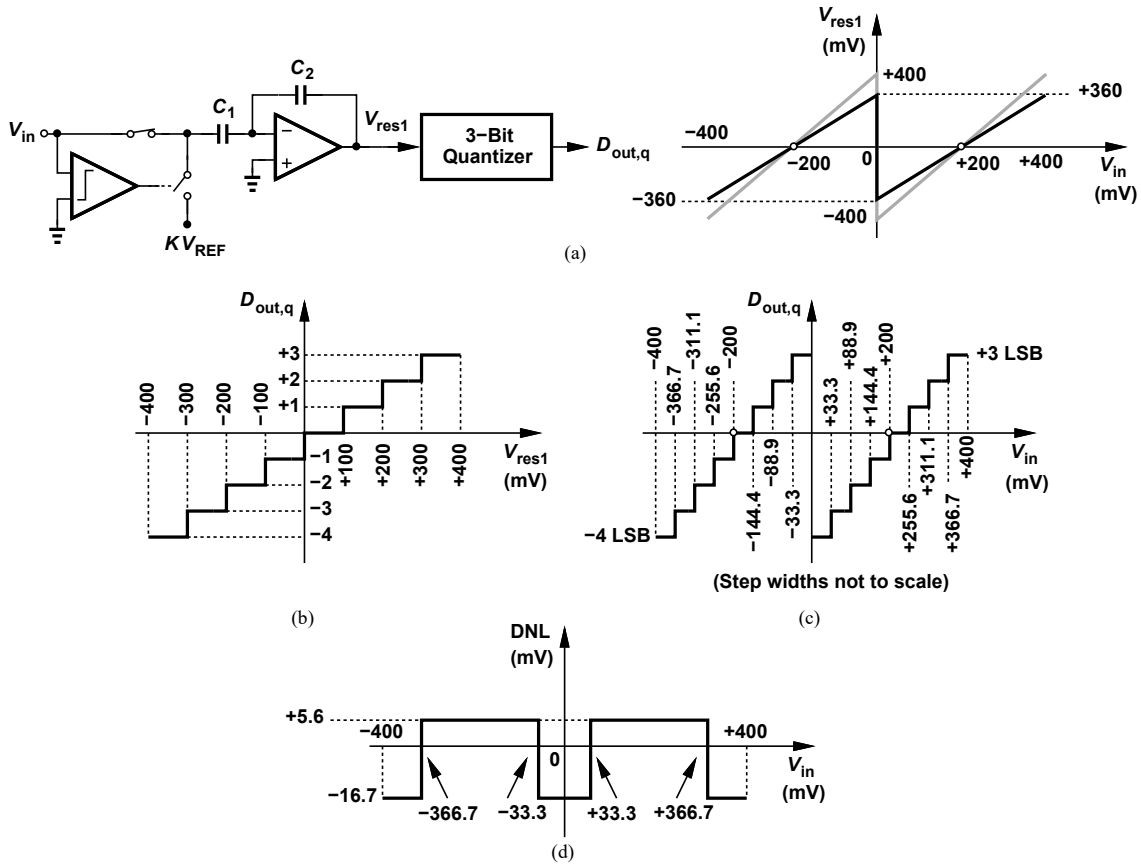


Figure 18.27 (a) Pipelined cascade and its residue plot, (b) ideal 3-bit quantizer characteristic, (c) quantized residue, and (d) resulting DNL.

Solution

(a) We have $V_{res1} = 0.9(2V_{in} \pm V_{REF})$ (why?), noting that it begins from -360 mV at $V_{in} = -400$ mV and reaches zero at $V_{in} = -200$ mV [Fig. 18.27(a)]. The quantizer characteristic, shown in Fig. 18.27(b), indicates that its input, V_{res1} , must cross decision thresholds at -300 mV, -200 mV, etc., for its output code to change. Our task is therefore to map V_{in} through V_{res1} to $D_{out,q}$.

Since $D_{out,q} = -4$ LSB for -400 mV $< V_{res1} < -300$ mV, we compute the corresponding range for V_{in} by writing -400 mV $< 0.9(2V_{in} + 400$ mV) < -300 mV and hence -422 mV $< V_{in} < -366.7$ mV. We repeat this calculation for $D_{out,q} = -3$ LSB, etc., arriving at the plot in Fig. 18.27(c). Note that $D_{out,q}$ reaches zero at $V_{in} = -200$ mV (why?).

(b) To construct the overall characteristic, we combine the MDAC comparator output with $D_{out,q}$. That is, we shift the right section of $D_{out,q}$ down by 4 LSB and the left section up by 4 LSB. The reader is encouraged to plot this characteristic. The DNL profile is shown in Fig. 18.27(d). (It is common practice to connect the dots by a solid line.)

Non-Flip-Around MDAC In order to compute the effect of the op amp's gain, we construct the equivalent circuit shown in Fig. 18.28(a) and seek V_{out}/V_{in} . We can follow one of three analysis methods here: (1) apply a step at the input and find the change in the output; (2) assume a general V_{in} and solve the circuit in the frequency domain; or (3) assume a general V_{in} and solve the circuit in the time domain. We pursue the second approach here and leave the other two as an exercise for the reader.

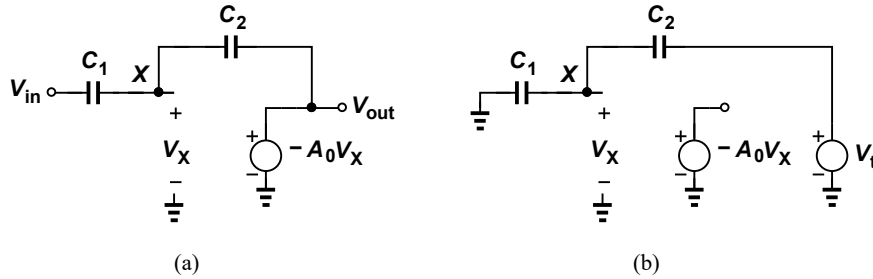


Figure 18.28 (a) Amplifier model and (b) open-loop circuit.

The current through C_1 is equal to $(V_{in} - V_X)C_1 s$. Upon flowing through C_2 , this current generates a voltage of $(V_{in} - V_X)C_1/C_2$. That is,

$$V_X - V_{out} = (V_{in} - V_X) \frac{C_1}{C_2}. \quad (18.30)$$

Since $V_{out} = -A_0 V_X$,

$$\frac{V_{out}}{V_{in}} = \frac{-1}{1 + \frac{1}{A_0} \left(1 + \frac{C_1}{C_2}\right)} \frac{C_1}{C_2}. \quad (18.31)$$

Before interpreting this result, it is also instructive to compute the circuit's loop gain. Breaking the loop as in Fig. 18.28(b) and injecting a test voltage, V_t , we measure the returned signal, $-A_0 V_X$, obtaining a loop gain of $A_0 C_2 / (C_1 + C_2)$.

Equation (18.31) provides a wealth of information. First, we recognize that the term $(1/A_0)(1 + C_1/C_2)$, which is the inverse of the loop gain, produces gain error. Second, assuming that the loop gain is much greater than unity, we write

$$\frac{V_{out}}{V_{in}} \approx - \left[1 - \frac{1}{A_0} \left(1 + \frac{C_1}{C_2} \right) \right] \frac{C_1}{C_2}, \quad (18.32)$$

obtaining a normalized gain error of $\epsilon = -(1/A_0)(1 + C_1/C_2) = -3/A_0$ if $C_1 = 2C_2$. For example, if a 10-bit ADC requires that this error be less than 0.1%, then A_0 must exceed 3,000. This point underscores op amp design difficulties in high-resolution pipelined ADCs.

Example 18.7

The non-flip-around MDAC nominally generates a residue equal to $2V_{in} \pm V_{REF}$. Does the gain error found above also manifest itself in the V_{REF} term?

Solution

Yes, it does. From the circuit shown in Fig. 18.29, where $K = \pm 1/2$, we recognize that the change in V_A is amplified by Eq. (18.32). We write

$$V_{res1} = -(KV_{REF} - V_{in})(1 + \epsilon) \frac{C_1}{C_2} \quad (18.33)$$

$$= (1 + \epsilon)(2V_{in} \pm V_{REF}). \quad (18.34)$$

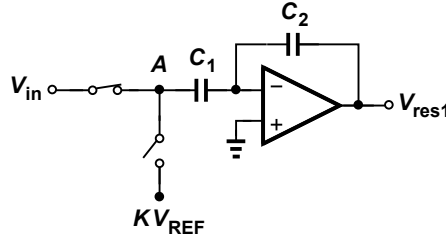


Figure 18.29 MDAC example.

The reader may wonder whether the ratio C_1/C_2 can be chosen slightly greater than 2 so as to compensate for the $1 + \epsilon$ term. This is difficult because ϵ depends on A_0 , which varies considerably with PVT.

Let us determine how the MDAC gain error translates to DNL and INL. The reader is encouraged to review Example 18.6. We observe from Eq. (18.33) that V_{res1} still crosses zero at $V_{in} = \pm V_{REF}/2$, but fails to reach $\pm V_{REF}$ at the extremes. As illustrated in Fig. 18.30(a), lines L_1 and L_2 are “pinned” at $V_{in} = \pm V_{REF}/2$ and simply rotate around these points due to the gain error. Thus, V_{res1} is equal to $-(1 + \epsilon)V_{REF}$ at $V_{in} = -V_{REF}$

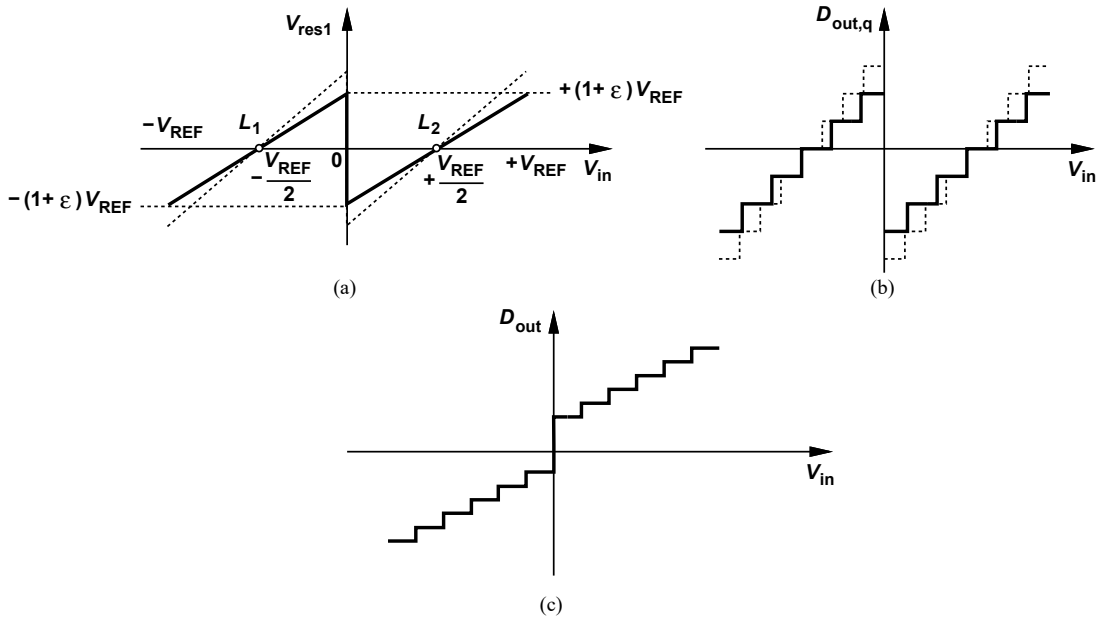


Figure 18.30 Effect of gain error on (a) residue plot, (b) quantized residue, and (c) overall characteristic.

and $+(1 + \epsilon)V_{REF}$ at $V_{in} = 0$ and $V_{in} = +V_{REF}$. Now, we assume the MDAC is followed by an ideal quantizer and, from the conceptual diagram of Fig. 18.3(d), predict that digitizing V_{res1} yields the behavior in Fig. 18.30(b). Combining the result with the MDAC comparator output produces the characteristic in Fig. 18.30(c).

As in the examples presented in Section 18.1.2, suppose the MDAC is followed by an ideal 3-bit quantizer having the characteristic shown in Fig. 18.2(b). Selecting $V_{REF} = 400$ mV (and hence 1 LSB = 50 mV) and $\epsilon = -0.3$,⁷ we recognize that L_1 and L_2 in Fig. 18.30(a) are bounded between $-0.7V_{REF} = -280$ mV and $+0.7V_{REF} = +280$ mV. Since the quantizer's decision thresholds occur at -300 mV, -200 mV, etc., we compute the V_{in} values at which V_{res1} reaches those thresholds. For L_1 , we have $V_{res1} = 0.7(2V_{in} + 400$ mV), concluding that V_{res1} crosses -200 mV and -100 mV at $V_{in} = -343$ mV and -271 mV, respectively [Fig. 18.31(a)]. Due to the large gain error, $|V_{res1}| < 300$ mV. Similarly, the decision thresholds of $D_{out,q}$ between -200 mV and zero are equal to -129 mV and -57 mV. The right section of the plot is obtained by setting $0.7(2V_{in} - 400$ mV) equal to -200 mV, $\dots$, $+200$ mV. Note that $D_{out,q}$ does not reach -4 LSB or $+3$ LSB because $|V_{res1}| < 300$ mV.

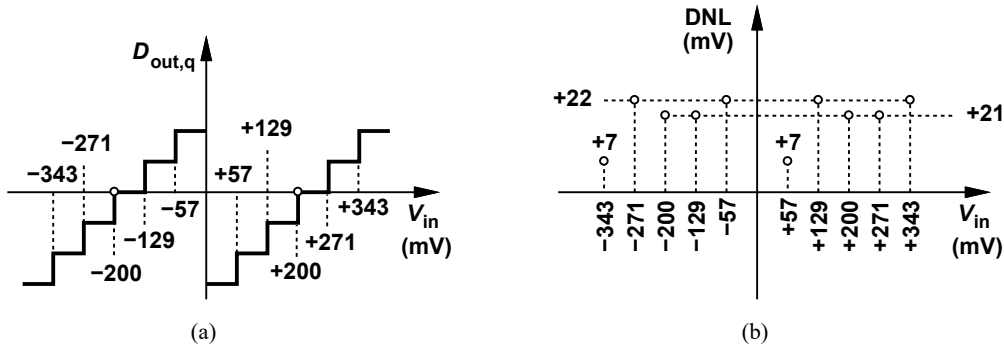


Figure 18.31 (a) Effect of gain error, and (b) resulting DNL.

The last step of our analysis involves adding the MDAC comparator characteristic to that in Fig. 18.31(a). Thus, the left section of Fig. 18.31(a) is shifted down by 4 LSB and the right section up by 4 LSB. Constructing this plot, the reader can show that codes -8 LSB, -1 LSB, 0 LSB, and $+7$ LSB are missing.

With an overall analog LSB of 50 mV, we can compute the DNL. For example, for -400 mV $< V_{in} < -343$ mV, $D_{out} = -3$ LSB, yielding $DNL = -343$ mV $- (-400$ mV) $- 50$ mV, which amounts to $+7$ mV. Shown in Fig. 18.31(b), the result can be normalized to 50 mV so as to express the DNL in LSB. Since D_{out} jumps from -2 LSB to $+1$ LSB at $V_{in} = 0$, we can say the two missing codes signify two DNL values of -1 LSB $= -50$ mV at $V_{in} = 0$. (These are not shown in the DNL plot.)

Flip-Around MDAC The effect of the finite op-amp gain on the flip-around MDAC is quantified with the aid of Fig. 18.32(a). The total charge on the right plates of C_1 and C_2 is equal to $-(C_1 + C_2)V_{in}$. We

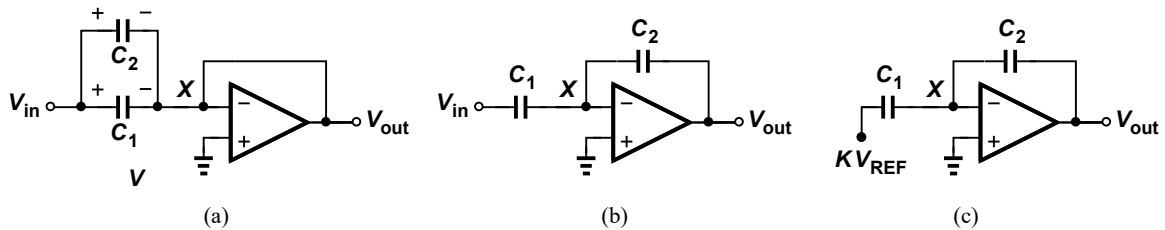


Figure 18.32 Flip-around stage in (a) sampling mode, (b) first amplification phase, and (c) second amplification phase.

study the transition to the amplification mode in “slow motion,” assuming that V_{in} is constant. When C_2 flips around the op amp [Fig. 18.32(b)], the total charge at X must still remain equal to $-(C_1 + C_2)V_{in}$. Since

⁷With such an unrealistically large gain error, the approximation in Eq. (18.32) is somewhat inaccurate.

$V_X = -V_{out}/A_0$, we equate the charges before and after the transition:

$$-C_1 \left(V_{in} + \frac{V_{out1}}{A_0} \right) - C_2 \left(V_{out1} + \frac{V_{out1}}{A_0} \right) = -(C_1 + C_2)V_{in}. \quad (18.35)$$

That is,

$$V_{out1} = \frac{C_2 V_{in}}{C_2 + \frac{C_1 + C_2}{A_0}}. \quad (18.36)$$

Now, the left plate of C_1 swings to KV_{REF} [Fig. 18.32(c)], where $K = \pm 1$. This change, $KV_{REF} - V_{in}$, is amplified by the circuit and added to V_{out1} . The gain in this mode is given by Eq. (18.31), leading to

$$V_{out2} = \frac{C_2 V_{in}}{C_2 + \frac{C_1 + C_2}{A_0}} - \frac{1}{1 + \frac{1}{A_0}(1 + \frac{C_1}{C_2})} \frac{C_1}{C_2} (KV_{REF} - V_{in}) \quad (18.37)$$

$$= \frac{1}{1 + \frac{1}{A_0}(1 + \frac{C_1}{C_2})} \left(1 + \frac{C_1}{C_2} \right) V_{in} - \frac{K}{1 + \frac{1}{A_0}(1 + \frac{C_1}{C_2})} \frac{C_1}{C_2} V_{REF}. \quad (18.38)$$

If $C_1 = C_2$ and $K = \pm 1$,

$$V_{out2} = \frac{1}{1 + \frac{2}{A_0}} (2V_{in} \pm V_{REF}), \quad (18.39)$$

implying a gain error of $\epsilon \approx -2/A_0$. Note that the magnitude of this error is less than that of the non-flip-around MDAC because of the greater loop gain in the flip-around topology.

Equation (18.39) reveals that the residue in the flip-around MDAC incurs less gain error but its behavior is similar to that of the non-flip-around counterpart.

Example 18.8

A differential N -bit pipelined ADC employs a 1-bit non-flip-around MDAC as its first stage. Assuming the other stages are ideal, determine the minimum op-amp gain necessary for $DNL = 0.2$ LSB.

Solution

Let us begin with two observations in regards to the ideal case. First, owing to the MDAC gain of 2, the analog LSB at the input of the *next* stage is twice that of the main input. Second, since $V_{res1} = 2V_{in} \pm V_{REF}$, for the residue to increase by 2 LSB, V_{in} must rise by 1 LSB. This is how input values corresponding to code transitions are computed.

We next express the actual MDAC output as $V_{res1} = (1 + \epsilon)(2V_{in} \pm V_{REF})$, noting that $V_{REF} = 2^{N-1}\Delta_{LSB}$, where Δ_{LSB} is the analog LSB of the overall ADC.⁸ Recall from Example 18.6 that the greatest DNL occurs near $V_{in} = \pm V_{REF}$ and $V_{in} \approx 0$. We begin from $V_{in} = -V_{REF}$ and raise the input so as to reach a code transition. For this to occur, V_{res1} must increase to $-V_{REF} + 2\Delta_{LSB}$ (Fig. 18.33):

$$(1 + \epsilon)(2V_{in1} + V_{REF}) = -V_{REF} + 2\Delta_{LSB}. \quad (18.40)$$

It follows that

$$V_{in1} = \frac{(-2 - \epsilon)V_{REF} + 2\Delta_{LSB}}{2(1 + \epsilon)}. \quad (18.41)$$

⁸For a differential ADC, the full scale is given by $2V_{REF} = 2^N \Delta_{LSB}$.

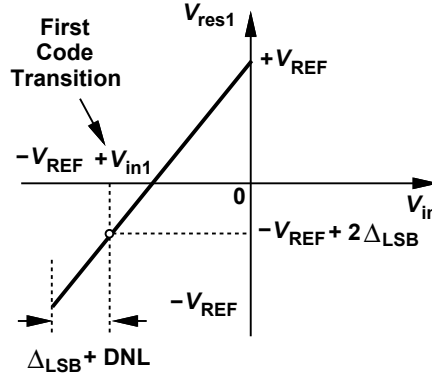


Figure 18.33 Effect of gain error.

To obtain the corresponding DNL, we subtract Δ_{LSB} from the absolute value of the difference between this value and $V_{in} = -V_{REF}$:

$$|DNL_{max}| = \frac{\epsilon}{2(1 + \epsilon)} (V_{REF} - 2\Delta_{LSB}). \quad (18.42)$$

For $|DNL| < 0.2 \text{ LSB}$, we have $[|\epsilon|/2(1 + |\epsilon|)](2^{N-1} - 2)\Delta_{LSB} < 0.2\Delta_{LSB}$. If $|\epsilon| \ll 1$,

$$|\epsilon| < \frac{0.2}{2^{N-2} - 1}. \quad (18.43)$$

For a non-flip-around MDAC, $\epsilon \approx -3/A_0$ and hence

$$A_0 > 15(2^{N-2} - 1). \quad (18.44)$$

A 10-bit ADC, for example, requires an op-amp gain of 3,800.

18.6.3 Op-Amp Noise

In high-resolution pipelined ADCs, the op amp power consumption is dictated by both its speed and its noise. We expect that the noise manifests itself in the sampling mode as well as the amplification mode. In this section, we consider only thermal noise sources, but the flicker noise of op amps may also become problematic in some cases.

Op-Amp Noise Representation We know from basic analog design that it is desirable to refer the noise of amplifiers to their input. However, in the presence of bandwidth limitations, the situation is more complex. We begin with the simple op amp model shown in Fig. 18.34 and note that the only noise quantity that can be physically measured is at the output and appears as a series voltage source having a spectral density of $\overline{V_{n,op}^2}^9$.

Example 18.9

Compute the total output rms noise for the circuit of Fig. 18.35(a). Assume $\overline{V_{n,op}^2}$ is white.

⁹One can alternatively construct a Norton equivalent for the op amp and its noise.

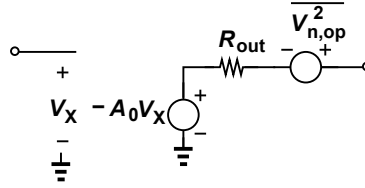


Figure 18.34 Simple op amp noise model.

Solution

Setting $V_{in} = V_X$ to zero, we note that $V_{n,op}$ experiences a first-order low-pass response as it travels to the

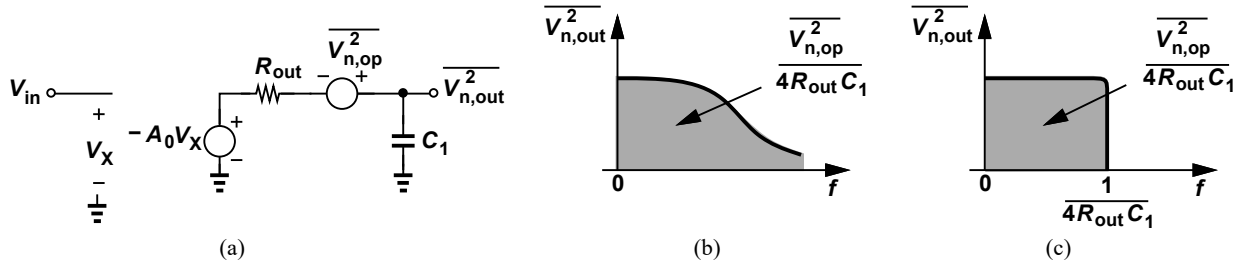


Figure 18.35 (a) Op amp driving a capacitance, (b) output noise spectrum, and (c) equivalent noise bandwidth.

output. The output noise spectrum is given by the product of $\overline{V_{n,op}^2}$ and the magnitude squared of the transfer function [Fig. 18.35(b)]:

$$\overline{V_{n,out}^2} = \overline{V_{n,op}^2} \frac{1}{R_{out}^2 C_1^2 (2\pi f)^2 + 1}. \quad (18.45)$$

To obtain the total output rms noise, we must integrate this spectrum from zero to $+\infty$ and take the square root of the result. Recall from Chapter 3 that the integration of $4kTR/[R^2 C^2 (2\pi f)^2 + 1]$ yields a value of kT/C . We then write Eq. (18.45) as

$$\overline{V_{n,out}^2} = \frac{\overline{V_{n,op}^2}}{4kTR_{out}} \frac{4kTR_{out}}{R_{out}^2 C_1^2 (2\pi f)^2 + 1}, \quad (18.46)$$

and, assuming that $\overline{V_{n,op}^2}$ is constant, we obtain

$$\int_0^\infty \overline{V_{n,out}^2} df = \frac{1}{4R_{out} C_1} \overline{V_{n,op}^2}. \quad (18.47)$$

The key point here is that the op amp white noise is equivalently integrated across a bandwidth of $1/(4R_{out} C_1)$ [Fig. 18.35(c)]. We call this quantity the “noise bandwidth” of the circuit and recognize that it is equal to $\pi/2$ times the -3 -dB bandwidth, $1/(2\pi R_{out} C_1)$. The rms noise emerges as

$$V_{n,out,rms} = \sqrt{\frac{\overline{V_{n,op}^2}}{4R_{out} C_1}}. \quad (18.48)$$

We analyze noise by going through a number of intuitive steps. Consider the simple op amp shown in Fig. 18.36. The noise of M_1 and M_2 appears directly at the input, and that of M_3 and M_4 can be referred to this port as well. Thus, V_n can represent the noise of all four transistors. For example, the white noise of

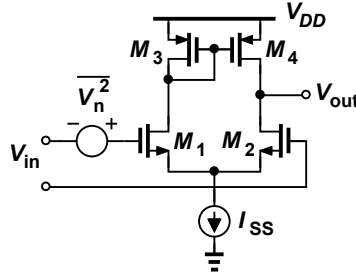


Figure 18.36 Simple op amp and its input-referred noise.

M_1 - M_4 reflects itself as a flat spectrum in V_n . We could simply say that the spectrum of V_n must be integrated across the MDAC's bandwidth so as to find the rms noise. But let us approach the problem more carefully.

Placing the op amp in unity-gain feedback, we note from Fig. 18.37(a) that the output noise spectrum is approximately equal to V_n . We wish to construct a Thevenin model and must compute the output resistance of the circuit. Modeling the op amp by a dependent current source and an open-loop output resistance [Fig. 18.37(b)], we can readily prove that the closed-loop output resistance is obtained from

$$R_{out1} = \frac{1}{G_m} || R_{out}. \quad (18.49)$$

If the open-loop gain, $G_m R_{out}$, is much greater than unity, we have $R_{out1} \approx 1/G_m$.

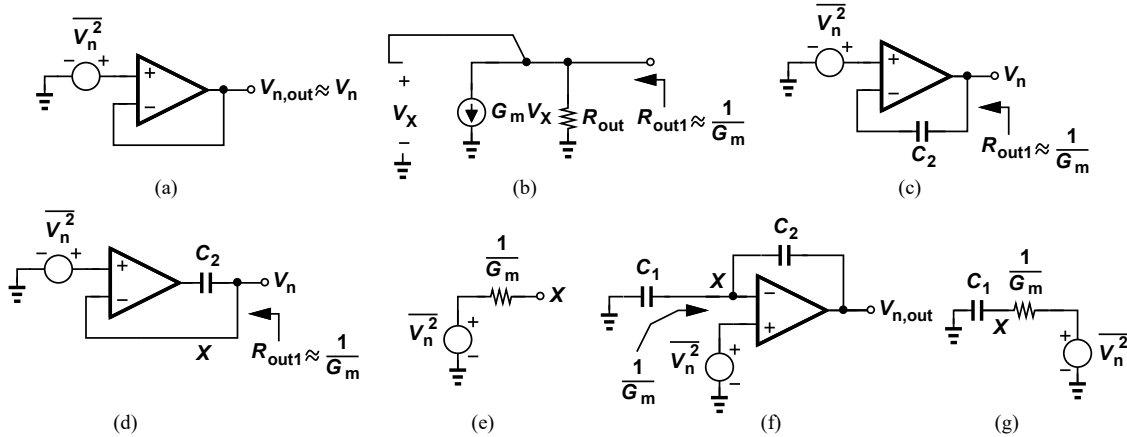


Figure 18.37 (a) Unity-gain amplifier and its input noise, (b) equivalent circuit for finding output resistance, (c) addition of C_2 to feedback loop, (d) impedance seen at X , (e) simplified model, (f) amplifier with its input noise, and (g) equivalent circuit.

Next, we insert a capacitor in the feedback path [Fig. 18.37(c)] and recognize that the output noise is still equal to V_n if the input capacitance of the op amp is neglected. Since C_2 carries no current and, therefore, sustains a zero voltage drop, we can prove that $R_{out1} \approx 1/G_m$. Now, we move C_2 to the output of the op amp [Fig. 18.37(d)]. The reader can prove that the noise and output impedance measured at X are the same as before if $R_{out} C_2$ is large enough and hence $1/(2\pi R_{out} C_2)$ is well below the frequencies of interest. We conclude that the noise Thevenin equivalent of the circuit consists of a voltage source equal to V_n and an output resistance equal to $1/G_m$ [Fig. 18.37(e)]. In the next step, we attach C_1 to the structure [Fig. 18.37(f)].

Using the Thevenin equivalent of the op amp and C_2 [Fig. 18.37(g)], we can express V_X as

$$V_X = \frac{\frac{1}{C_1 s}}{\frac{1}{C_1 s} + \frac{1}{G_m}} V_n \quad (18.50)$$

$$= \frac{G_m}{C_1 s + G_m} V_n. \quad (18.51)$$

This voltage generates a current of $V_X C_1 s$ in C_1 , yielding

$$V_{n,out} = \frac{V_X C_1 s}{C_2 s} + V_X \quad (18.52)$$

$$= \left(1 + \frac{C_1}{C_2}\right) \frac{G_m}{C_1 s + G_m} V_n. \quad (18.53)$$

This is the output noise of the overall amplifier due to the op amp. As expected, the low-frequency gain from the V_n to $V_{n,out}$ is equal to $1 + C_1/C_2$. The 3-dB bandwidth in hertz is given by $f_{3dB} = G_m/(2\pi C_1)$.

Equation (18.53) allows us to obtain the amplifier output noise spectrum and integrate it so as to arrive at the total rms noise. If V_n is white,

$$V_{n,out,rms}^2 = \left(1 + \frac{C_1}{C_2}\right)^2 \overline{V_n^2} \int_0^\infty \frac{G_m^2}{4\pi^2 C_1^2 f^2 + G_m^2} df, \quad (18.54)$$

where $\overline{V_n^2}$ denotes the *one-sided* spectrum of V_n . Since

$$\int \frac{dx}{a^2 x^2 + 1} = \frac{1}{a} \tan^{-1} ax, \quad (18.55)$$

we have

$$V_{n,out,rms}^2 = \left(1 + \frac{C_1}{C_2}\right)^2 \overline{V_n^2} \frac{G_m}{4C_1}. \quad (18.56)$$

We can view this result as follows: $\overline{V_n^2}$ is amplified by a factor of $(1 + C_1/C_2)^2$ and integrated over a bandwidth equal to $G_m/(4C_1)$. This “noise bandwidth” is $\pi/2$ times $f_{3dB} = G_m/(2\pi C_1)$.

The output rms noise can be divided by the closed-loop gain to find the input-referred value:

$$V_{n,in,rms}^2 = \overline{V_n^2} \frac{G_m}{4C_1}. \quad (18.57)$$

In summary, the total input-referred noise is equal to the spectrum of V_n multiplied by the MDAC noise bandwidth. We utilize these results to quantify noise in non-flip-around and flip-around MDACs.

We should caution the reader about two approximations in the foregoing analysis. First, the load capacitance (e.g., the input sampling capacitor of the next MDAC) is neglected. Second, the op amp is assumed to have no internal poles. In practice, neither of these is valid, requiring careful simulations.

Non-Flip-Around MDAC Figure 18.38(a) shows this MDAC in the sampling mode, with V_n denoting the input-referred noise of the op amp. We observe that the noise appears at X and is sampled on C_1 . This situation is similar to that in Fig. 18.37(g) and follows Eq. (18.51). Since Eqs. (18.51) and (18.53) differ only by a factor of $1 + C_1/C_2 = 3$, we can find the rms noise sampled on C_1 by simply dividing Eq. (18.56) by $(1 + C_1/C_2)^2$ and hence obtain $\overline{V_n^2} G_m/(4C_1)$. The corresponding charge on C_1 is given by $C_1 \sqrt{\overline{V_n^2} G_m/(4C_1)}$.

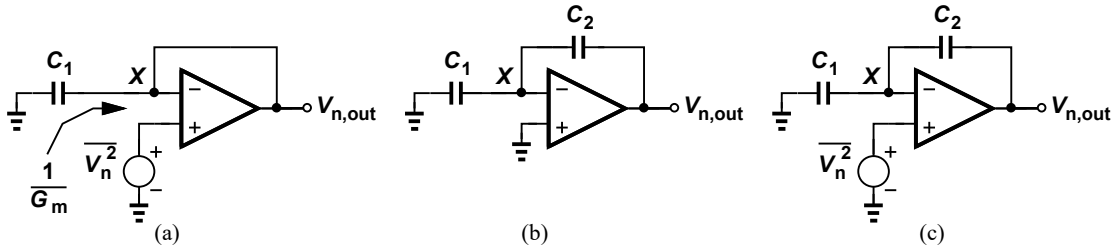


Figure 18.38 (a) Noise in sampling mode, (b) effect in amplification mode, and (c) overall result.

In the amplification mode [Fig. 18.38(b)], the noise stored on C_1 is transferred to C_2 because X behaves as a virtual ground. The output noise is therefore equal to $(C_1/C_2)V_{n,\text{rms}}$. To this value we must add the noise generated by the op amp in this mode [Fig. 18.38(c)]. From Eq. (18.56),

$$V_{n,\text{MDAC}}^2 = \overline{V_n^2} \frac{G_m}{4C_1} \cdot \frac{C_1^2}{C_2^2} + \left(1 + \frac{C_1}{C_2}\right)^2 \overline{V_n^2} \frac{G_m}{4C_1} \quad (18.58)$$

$$= \frac{13}{4} \frac{G_m}{C_1} \overline{V_n^2}. \quad (18.59)$$

The square root of this quantity gives the rms noise, which we select to be about 10% to 20% of the LSB size at the MDAC output.

The reader may wonder if the two terms in the above equation are correlated as they both contain $\overline{V_n^2}$. They are not because the first is frozen in the sampling mode while the second arises in the amplification mode. We are thus able to add the power spectral densities.

Flip-Around MDAC Consider this MDAC in the sampling mode [Fig. 18.39(a)]. We recognize that this arrangement is analogous to that in Fig. 18.13(a), where the op-amp offset is studied. We then conclude that the op-amp noise sampled on C_1 and C_2 in this mode (and hence “frozen”) is *canceled* when the circuit enters the amplification mode. This is another advantage of the flip-around MDAC.

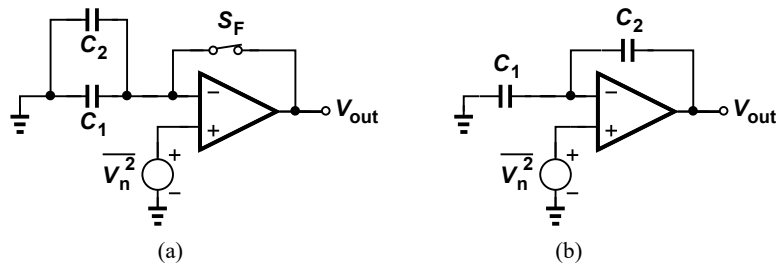


Figure 18.39 Noise in (a) sampling, and (b) amplification modes.

In the amplification mode [Fig. 18.39(b)], the topology resembles that in Fig. 18.37(f) and follows Eq. (18.56). But this value should be divided by $9/4 = 2.25$ because $C_1 = C_2$ here (rather than $C_1 = 2C_2$).

Problem of Capacitive Loading The foregoing noise calculations have assumed that the op-amp noise can be modeled by an input-referred spectrum, $\overline{V_n^2}$, as observed for the simple realization of Fig. 18.36. Two-stage op amps, on the other hand, pose additional difficulties due to their internal pole. When the op amp drives the input capacitor of another MDAC, its input-referred noise is different from when it does not.

An interesting point here is that the op-amp input-referred noise changes from the sampling mode to the amplification mode. Thus, Eqs. (18.56) and (18.58) are merely approximations and must be supplemented by detailed simulations.

18.7 Effect of Capacitor Mismatch

The non-flip-around and flip-around MDACs described previously assume a certain ratio for their capacitors so as to provide a residue of the form $2V_{in} \pm V_{REF}$. In reality, the capacitors incur random mismatches during fabrication, introducing gain error.

Let us first consider a non-flip-around topology in which $C_1 = 2C$ and $C_2 = C + \Delta C$, where ΔC denotes the mismatch [Fig. 18.40(a)]. If $V_{in} < 0$, node A swings to $-V_{REF}/2$, producing

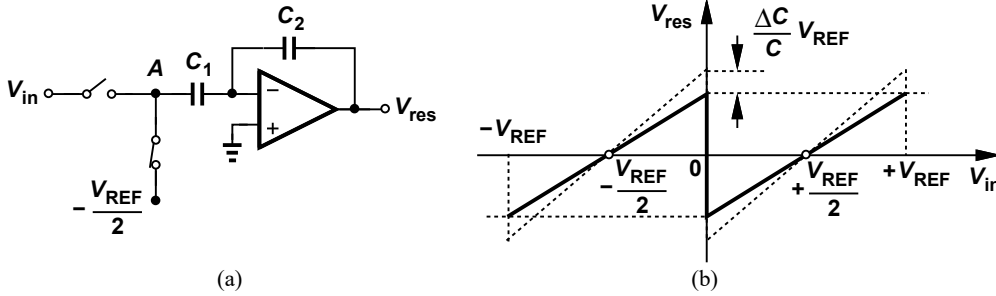


Figure 18.40 (a) MDAC, and (b) effect of capacitor mismatch on residue.

$$V_{res} = \left(-\frac{C_1}{C_2}\right) \left(-\frac{V_{REF}}{2} - V_{in}\right) \quad (18.60)$$

$$= \frac{1}{1 + \frac{\Delta C}{C}} (2V_{in} + V_{REF}). \quad (18.61)$$

If $|\Delta C/C| \ll 1$, we have

$$V_{res} = \left(1 - \frac{\Delta C}{C}\right) (2V_{in} + V_{REF}). \quad (18.62)$$

As V_{in} approaches zero, the residue incurs an error equal to $(\Delta C/C)V_{REF}$. As shown in Fig. 18.40(b), the residue plots are “pinned” at $V_{in} = \pm V_{REF}/2$ and rotate around these points for different $\Delta C/C$ values. The reader may recognize that this behavior is similar to that in Fig. 18.27 due to the op-amp gain error.

Example 18.10

Suppose the MDAC studied above serves as the first stage of an N -bit differential pipelined ADC. What is the maximum tolerable value of $\Delta C/C$ if the DNL must remain below 0.2 LSB? Assume $\Delta C/C < 0$ and the other stages are ideal.

Solution

We must first determine how the residue error translates to DNL. Since $\Delta C/C < 0$, Eq. (18.62) implies that V_{res} exceeds $+V_{REF}$ as V_{in} approaches zero from negative values [Fig. 18.41(a)]. Similarly, V_{res} displays an undershoot for input levels slightly greater than zero. The remaining ADC stages, on the other hand, can digitize only residue values confined to $[-V_{REF} + V_{REF}]$. The digitized residue is shown in Fig. 18.41(b), exhibiting saturation for $V_{in1} < V_{in} < V_{in2}$. The overall ADC thus generates $10 \cdots 0$ for $V_{in1} < V_{in} < V_{in2}$ [Fig. 18.41(c)], experiencing a maximum DNL equal to $V_{in2} - V_{in1}$. To obtain V_{in1} , we consider point A in Fig. 18.41(a) and write from (18.62)

$$\left(1 - \frac{\Delta C}{C}\right) (2V_{in1} + V_{REF}) \approx V_{REF}, \quad (18.63)$$

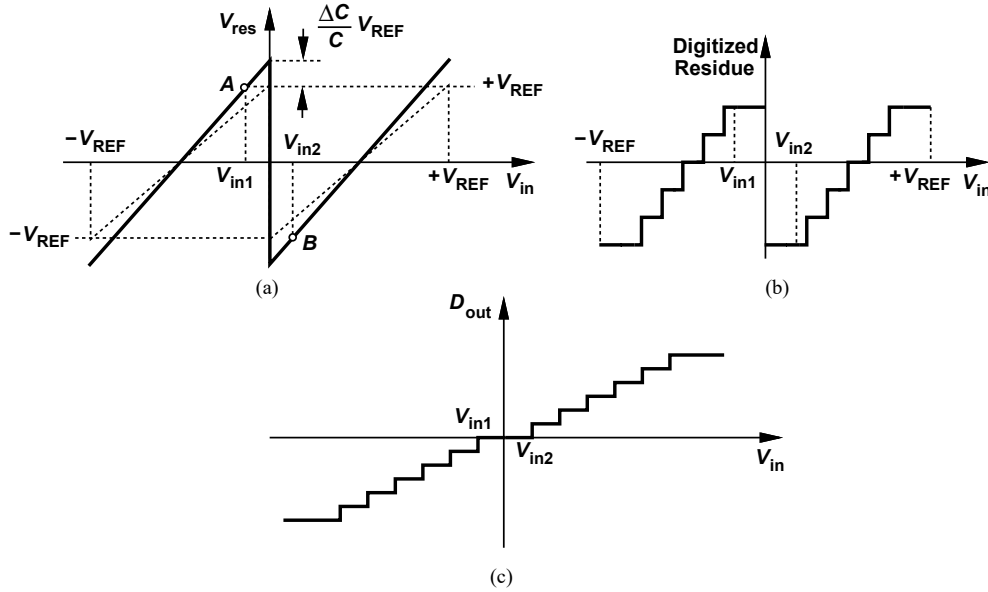


Figure 18.41 (a) Effect of capacitor mismatch, (b) quantized residue, and (c) effect on overall output.

and hence

$$V_{in1} \approx \frac{\Delta C}{2C} V_{REF}. \quad (18.64)$$

(Note that $\Delta C < 0$.) It follows that

$$|DNL_{max}| = 2 \times \frac{\Delta C}{2C} V_{REF}. \quad (18.65)$$

For this value to be less than $0.2 \text{ LSB} = 0.2(2V_{REF})/2^N$, we write $V_{REF} = 0.4 \text{ V}$ and

$$\left| \frac{\Delta C}{C} \right| < \frac{0.4}{2^N}. \quad (18.66)$$

For example, a 12-bit ADC requires a capacitor mismatch less than 0.01%, a stringent bound. Note that D_{out} also exhibits saturation near the end points.

The foregoing example deals with the case of $\Delta C/C < 0$. For $\Delta C/C > 0$, Fig. 18.40(b) and Eq. (18.62) suggest that the residue does *not* reach the extreme values of $\pm V_{REF}$. Upon combining the digitized residue [Fig. 18.42(a)] with the first-stage comparator output, we obtain the overall characteristic shown in Fig. 18.42(b), where missing codes occur at $V_{in} = -V_{REF}$, 0, and $+V_{REF}$.

We now repeat the mismatch analysis for the flip-around MDAC shown in Fig. 18.43(a), where the right plates of C_1 and C_2 are shown with thicker lines so that we can follow them more closely. We assume $C_1 = C$ and $C_2 = C + \Delta C$. The input is sampled on C_1 and C_2 after S_F turns off, leaving a total charge equal to $-(C_1 V_{in} + C_2 V_{in})$ on the right plates of C_1 and C_2 (why?). This charge is conserved as the circuit enters the amplification mode [Fig. 18.43(b)], *regardless* of the voltage on the left plate of C_1 . Assuming an ideal op amp, we allow this plate to swing from V_{in} to $-V_{REF}$ while X remains at the virtual ground. Thus, the charge on the right plate of C_1 is equal to $-C_1 V_{REF}$. We also know that the charge on the left (thicker) plate of C_2 is given by $-C_2 V_{res}$. These two charges must add up to the original amount,

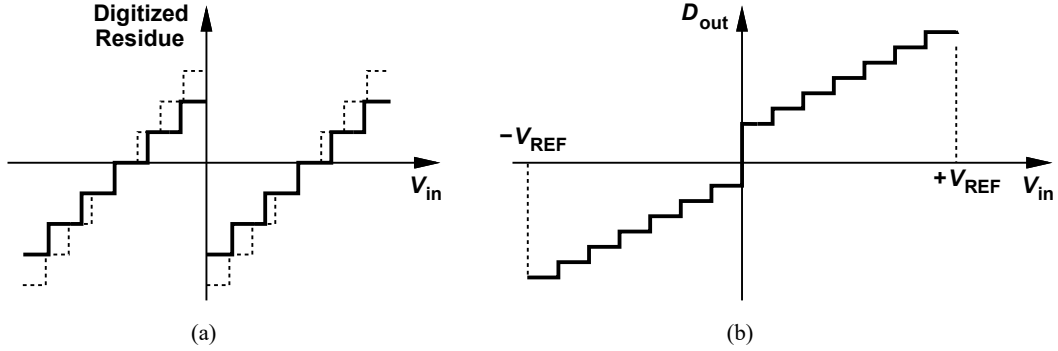


Figure 18.42 Effect of capacitor mismatch on (a) quantized residue, and (b) overall characteristic.

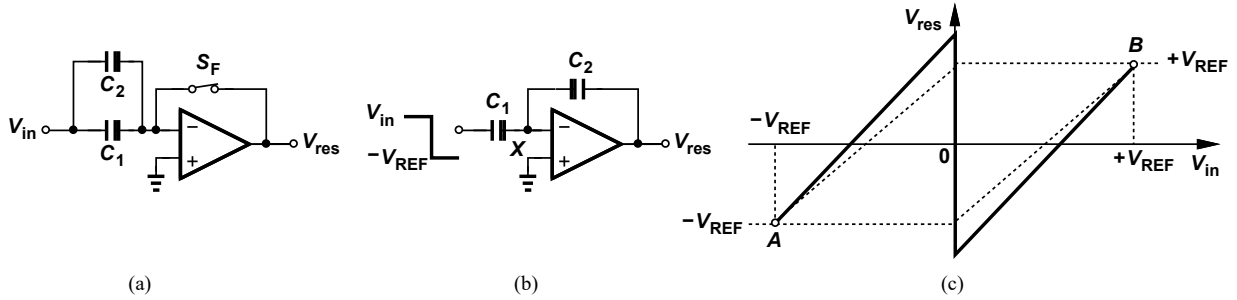


Figure 18.43 (a) Sampling mode, (b) amplification mode, and (c) effect of capacitor mismatch.

that is, $-C_1 V_{in} - C_2 V_{in} = -C_1 V_{REF} - C_2 V_{res}$. It follows that

$$V_{res} = \frac{C_1 + C_2}{C_2} V_{in} + \frac{C_1}{C_2} V_{REF} \quad (18.67)$$

$$= \frac{1}{1 + \frac{\Delta C}{C}} \left[\left(2 + \frac{\Delta C}{C} \right) V_{in} + V_{REF} \right]. \quad (18.68)$$

Interestingly, this result differs from that obtained for the non-flip-around topology [Eq. (18.62)]. Specifically, V_{res} does reach $-V_{REF}$ at $V_{in} = -V_{REF}$ or $+V_{REF}$ at $V_{in} = +V_{REF}$ [Fig. 18.43(c)]. We say the residue plots are pinned at points A and B and rotate around them for different values of $\Delta C/C$. As V_{in} approaches zero, (18.68) indicates that $V_{res} \approx V_{REF}/(1 + \Delta C/C) \approx (1 - \Delta C/C)V_{REF}$; i.e., the error is equal to $(\Delta C/C)V_{REF}$. If $\Delta C/C < 0$, the digitized residue behaves in the manner illustrated in Fig. 18.43(c). The capacitor mismatch budget is the same for the non-flip-around and flip-around MDACs.

18.8 Effect of the Reference Bounce

We have seen in Chapters 8 and 11 that capacitor DACs draw large transient currents from their reference(s), potentially suffering from a long settling time. A similar phenomenon arises in MDACs.

Shown in Fig. 18.44 is an MDAC entering the amplification and subtraction mode. The reference is modeled by KV_{REF} and R_{REF} . We note that the reference bounce negligibly affects the settling if R_{REF} is much less than the on-resistance of S_2 . As explained in Chapter 11, this condition translates to a high power consumption in the reference generator. Moreover, this generator's output impedance is typically *inductive*, causing long settling times. The analysis is similar to that in Chapter 15 for SAR ADCs.

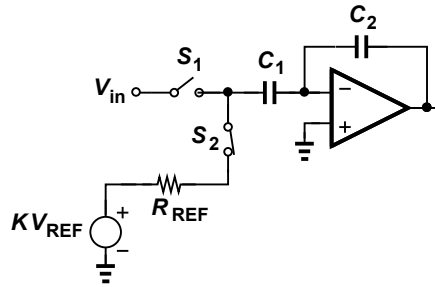


Figure 18.44 MDAC using a reference with a finite output resistance.

An interesting difference between SAR and pipelined ADCs is that, in a given cycle, the former connects only one capacitor DAC to the reference whereas the latter connects several. As illustrated in Fig. 18.45, every other stage in a pipeline operates in the amplification mode, drawing transient currents such as I_1 , I_3 , etc., from the reference. The bounce thus exhibits a more complex behavior and must be studied carefully.

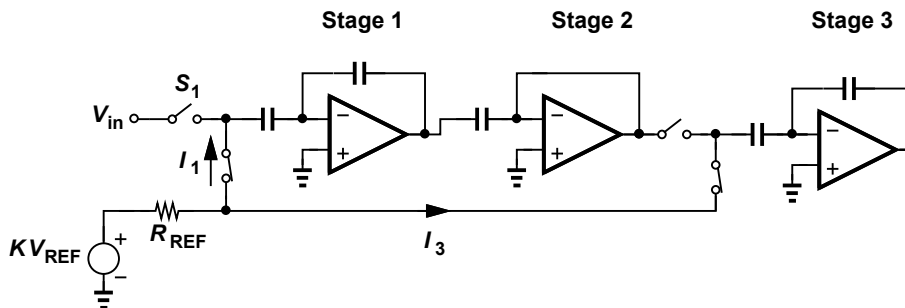


Figure 18.45 Pipelined cascade using a reference with a finite output resistance.

It is common to scale *down* the op amp and MDAC capacitors for stages deeper into the pipeline. Beyond our scope here, this method saves power but raises the noise of these stages. A byproduct of scaling is that the reference bounce is alleviated.

18.9 Timing Mismatch

In our study of MDACs, we have assumed that the comparator and the sampling capacitor(s) capture exactly the same point on the ADC input signal. As depicted in Fig. 18.46(a), when CK makes a transition, S_F turns off, sampling V_{in} on C_a , and the comparator is clocked, computing the sign of V_{in} . In reality, however, the signal paths through C_a and the comparator are quite different, exhibiting a timing mismatch. Consequently, the comparator decision may not agree with the voltage sampled on C_a [Fig. 18.46(b)].

Example 18.11

For an input of the form $V_{in} = V_0 \cos \omega_1 t$, how much timing mismatch can be tolerated in Fig. 18.46(a) if the error should not exceed 0.2 LSB?

Solution

The maximum input slew rate is equal to $V_0 \omega_1$. For a timing mismatch of Δt , the error voltage is given by $V_0 \omega_1 \Delta t$. We then have

$$\Delta t < \frac{0.2 \text{ LSB}}{V_0 \omega_1}. \quad (18.69)$$

Problems

- 18.1. Suppose the quantizer in Fig. 18.3(a) has a resolution of 2 bits.
 - (a) Plot $D_{out,q}$ versus V_a [as in Fig. 18.2(b)].
 - (b) Plot $D_{out,q}$ versus V_{in} .
- 18.2. Repeat the analysis depicted in Fig. 18.6 if the comparator offset, V_{OS} , is negative.
- 18.3. Suppose we increase the resolution of the quantizer in Fig. 18.7(a) to 3 bits. Construct the plots of Fig. 18.8 for this case. Next, construct the plots of Fig. 18.9.
- 18.4. If the offset of Comp_2 in Fig. 18.7(a) has an offset of -90 mV rather than $+90$ mV, construct the plots of Figs. 18.11 and 18.12.
- 18.5. Repeat the analysis depicted in Fig. 18.15 if the op-amp offset is -102 mV rather than $+102$ mV.
- 18.6. In the circuit of Fig. 18.23(a), how long does it take for V_{out} to reach within 0.1% of its final value? Assume $(1 + A_0)C_2 \gg C_1$.
- 18.7. Explain what happens to the time constant of the circuit shown in Fig. 18.24(c) if C_1 becomes very large.
- 18.8. Using the model depicted in Fig. 18.24(c), compare the time constants of non-flip-around and flip-around MDACs in the amplification mode.
- 18.9. Repeat the analysis depicted in Fig. 18.26 for a gain error of $+10\%$.
- 18.10. Repeat the analysis depicted in Fig. 18.27 for a gain error of $+10\%$.
- 18.11. Repeat the analysis shown in Fig. 18.27 if the quantizer has a resolution of 2 bits.
- 18.12. Repeat the analysis depicted in Fig. 18.30 if ϵ is positive.
- 18.13. Consider the two non-flip-around MDACs in a cascade, assuming that the first is ideal. If the op amp in the second stage has a gain of A_0 , use an analysis similar to that in Fig. 18.30 to determine the input-output characteristic. Assume the output of the second MDAC is digitized by an ideal 2-bit quantizer.
- 18.14. Express $\overline{V_n^2}$ in Fig. 18.36 in terms of the thermal noise of M_1 - M_4 . Does the noise of I_{SS} contribute to V_{out} and hence to $\overline{V_n^2}$? (Hint: with $V_{in} = 0$, the drain voltage of M_3 is equal to that of M_4 .)
- 18.15. Equation (18.56) suggests that the integrated noise voltage approaches infinity if $C_1 \rightarrow 0$. Explain this trend intuitively.
- 18.16. Explain why the noise in the sampling mode is canceled in a flip-around MDAC but not in the non-flip-around topology.
- 18.17. Repeat the analysis illustrated in Fig. 18.41 if V_{res} is digitized by a 2-bit quantizer.
- 18.18. Suppose the residue shown in Fig. 18.43(c) is digitized by a 2-bit quantizer. Plot the quantizer output versus V_{in} .
- 18.19. Determine the settling time constant of the circuit shown in Fig. 18.44 if the op amp is modeled by a simple Norton model. (Hint: recall that the impedance seen to the right of C_1 is equal to $1/G_m$.)
- 18.20. Determine the output noise spectrum in Fig. 18.44 due to the noise of R_{REF} .

Advanced SAR and Pipelining Techniques

Our study of SAR and pipelined ADCs in Chapters 14, 16, and 17 has laid the foundation for designing such systems. Extensive work on these architectures in the past decades has introduced numerous variants that push the performance envelope. Chief among these are the concepts of “overlap” and “redundancy,” 1.5-bit-per-stage pipelines, and hybrid SAR/pipelined architectures. This chapter deals with these principles. While studied in the context of pipelined and SAR converters, these methods apply to nearly all of today’s Nyquist-rate ADCs.

19.1 Overlap in Hybrid Flash/SAR ADCs

We have seen in Chapter 16 and Section 18.9 that the hybrid flash/SAR ADC incurs errors due to offsets in the flash stage as well as timing mismatches between this stage and the DAC sampling path. The use of overlap and redundancy can greatly alleviate these issues. We now study these concepts.

19.1.1 Comparator Offset Revisited

We concluded in Chapter 13 that if a comparator in the flash front end exhibits an offset of V_{OS} , then a DNL of the same value arises in the overall ADC characteristic. We revisit the offset issue here so as to understand how overlap relaxes it.

Consider the hybrid architecture shown in Fig. 19.1(a). Suppose the flash converter decides that $V_j < V_{in} < V_{j+1}$, i.e., Comp_j and Comp_{j+1} deliver high and low outputs, respectively. Then, ideally, V_{DAC} is set to V_j , and the SAR searches in the range of $[V_j \ V_{j+1}]$ for a more accurate representation of the input [Fig. 19.1(b)], digitizing the difference $V_{in1} - V_{DAC}$. The digital outputs of the flash and SAR ADCs are then properly combined as to yield the overall code. Note that the phrase “SAR search range” in Fig. 19.1(b) is somewhat inaccurate because the SAR ADC, in fact, receives an input ranging from *zero* to $V_{j+1} - V_j$ rather than from V_j to V_{j+1} . As in pipelined ADCs, we call $V_{in1} - V_{DAC}$ the “residue,” noting that it takes on a value between zero and $V_{j+1} - V_j$ as V_{in} varies from V_j to V_{j+1} .

In our study here, it is important to distinguish between the LSB of the flash stage, which is equal to $V_{j+1} - V_j$, and that of the SAR loop, which is equal to $(V_{j+1} - V_j)/2^m$, where m denotes the SAR resolution. The overall ADC LSB, called the “system LSB,” is equal to the SAR LSB.

Now, let us assume that the comparator sensing V_j in the flash stage has an offset of V_{OS} . The particular value of the ADC input voltage is denoted by V_{in1} . We study two cases: $V_{OS} > 0$ or $V_{OS} < 0$, recognizing that Comp_j generates a ONE only if $V_{in} > V_j + V_{OS}$. If $V_{OS} > 0$ and $V_j + V_{OS} > V_{in1}$ [Fig. 19.1(c)], then this comparator output does *not* change as V_{in} crosses V_j and even reaches a value of V_{in1} . The flash ADC therefore generates a digital output corresponding to V_{j-1} , setting V_{DAC} to the same value. The SAR loop attempts to digitize $V_{in1} - V_{DAC} = V_{in1} - V_{j-1}$, but this difference is *greater* than the SAR ADC’s full scale of $V_j - V_{j-1}$. Consequently, the SAR output “saturates,” failing to represent V_{in1} accurately.

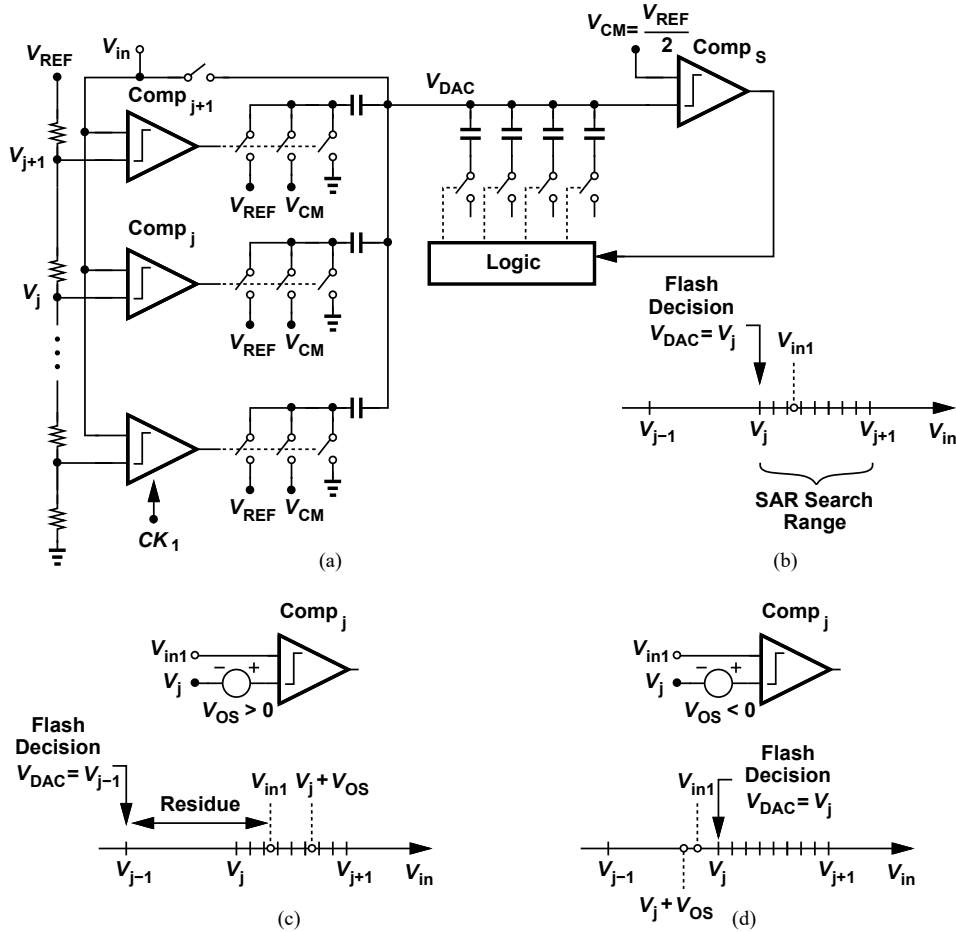


Figure 19.1 (a) Hybrid flash/SAR ADC, (b) ideal input range, (c) effect of positive comparator offset, and (d) effect of negative comparator offset.

If $V_{OS} < 0$ and V_{in1} is slightly less than V_j , the comparator may generate a high output even for $V_{in} < V_{in1}$ [because $V_{in1} - (V_j + V_{OS}) > 0$] [Fig. 19.1(d)], yielding $V_{DAC} = V_j$. Consequently, $V_{in1} - V_{DAC}$ is *negative*, causing the SAR loop to generate a zero output regardless of the value of V_{in1} .

Example 19.1

Does the foregoing study mean that no offset can be tolerated in the flash comparators?

Solution

No, it does not. A finite offset is acceptable if it leads to $DNL < 1$ LSB for the overall system. The reader can prove that $|V_{OS}| < 1$ LSB satisfies this condition.

19.1.2 Overlap and Redundancy

In order to accommodate positive and negative comparator offsets, we make two modifications to the hybrid architecture [Fig. 19.2(a)]: (a) if the flash stage in Fig. 19.1(a) selects V_j as the analog estimate (i.e., if Comp_j produces a high level but Comp_{j+1} does not), we set the DAC value to be halfway between V_{j-1} and V_j , and (b) we double the full scale of the SAR loop and increase its resolution by 1 bit. Let us see how the resulting

design operates with $V_{OS} > 0$ or $V_{OS} < 0$. Shown in Fig. 19.2(b) is the case of $V_{OS} > 0$, where the flash front end believes that $V_{in1} < V_j$ and recommends V_{j-1} as the DAC's initial value even though $V_{in1} > V_j$. In this case, V_{DAC} is set to $(V_{j-2} + V_{j-1})/2$, leading to a residue equal to $V_{in1} - (V_{j-2} + V_{j-1})/2$. This value is greater than $V_j - V_{j-1}$ but is digitized properly because the SAR search range has been doubled. For $V_{OS} < 0$ and $V_{in2} < V_j$, Fig. 19.2(c) reveals that V_{DAC} is equal to $(V_j + V_{j-1})/2$, avoiding a *negative* residue, and the SAR stage successfully digitizes $V_{in2} - V_{DAC}$.

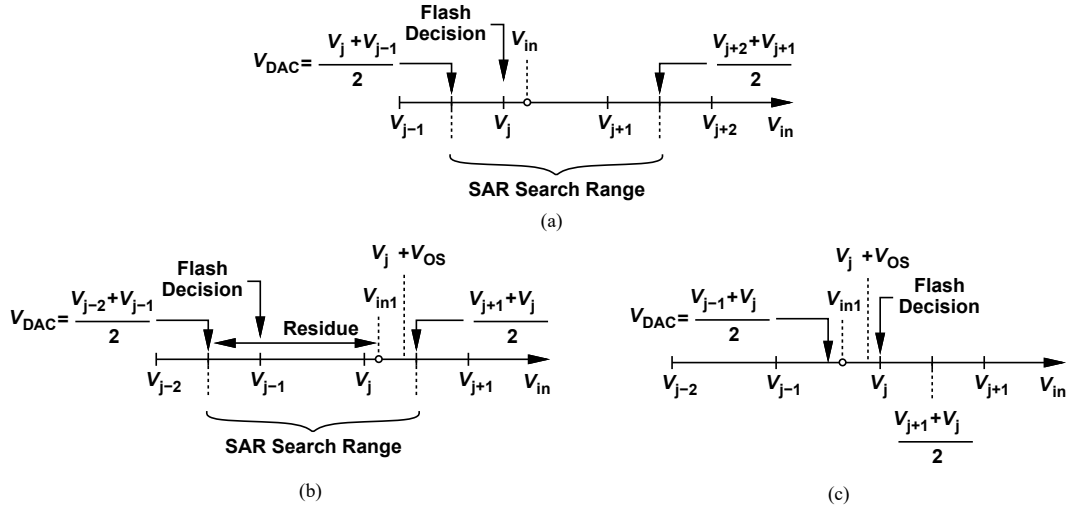


Figure 19.2 Overlap in flash/SAR ADC: (a) ideal case, (b) with V_{in} slightly greater than V_j and $V_{OS} > 0$, and (c) with V_{in} slightly less than V_j and $V_{OS} < 0$.

We say the modified architecture employs “overlap” because the flash and SAR search ranges have an overlap equal to $(V_{j-1} - V_{j-2})/2$ in Fig. 19.2(a). We also say the ADC incorporates 1 bit of “redundancy” because the SAR loop’s resolution has been raised by 1 bit while the overall ADC resolution has not changed.

Example 19.2

For an overall resolution of 6 bits, the flash and SAR stages in Fig. 19.1(a) have resolutions equal to 3 bits and 4 bits, respectively. Assume no offsets and $V_{REF} = 640$ mV. How do we combine their digital outputs? What is the overall resolution?

Solution

For a full-scale range of zero to V_{REF} , the flash LSB is equal to $V_{REF}/8 = V_j - V_{j-1} = 80$ mV. The SAR LSB is equal to 10 mV (why?). Suppose $V_{in1} = V_{REF}/2 + 1.6$ LSB, where LSB refers to that of the overall system (and is equal to that of the SAR). With $V_{OS} = 0$, the flash ADC produces a binary output equal to $D_F = 100$ ($\equiv 4$) but the DAC value that it delivers corresponds to $V_{REF}/2 - V_{REF}/16 = 280$ mV (Fig. 19.3). The SAR search range extends from 280 mV to $V_{REF}/2 + V_{REF}/8 + V_{REF}/16 = 440$ mV in 16 steps. It follows that $V_{in1} - V_{DAC} = V_{REF}/16 + 1.6$ LSB = 5.6 LSB, leading to a binary value of $D_S = 0101$ for the SAR output.

To combine D_F and D_S properly, we recognize that 1 LSB of the flash stage is equal to 8 LSB of the SAR loop. Thus, $D_F = 100$ must be multiplied by 8 before it is added to the SAR result. In the binary domain, this is accomplished by shifting the bits of D_F to the left three times. We must also subtract a value equivalent to $V_{REF}/16 \equiv 100$ from D_F so as to represent the actual quantity given to the DAC. (Recall that $V_{DAC} = 280$ mV rather than 320 mV.) We therefore write

$$\begin{array}{r}
 100000 \\
 - 000100 \\
 + \quad 0101 \\
 \hline
 100001
 \end{array}$$

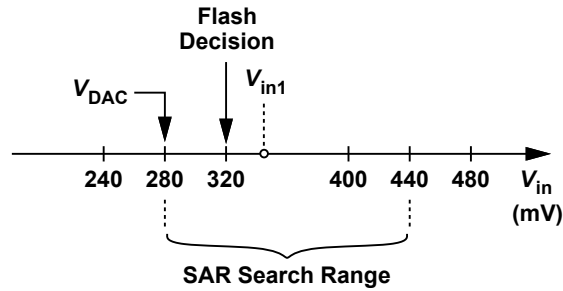


Figure 19.3 Numerical example of overlap in a flash/SAR ADC.

As expected, the overall output is equal to 33 LSB and the system's resolution is 6 bits.

We must now determine how much comparator offset can be tolerated in the presence of overlap. We saw in Fig. 19.2(a) that a positive V_{OS} can change the flash output from V_j to V_{j-1} , and we decided to provide a value of $(V_{j-1} + V_{j-2})/2$ by the DAC. Suppose, in Fig. 19.2(b), V_{OS} becomes larger, pushing $V_j + V_{OS}$ toward $(V_j + V_{j+1})/2$. We also assume that V_{in1} is only slightly less than $V_j + V_{OS}$ [Fig. 19.4(a)]. The residue thus increases but remains within the SAR range unless $V_j + V_{OS}$ exceeds $(V_j + V_{j+1})/2$. That is,

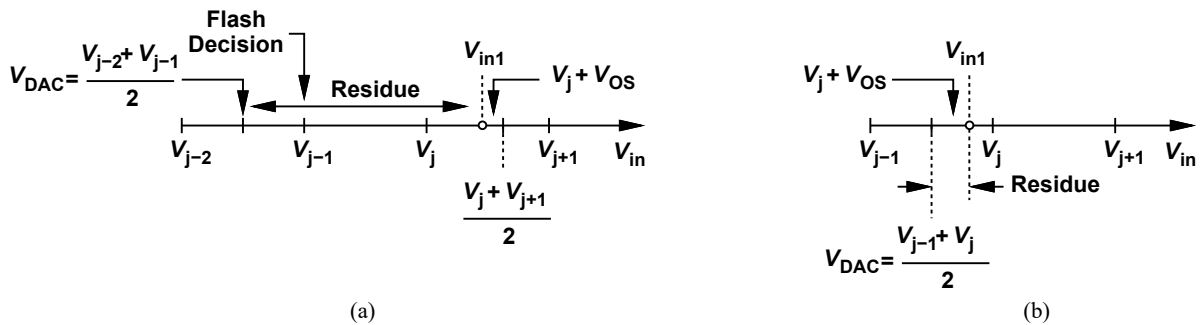


Figure 19.4 Offset tolerance in the presence of overlap.

the system can tolerate offsets as large as one half of the flash LSB, a considerable improvement compared to the case of no overlap.

Similarly, as depicted in Fig. 19.4(b), V_{OS} can be as negative as 0.5 flash LSB before the residue becomes negative.

Example 19.3

In the hybrid ADC of Example 19.2, how should V_{DAC} be set if the flash converter decides that $0 < V_{in} < V_{REF}/8 = 80 \text{ mV}$ (Fig. 19.5)?

Solution

In this case, V_{DAC} is simply set to zero, and the SAR search range extends from zero to $2V_{REF}/8$. The reader can prove that the offset of Comp_1 can still be nearly as large as $V_{REF}/8$ without introducing any error.

The foregoing study has assumed 1 bit of overlap and redundancy. One can choose greater values to accommodate larger offsets and vice versa.

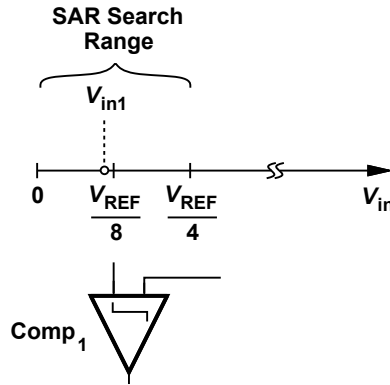


Figure 19.5 Choice of V_{DAC} for $0 < V_{in} < V_{REF}/8$.

19.1.3 Effect of Timing Mismatch

The timing mismatch issue described in Section 18.9 for pipelined ADCs also applies to hybrid flash/SAR architectures. As depicted in Fig. 19.6(a), the flash comparators and the DAC present different sampling

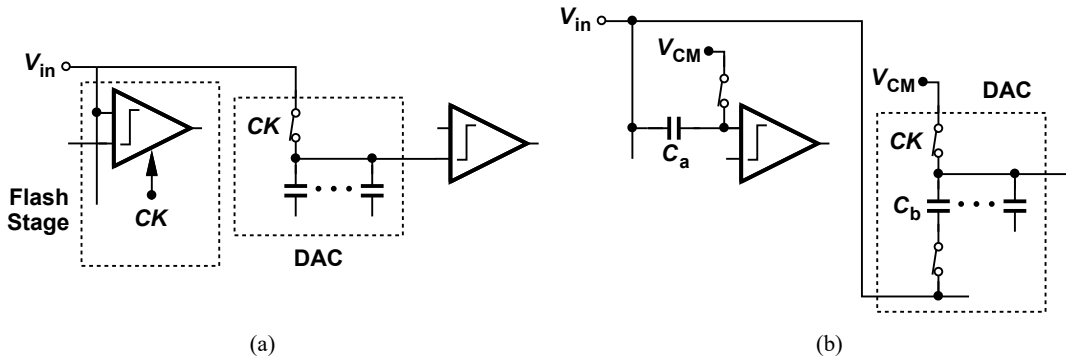


Figure 19.6 (a) Problem of timing mismatch, and (b) nominally identical signal paths for reducing the mismatch.

paths to V_{in} , suffering from a systematic delay mismatch, Δt . Our analysis in Section 18.9 reveals that the maximum discrepancy between the values sampled by the flash and DAC circuits is equal to $V_0 \omega_1 \Delta t$ for $V_{in} = V_0 \cos \omega_1 t$. We can say that, equivalently, the voltage sampled on the DAC is “correct” but the flash comparator making a critical decision has a maximum offset of $V_0 \omega_1 \Delta t$.

It is possible to reduce the delay mismatch by selecting a comparator topology whose sampling path mimics that of the DAC. Shown in Fig. 19.6(b) is an example where C_a and C_b sample V_{in} for the comparator and the DAC, respectively. We strive to maintain the same time constant for the two paths by scaling the widths of the switches in proportion to the capacitor values. For example, if $C_b = 2C_a$, then the switches in series with C_b should be twice as wide as those in series with C_a .

Overlap and redundancy can accommodate this error as well. The overall ADC output is correct if the flash comparator static offset, V_{OS} , and the timing-induced error add up to less than 0.5 LSB of the flash stage.

19.1.4 Residue in Hybrid Flash/SAR ADCs

As explained in Section 19.1.1, the difference between the analog input, V_{in} , and the initial DAC voltage prescribed by the flash front end can be considered a residue quantity. We now extend our knowledge of the residue in pipelined ADCs (Chapters 17 and 18) to this architecture. In this section, the terms “LSB” or “flash LSB” refer to the LSB of the flash stage.

Let us assume an ideal system with no offsets and plot the residue for two cases, namely, without overlap or with. In the former case, the flash stage sets V_{DAC} to V_j if $V_j < V_{in1} < V_{j+1}$ [Fig. 19.7(a)], yielding

$V_{res} = V_{in1} - V_j$. The residue thus behaves as shown in Fig. 19.7(b). With 1 bit of overlap, on the other hand, $V_{DAC} = (V_j + V_{j-1})/2$ [Fig. 19.7(c)], and V_{res} varies from 0.5 LSB to 1.5 LSB [Fig. 19.7(d)]. Note that the input range of the SAR loop extends from zero to 2 LSB, but only half of it is exercised in this ideal case.

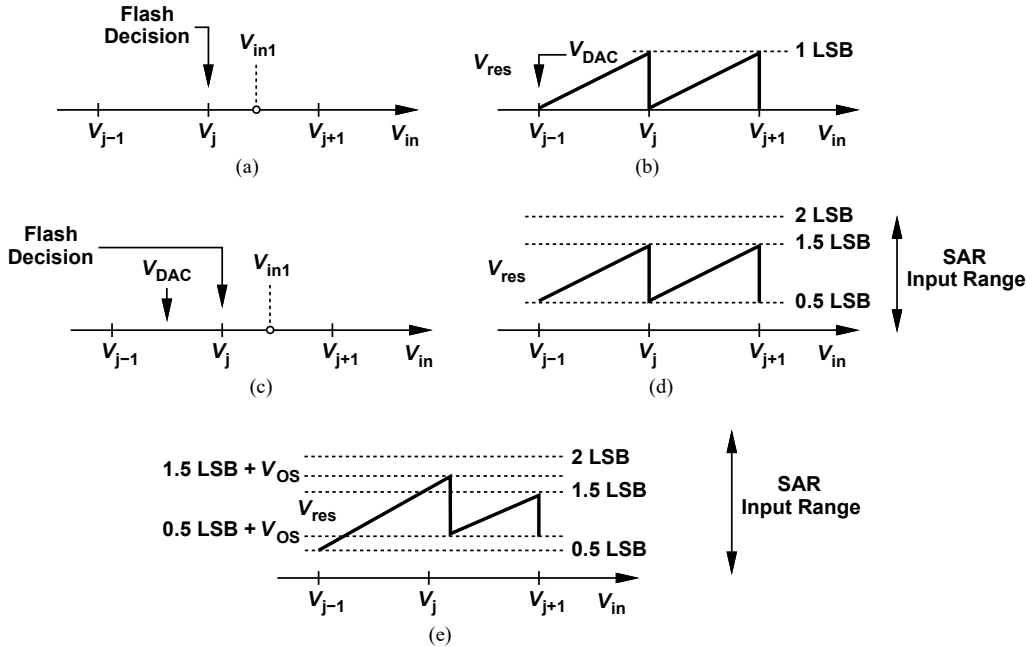


Figure 19.7 (a) Input range example, (b) residue plot, (c) input range with overlap, (d) behavior of residue with overlap, (e) effect of comparator offset.

We now assume that the flash comparator sensing V_j exhibits an offset, V_{OS} . The flash decision threshold at V_j therefore shifts by V_{OS} . Suppose the shift is to the right [Fig. 19.7(e)], then V_{res} exceeds 1.5 LSB as V_{in} crosses V_j , an effect accommodated by the SAR ADC if $|V_{OS}| < 0.5$ LSB. For V_{in} slightly greater than $V_j + V_{OS}$, $V_{res} = V_{in} - (V_j + V_{j-1})/2 \approx V_{OS} + 0.5$ LSB.

19.1.5 Effect of SAR Offset

The comparator in the hybrid architecture's SAR loop can suffer from an offset, V_{OS} . In this section, the term "LSB" refers to that of the SAR and the system. How does V_{OS} manifest itself in the overall characteristic? Assuming an ideal flash stage and no overlap, we consider the scenario shown in Fig. 19.8(a), noting that V_{DAC} is set to V_j and the SAR ADC must digitize $V_{in1} - V_{DAC}$. However, due to its own offset, this stage interprets the residue as $V_{in1} - V_{DAC} + V_{OS}$, which can exceed the SAR full scale ($= V_{j+1} - V_j$) if $V_{OS} > 0$, causing saturation in its output code [Fig. 19.8(b)]. That is, the SAR ADC delivers 11...1 for $V_{j+1} - V_{OS} < V_{in1} < V_{j+1}$. This result allows us to plot the SAR output as a function of the main input [Fig. 19.8(c)]. In a manner similar to that in Chapter 16, we now add the SAR code and the flash output [Fig. 19.8(d)], arriving at the overall characteristic shown in Fig. 19.8(e). The SAR comparator offset thus directly translates to DNL and INL.

Do overlap and redundancy alleviate this phenomenon? Returning to Fig. 19.8(a) and assuming that the flash stage is ideal and setting V_{DAC} to $(V_j + V_{j-1})/2$, we observe that the "equivalent" residue, V_{res} , is affected by V_{OS} . Specifically, $V_{res} \approx (V_j + V_{j-1})/2 + V_{OS}$ if V_{in1} is slightly greater than V_j (Fig. 19.9), and $V_{res} \approx V_{j+1} - (V_j + V_{j-1})/2 + V_{OS}$ if V_{in1} is slightly less than V_{j+1} . This V_{res} range is readily accommodated by the SAR loop if we have 1 bit of overlap and V_{OS} is not excessively large. That is, the SAR output does not saturate. Unfortunately, however, the overall ADC still incurs an error. The following example elaborates on this point.

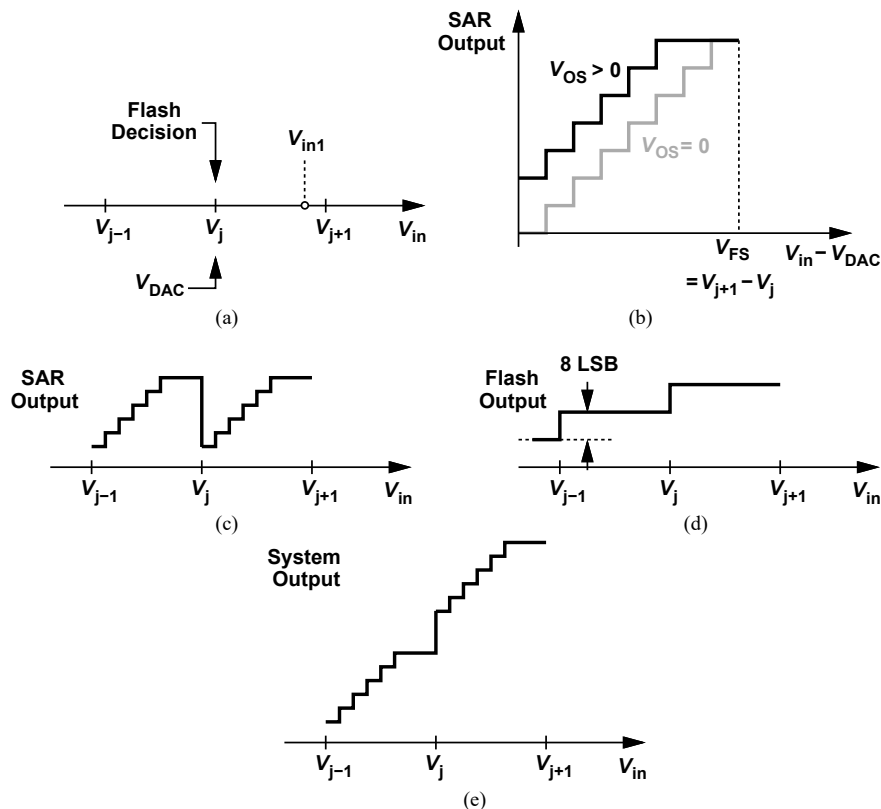


Figure 19.8 (a) Input range example, (b) effect of SAR comparator offset, (c) resulting SAR output, (d) flash output, and (e) overall characteristic.

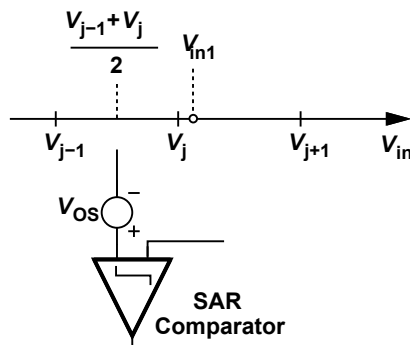


Figure 19.9 Effect of SAR comparator offset.

Example 19.4

Consider the hybrid ADC described in Example 19.2, where the system LSB is given by $640 \text{ mV}/64 = 10 \text{ mV}$. Suppose $V_{in1} = V_{REF}/2 + 1.6 \text{ LSB} = 336 \text{ mV}$. If the flash stage is ideal and the SAR comparator suffers from an offset of $-2.2 \text{ LSB} = -22 \text{ mV}$, determine the resulting error in the overall output.

Solution

The flash stage sets V_{DAC} to $V_{REF}/2 - V_{REF}/16 = 320 \text{ mV} - 40 \text{ mV} = 280 \text{ mV}$, ideally generating

a residue equal to $336 \text{ mV} - 280 \text{ mV} = 56 \text{ mV}$ (Fig. 19.10). Owing to its offset, however, the SAR loop subtracts 22 mV from this value, yielding a code of 0011 to represent $34 \text{ mV} \equiv 3.4 \text{ LSB}$. Next, we write the code corresponding to $V_{DAC} = 280 \text{ mV} \equiv 28 \text{ LSB}$ as 011100. Upon adding this value to the SAR output, we have $011100 + 000011 = 011111 \equiv 31 \text{ LSB}$. The result is therefore 2 LSB less than the ideal value.

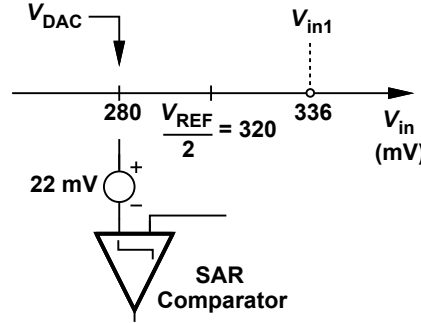


Figure 19.10 Numerical example for SAR offset.

In summary, the hybrid flash/SAR ADC cannot correct for the SAR offset by overlap and redundancy. This is fundamentally because the SAR offset shifts the residue and hence the SAR output while the flash stage has no knowledge of it. This offset must be maintained well below 1 LSB, often requiring comparator offset cancelation (Chapter 6).

19.2 1.5-Bit/Stage Pipelined ADCs

Our study of pipelined ADC imperfections in Chapter 18 suggests that the MDAC comparator offset directly translates to DNL and INL. We also noted in Section 19.1.2 that overlap and redundancy accommodate comparator offsets in hybrid flash/SAR architectures. We then wonder whether pipelined converters can benefit from the same technique.

19.2.1 Basic Idea

Consider the MDAC shown in Fig. 19.11(a) and recall from Chapter 17 that the residue, V_{res} , is equal to $2V_{in} + V_{REF}$ if $V_{in} < 0$ and $2V_{in} - V_{REF}$ if $V_{in} > 0$. With comparator offset, on the other hand, V_{res} incurs an overrange or underrange [Fig. 19.11(b)]. The excessively positive or negative residue causes saturation in the subsequent MDACs. How can we avoid this phenomenon? We surmise that the circuit must be so modified as to ensure V_{res} is confined to an acceptable range in the vicinity of the decision threshold(s).

Let us apply redundancy and employ *two* comparators in the MDAC [Fig. 19.11(c)], one sensing $-V_{REF}/4$ and the other, $+V_{REF}/4$. We wish the comparators to discern three distinct ranges for the input and generate the residue as follows:

$$V_{res} = 2V_{in} + V_{REF} \text{ if } -V_{REF} < V_{in} < -\frac{V_{REF}}{4} \quad (19.1)$$

$$= 2V_{in} \text{ if } -\frac{V_{REF}}{4} < V_{in} < +\frac{V_{REF}}{4} \quad (19.2)$$

$$= 2V_{in} - V_{REF} \text{ if } +\frac{V_{REF}}{4} < V_{in} < +V_{REF}. \quad (19.3)$$

Depicted in Fig. 19.11(d) for the ideal case, the MDAC's behavior can be articulated as follows. If $|V_{in}| > V_{REF}/4$, we amplify it by a factor of 2 and add or subtract V_{REF} (as we did in the 1-bit MDAC). On

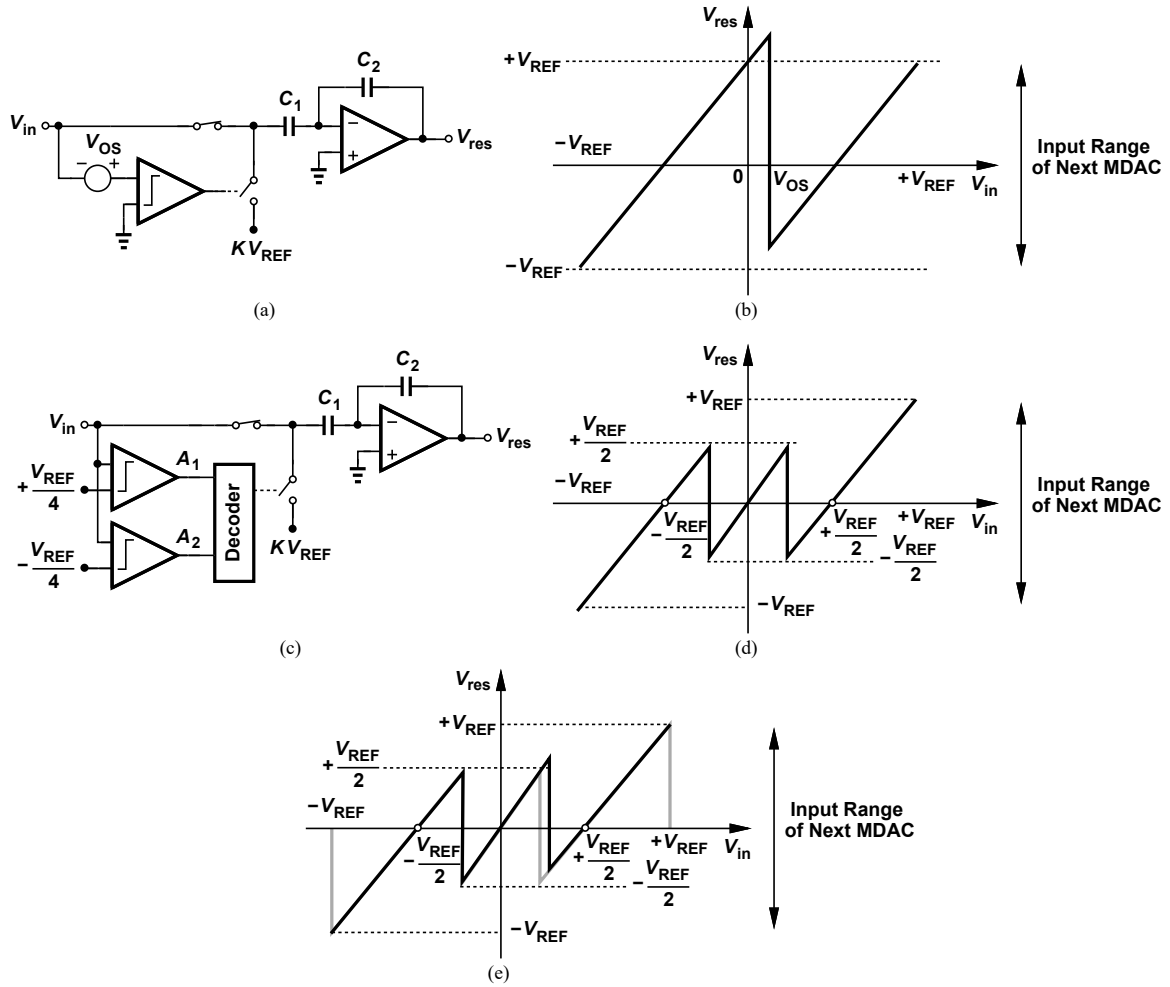


Figure 19.11 (a) MDAC circuit, (b) effect of comparator offset on residue, (c) 1.5-bit-per-stage MDAC, (d) resulting residue behavior, and (e) effect of comparator offset.

the other hand, if $|V_{in}| < V_{REF}/4$, we simply amplify it but do not add or subtract V_{REF} . The key distinction here is that V_{res} is now limited to $\pm V_{REF}/2 + 2V_{OS}$ near the comparator decision thresholds [Fig. 19.11(e)], whereas it exceeded $+V_{REF}$ in Fig. 19.11(b). Consequently, our new scheme can accommodate offsets as high as $V_{REF}/4$ before it experiences overrange or underrange (why?).

The topology shown in Fig. 19.11(c) is called a “1.5-bit” MDAC as it incorporates two comparators and provides three decision ranges.¹ The 0.5-bit redundancy dramatically relaxes the tolerable comparator offset. The cost includes one extra comparator and a finite delay associated with decoding the comparators’ thermometric output, both acceptable. This type of MDAC is thus much more popular than the 1-bit counterpart. Note that the comparator outputs form a thermometer code, A_2A_1 , going from 00 to 10 and 11 as V_{in} varies from $-V_{REF}$ to $+V_{REF}$.

An interesting question here is, how do we extract 1 bit of information from A_2A_1 ? We return to this point later as we combine the logical outputs of cascaded MDACs. But we should remark that the *last* stage in such a pipelined ADC employs only one comparator (and no residue amplification). That is, this last stage provides a 1-bit output.

¹Note that $2^{1.5} \approx 3$.

Example 19.5

How should the decoder in Fig. 19.11(c) be designed?

Solution

The thermometric output assumes values given by $A_2A_1 = 00, 10$, and 11 as V_{in} varies from $-V_{REF}$ to $+V_{REF}$. Shown in Fig. 19.12 for a flip-around MDAC, the decoder switches the left plate of C_1 to $-V_{REF}$, zero, or $+V_{REF}$ accordingly.

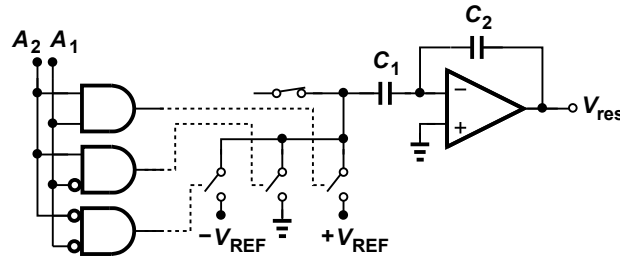


Figure 19.12 Logic gates for driving switches.

19.2.2 Residue Behavior

It is instructive to construct the residue plots for two cascaded pipelined 1.5-bit MDACs. Beginning with the first residue, V_{res1} , depicted in Fig. 19.13(a), we observe that, if the MDAC input varies from $-V_{REF}$ to zero, the residue traverses the plot in the left half plane. We call this part of the characteristic “Section A” and draw it separately as shown in Fig. 19.13(b). Similarly, for $0 < V_{in} < +V_{REF}/2$, we obtain Section B, and for $-V_{REF}/2 < V_{in} < 0$, Section C. The reason for defining these sections becomes clear below. Note that these plots apply to any 1.5-bit MDAC in a pipeline.

From Eqs. (19.1)-(19.3), the output of the second MDAC, V_{res2} , is given by

$$V_{res2} = 2V_{res1} + V_{REF} \quad \text{if} \quad -V_{REF} < V_{res1} < -\frac{V_{REF}}{4} \quad (19.4)$$

$$= 2V_{res1} \quad \text{if} \quad -\frac{V_{REF}}{4} < V_{res1} < +\frac{V_{REF}}{4} \quad (19.5)$$

$$= 2V_{res1} - V_{REF} \quad \text{if} \quad +\frac{V_{REF}}{4} < V_{res1} < +V_{REF}. \quad (19.6)$$

To plot V_{res2} versus the main input, we must map V_{in} through V_{res1} according to Eqs. (19.1)-(19.3) and then through Eqs. (19.4)-(19.6). The procedure proves more intuitive if done graphically. Suppose $-V_{REF} < V_{in} < -V_{REF}/2$. Then, V_{res1} varies from $-V_{REF}$ to zero. The second stage interprets V_{res1} as its input and delivers its own residue according to Section A in Fig. 19.13(b) (why?). Illustrated in Fig. 19.14(a), this result exhibits a slope of 4, as expected of two cascaded MDACs each having a gain of 2.

We now consider the range $-V_{REF}/2 < V_{in} < -V_{REF}/4$, observing from Fig. 19.13(a) that V_{res1} correspondingly goes from zero to $+V_{REF}/2$. Applied to the second MDAC, this output invokes Section B from Fig. 19.13(b). The result is depicted in Fig. 19.14(b). Similarly, as V_{in} increases from $-V_{REF}/4$ to zero, V_{res1} varies from $-V_{REF}/2$ to zero, delivering Section C to V_{res2} [Fig. 19.14(c)]. We then “stitch” the three V_{res2} plots and use symmetry to extend our findings to the right half plane. The overall behavior is shown in Fig. 19.14(d).

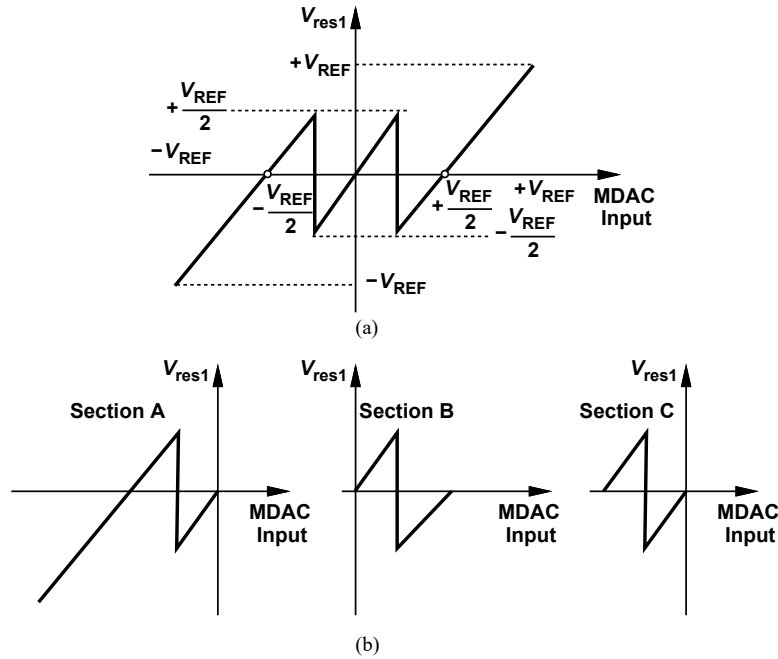


Figure 19.13 Residue behavior and its individual sections.

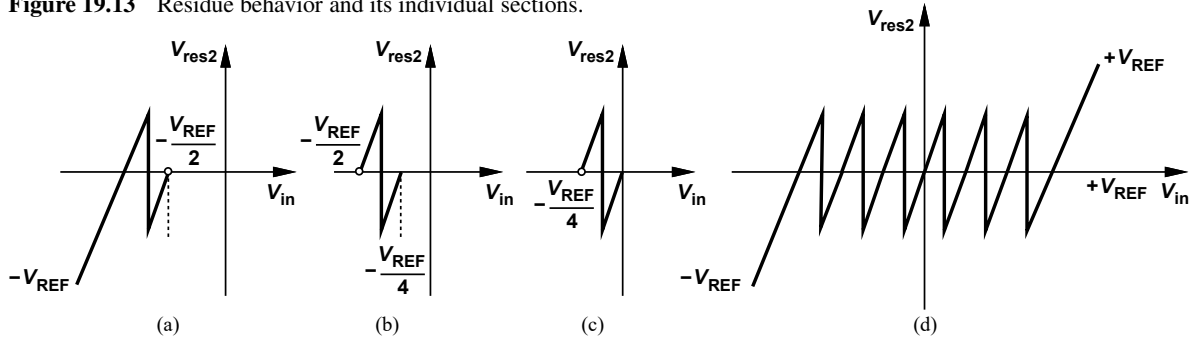


Figure 19.14 Different sections of second residue characteristic.

19.2.3 Cascade Example

In order to develop a deeper understanding of the 1.5-bit-per-stage architecture, let us design a 3-bit pipelined ADC that incorporates 1.5-bit MDACs.² We are particularly interested in the comparator outputs and wish to decode and combine them so as to obtain the overall ADC output. Recall from Example 19.5 that the thermometer code is given by $A_2A_1 = 00, 10, 11$. We convert A_2A_1 to binary values of 00, 01, and 10, respectively (Fig. 19.15). Denoting the binary output of stage j by $D_{2,j}D_{1,j}$, for $j = 1$ and 2, we combine the results as follows:

$$\begin{array}{r} D_{2,1} D_{1,1} \\ + \quad D_{2,2} D_{1,2} \\ + \quad D_{1,3} \\ \hline \end{array}$$

where $D_{1,3}$ is the third-stage comparator output.³ The sum serves as the overall ADC output, as illustrated by the following example.

²The third stage contains only one comparator and no residue amplification.

³A right shift by 1 bit represents division by 2. That is, $D_{2,2}D_{1,2}$ is shifted so that its equivalent value is divided by 2.

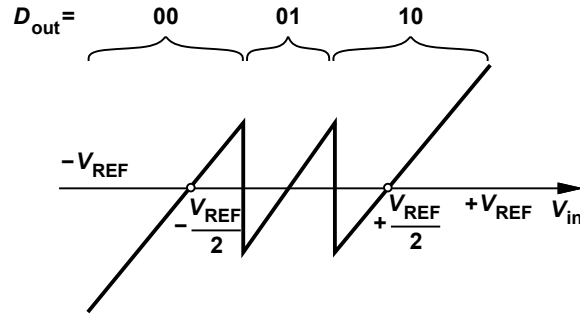


Figure 19.15 Assignment of digital output codes to different residue sections.

Example 19.6

The 3-bit ADC described above senses an input equal to $V_{in1} = +1.2$ LSB, $V_{in2} = -1.2$ LSB, or $V_{in3} = +3.2$ LSB. Determine the digital output in each case. Assume the digital output is equal to zero for $-4 \text{ LSB} < V_{in} < -3 \text{ LSB}$.

Solution

From Fig. 19.16(a), we note that $V_{in1} = +1.2 \text{ LSB} > +V_{REF}/4 = +1 \text{ LSB}$, obtaining $D_{2,1}D_{1,1} = 10$. Also, $V_{res1} = 2V_{in1} - V_{REF} = -1.6 \text{ LSB}$. The second stage decides that $V_{res1} < -V_{REF}/4$, generating $D_{2,2}D_{1,2} = 00$ and $V_{res2} = 2V_{res1} + V_{REF} = +0.8 \text{ LSB}$. It follows that the third stage produces $D_{1,3} = 1$. We add the results as

$$\begin{array}{r} 10 \\ + 00 \\ + 1 \\ \hline 101 \end{array}$$

As expected, the digital output is equal to 5.

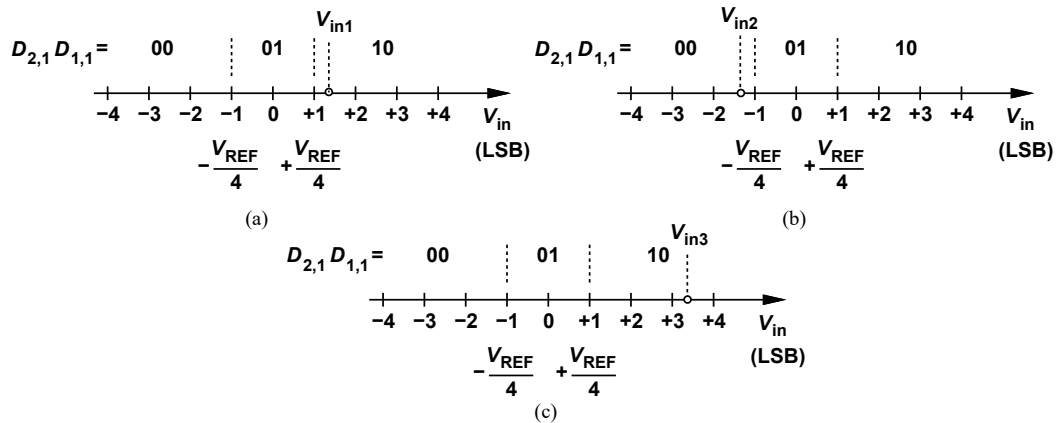


Figure 19.16 Examples of digital codes for different input values.

For the case of $V_{in2} = -1.2 \text{ LSB} < -V_{REF}/4$ [Fig. 19.16(b)], we have $D_{2,1}D_{1,1} = 00$ and $V_{res1} = 2V_{in2} + V_{REF} = +1.6 \text{ LSB}$. Sensing this value, the second stage produces $D_{2,2}D_{1,2} = 10$ and $V_{res2} = 2V_{res1} - V_{REF} = -0.8 \text{ LSB}$. The third stage thus yields $D_{1,3} = 0$.

The overall output is expressed as

$$\begin{array}{r} 00 \\ + 10 \\ + 0 \\ \hline 010 \end{array}$$

For $V_{in3} = +3.2 \text{ LSB} > +V_{REF}/4$ [Fig. 19.16(c)], the first MDAC gives $D_{2,1}D_{1,1} = 10$ and $V_{res1} = 2V_{in3} - V_{REF} = +2.4 \text{ LSB}$. Thus, the second MDAC generates $D_{2,2}D_{1,2} = 10$ and $V_{res2} = 2V_{res1} - V_{REF} = +0.8 \text{ LSB}$. It follows that $D_{1,3} = 1$ and hence

$$\begin{array}{r} 10 \\ + 10 \\ + 1 \\ \hline 111 \end{array}$$

The reader is encouraged to repeat this example while assuming a finite offset for different comparators.

Let us return to the pipelined ADC imperfections in Chapter 18 and determine which ones can be accommodated in the 1.5-bit architecture. We observe that the timing mismatch between the MDAC sampling path and the comparator input path lends itself to correction here because it is similar to comparator offset. However, gain errors in the MDAC output due to the finite op amp gain and capacitor mismatches are *not* corrected. In other words, only vertical shifts in the residue plot can be tolerated in this architecture.

19.2.4 Omission of Resistor Ladder

The two comparators in Fig. 19.11(c) require reference voltages equal to $\pm V_{REF}/4$, naturally calling for a resistive divider. One can avoid the static power thus consumed by employing *capacitive* dividers instead. Recall from Chapter 6 that fully-differential comparators can incorporate a capacitive network to sample the input and subtract it from the reference. Figure 19.17(a) depicts the idea.⁴ We now decompose C_1 into two

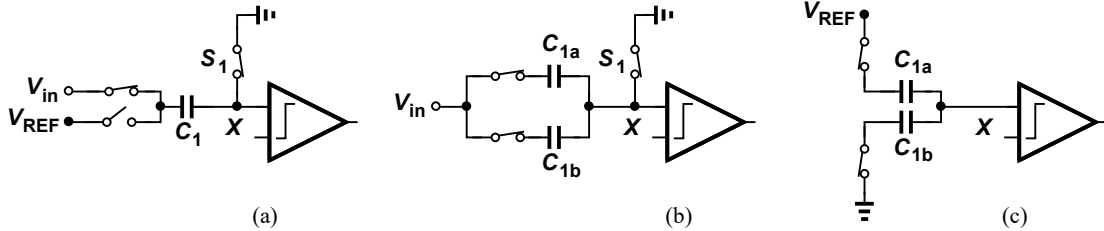


Figure 19.17 (a) Comparator with sampling capacitor, (b) decomposition of C_1 , and (c) switching of capacitors' left plates to V_{REF} and zero.

and sample V_{in} on both [Fig. 19.17(b)]. After switch S_1 turns off, the total charge on the right plates of C_{1a} and C_{1b} is given by $-(C_{1a} + C_{1b})V_{in}$ and is conserved. Next, the left plates of C_{1a} and C_{1b} swing to V_{REF} and zero, respectively [Fig. 19.17(c)]. The total charge on their right plates is now expressed as $(V_X - V_{REF})C_{1a} + V_X C_{1b} = -V_{REF}C_{1a} + V_X(C_{1a} + C_{1b})$. This amount must be equal to the original value, $-(C_{1a} + C_{1b})V_{in}$, yielding

$$V_X = -V_{in} + V_{REF} \frac{C_{1a}}{C_{1a} + C_{1b}}. \quad (19.7)$$

The comparator detects the polarity of V_X and hence interprets the second term on the right as its decision threshold. Selecting $C_{1a}/(C_{1a} + C_{1b}) = 1/4$ and hence $C_{1b} = 3C_{1a}$, we set this threshold to $+V_{REF}/4$.

⁴In practice, S_1 must be tied to a proper common-mode level rather than to ground.

The other comparator in the 1.5-bit MDAC operates similarly, except that the left plates of its C_{1a} and C_{1b} swing to zero and V_{REF} , respectively.

The switching scheme in Fig. 19.17 suffers from attenuation due to the comparator input capacitance, C_{comp} . For this reason, we choose $C_{1a} + C_{1b}$ to be 5 to 10 times C_{comp} .

19.3 Open-Loop Residue Amplification

The non-flip-around and flip-around MDACs studied thus far rely on op amps for precise residue amplification. The power-speed-gain trade-offs in op amps, however, present formidable challenges. We thus ponder the use of open-loop gain stages, seeking more relaxed constraints. Differential pairs prove attractive for this purpose.

The transformation of the previous MDACs to open-loop topologies proceeds as follows. We begin with the arrangement in Fig. 19.18(a), where (bootstrapped) switches S_1 and S_2 and capacitors C_1 and C_2 sample the analog input. The differential pair incorporates resistive degeneration for linearity and amplifies

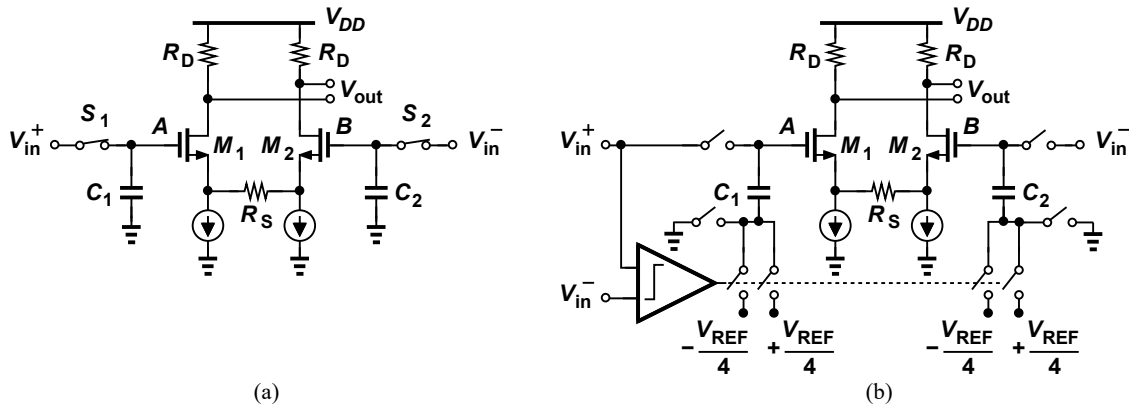


Figure 19.18 (a) Open-loop residue amplifier, and (b) its use in an MDAC.

the result. But this is not what we wish! The amplifier must amplify only the *residue*. That is, we must first generate a difference of the form $(V_{in}^+ - V_{in}^-) \pm KV_{REF}$ between nodes A and B. Illustrated in Fig. 19.18(b), this is accomplished by switching the bottom plates of C_1 and C_2 according to the comparator's decision. For example, if $V_{in}^+ < V_{in}^-$, these plates switch to $+V_{REF}/4$ and $-V_{REF}/4$, respectively, yielding $V_A = V_{in}^+ + V_{REF}/4$, $V_B = V_{in}^- - V_{REF}/4$, and hence $V_A - V_B = (V_{in}^+ - V_{in}^-) + V_{REF}/2$. For $V_{in}^+ > V_{in}^-$, we have $V_A - V_B = (V_{in}^+ - V_{in}^-) - V_{REF}/2$. Upon amplifying this result by a factor of 2, the differential pair provides $V_{out} = 2(V_{in}^+ - V_{in}^-) \pm V_{REF}$, as in the MDACs studied before.

The open-loop operation provided by the differential pair in Fig. 19.18(b) leads to much faster settling than that of closed-loop op amps. Nevertheless, this solution entails several drawbacks. First, the linearity may not suffice for sensing an input difference of $(V_{in}^+ - V_{in}^-) \pm V_{REF}/2$. For example, consider a 10-bit ADC where $V_{in}^+ - V_{in}^-$ is slightly greater than zero, i.e., $V_A - V_B \approx V_{REF}/2$, and $V_{REF} = 500$ mV. The differential pair must therefore amplify an input of 250 mV with at least 10-bit linearity, a daunting task.

Second, even if with adequate linearity, the gain of the circuit must be very precisely equal to 2. Recall from Chapter 18 that the gain error resulting from the finite op amp gain and capacitor mismatch translates to DNL and INL. In other words, the maximum residue generated by the circuit of Fig. 19.18(b) must precisely match the input full scale of the remainder of the pipeline. We recognize the difficulties here: (a) the input capacitance of M_1 and M_2 attenuates the voltage changes applied to the bottom plates of C_1 and C_2 , and (b) the voltage gain of the differential pair, given by $R_D/(R_S/2 + 1/g_{m1,2})$, cannot be exactly set to 2.0 in the presence of PVT variations. Both effects introduce substantial gain error.

The third drawback is that the differential pair's offset voltage is not canceled, a point of contrast to the flip-around MDAC (Chapter 18).

Some of the foregoing issues are alleviated if the pipelined ADC incorporates a flash front end, i.e., in a manner similar to the hybrid flash/SAR architecture studied in Section 16.1.3. We can thus deliver only a small residue to the differential pair. For example, a 5-bit flash stage with 1-bit of redundancy reduces the residue amplitude by a factor of 16.

19.4 Hybrid SAR/Pipelined ADCs

The analogy between pipelining and an assembly line (Chapter 17) suggests that different stages in the former need not be identical, just as different stations in the latter are not. We can then envision a multitude of arrangements employing both SAR and pipelining concepts. Shown in Fig. 19.19 are some examples that we study in this section.

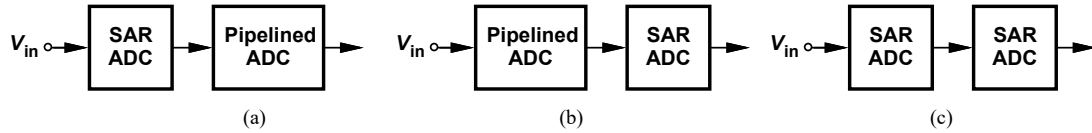


Figure 19.19 Different permutations of SAR and pipelined ADCs.

Our study of SAR and pipelined ADCs in the previous chapters indicates that both rely on binary search, but with vastly different clocking schemes and analog operations. Specifically, the SAR architecture does not require precision amplification while taking $N - 1$ cycles to resolve N bits, whereas the pipelined chain relies on high-gain op amps but determines 1 bit per cycle. Fortunately, the op-amp gain bounds are relaxed as the signal travels through the cascade. We thus envision that a SAR loop followed by a few pipelined stages can benefit from the advantages of both architectures.

19.4.1 Hybrid Example 1

Before constructing the first architecture, we recall two points from our previous investigations. First, a k -bit top-plate-sampling SAR converter requires that its DAC settle $k - 1$ times (Chapter 14). Second, at the end of conversion, the differential DAC voltage in such a loop is no more than 1 LSB = $V_{REF}/2^k$. This occurs because the circuit subtracts $V_{REF}/2$, $V_{REF}/4$, etc., from the input in successive cycles.

Suppose we wish to develop a 10-bit ADC that resolves 6 bits by a SAR loop and 4 bits by a pipeline. We begin with the top-plate-sampling SAR loop shown in Fig. 19.20(a), where one comparator input is initially equal to V_{in} . The comparator is clocked at $t = 0$ and the DAC capacitors' bottom plates are switched such that V_{DAC} decreases by $V_{REF}/2$, reaching a value of $V_{in} - V_{REF}/2$. This proceeds for a total of five cycles⁵ and yields a V_{DAC} equal to $V_{in} - \alpha V_{REF}/64$, where α is an integer [Fig. 19.20(b)]. We call this result the “residue” and note that it has a maximum value of $V_{REF}/64$ (why?). The concept is similar to that in pipelined ADCs, except that this residue does not contain an *amplified* copy of the main input. After the fifth settling of the DAC, the comparator is clocked once more (at $t = t_1$). We have thus resolved 6 bits.

For the remaining 4 bits, we turn to a pipelined cascade that treats V_{res} as its input [Fig. 19.20(c)]. Based on the last decision made by the SAR comparator, the first pipelined MDAC generates a new residue, V_{res2} . But V_{res2} is *not* of the form $2V_{res1} \pm V_{REF}$ because the maximum value of V_{res1} and hence the full scale of the pipeline are equal to $V_{REF}/64$ (why?). The first MDAC must therefore deliver $V_{res2} = 2V_{res1} \pm (V_{REF}/64)$.

How does the first MDAC sense V_{DAC} in Fig. 19.20(c) without drawing charge from the SAR DAC capacitors? This requires that we interpose a buffer between the DAC and the MDAC [Fig. 19.20(d)]. During SAR operation, the input capacitors of the MDAC track V_{res1} . When the SAR conversion is completed, the SAR DAC capacitors begin to sense V_{in} while the MDAC enters the amplification mode. The pipeline must employ three more stages for an overall resolution of 10 bits.

The hybrid architecture described above offers three advantages. First, since the SAR DAC settles only five times, the conversion takes less time than in a 10-bit loop. Second, the lower SAR resolution reduces

⁵Recall that the top-plate-sampling loop in Chapter 14 takes $N - 1$ cycles to resolve N bits.

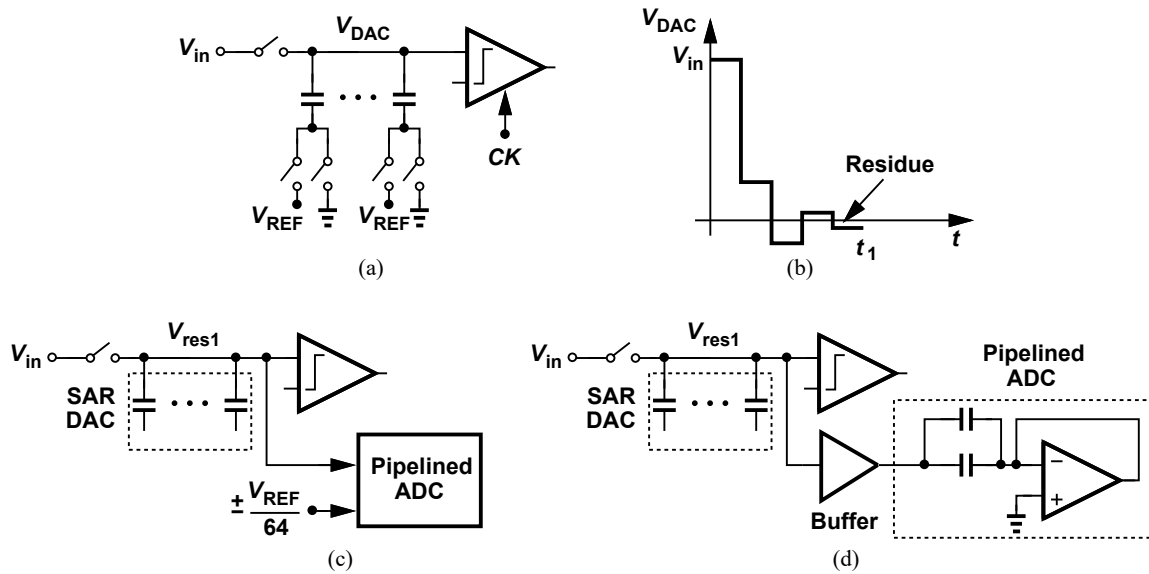


Figure 19.20 (a) Top-plate-sampling SAR ADC, (b) V_{DAC} behavior, (c) SAR loop followed by pipelined ADC, and (d) use of a buffer.

the complexity of its DAC(s). Third, since the first MDAC senses and generates a small residue, its op amp gain and swing requirements are greatly relaxed. Nonetheless, this approach faces three issues. First, the buffer must present a precise gain so that the maximum value of V_{res1} matches the full scale of the pipeline. (Recall from Chapter 18 that gain error in the residue translates to DNL and INL.) In our 10-bit example, the SAR loop resolves 5 bits, dictating a gain error of roughly less than $1/64 \approx 1.6\%$ for the buffer, a difficult constraint. Second, with a full scale equal to $V_{REF}/64$, the pipeline necessitates a reference of this value and hence a resistive divider. Third, the SAR comparator offset and the buffer offset corrupt V_{res1} and must be well below 1 LSB of the overall ADC.

19.4.2 Hybrid Example 2

It is possible to construct a cascade consisting of a front-end MDAC and a SAR loop [1]. As illustrated in Fig. 19.21(a), while the flip-around MDAC samples V_{in} , the SAR ADC digitizes the previous residue. Next, the MDAC generates a new residue, impressing it on the SAR DAC capacitors [Fig. 19.21(b)].

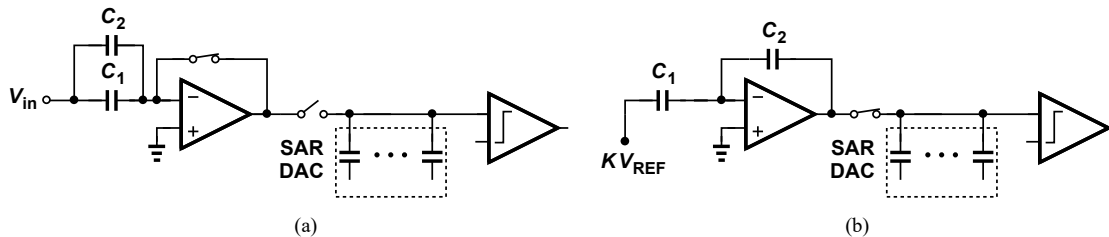


Figure 19.21 (a) Flip-around MDAC followed by SAR ADC, and (b) MDAC in amplification mode.

The foregoing architecture does not appear to offer any advantages. The MDAC reduces the SAR conversion time by only one cycle, i.e., the SAR loop still proves a serious speed bottleneck. The MDAC resolution can be raised if the single comparator is replaced with a flash sub-ADC. For example, [1] employs 2.5 bits. Nonetheless, the SAR section still does not outperform a simple chain of MDACs. Indeed, we surmise that, if the first MDAC does satisfy the overall ADC's speed and resolution requirements, then the following MDACs in a simple pipeline would also provide the necessary performance—and their power consumption could be

scaled to some extent. For these reasons, this permutation of pipelining and SAR stages does not appear particularly efficient.

Example 19.7

Suppose the first stage in Fig. 19.21 resolves 2 bits and amplifies the residue by a factor of 2 [Fig. 19.22(a)]. Assume $C_1 = \dots = C_4 = C_F/4$. Plot V_{res1} as a function of V_{in} . How should the SAR loop reference voltage be chosen?

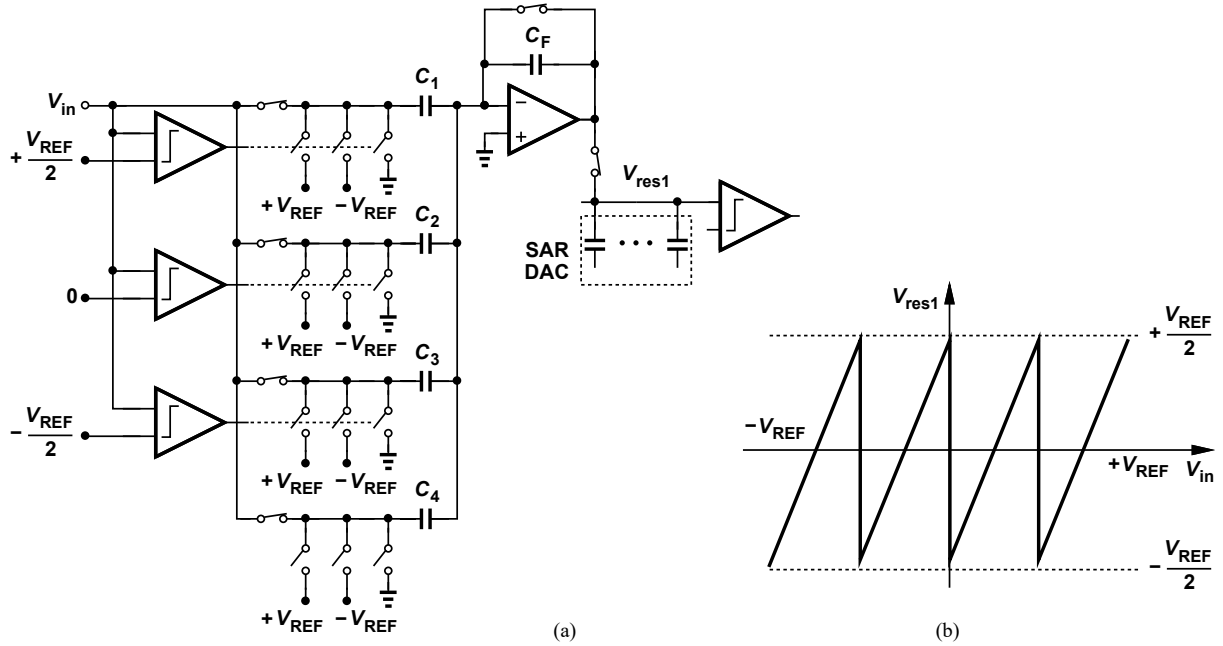


Figure 19.22 (a) Cascade of flash, pipelined, and SAR stages, and (b) behavior of residue.

Solution

The three comparators in the flash sub-ADC compare V_{in} to $-V_{REF}/2$, 0 , and $+V_{REF}/2$. The residue plot thus contains four sections [Fig. 19.22(b)]. For example, the leftmost section is given by $V_{res1} = V_{in} + 3V_{REF}/4$ and occurs for $V_{in} < -V_{REF}/2$.

The 2-bit MDAC operates as follows. If $-V_{REF} < V_{in} < -V_{REF}/2$, then the left plate of C_4 swings to zero and those of C_1 – C_3 to $-V_{REF}$. The reader can prove that $V_{res1} = V_{in} + 3V_{REF}/4$.⁶ If $-V_{REF}/2 < V_{in} < 0$, then the left plates of C_4 and C_3 travel to zero and those of C_1 and C_2 to $-V_{REF}$, delivering $V_{res1} = V_{in} + V_{REF}/4$. For $V_{in} > 0$, similar events occur but the left plates switch to $+V_{REF}$ instead of $-V_{REF}$. Note that V_{res1} varies from $-V_{REF}/2$ to $+V_{REF}/2$. Thus, the SAR reference voltages are equal to $\pm V_{REF}/2$, requiring a resistive divider.

19.4.3 Hybrid Example 3

From the conceptual diagram in Fig. 19.19(a), we surmise that two (or more) SAR loops can be pipelined such that each requires fewer cycles for conversion. Shown in Fig. 19.23 is an example [2], where the residue generated by the first SAR stage is buffered and applied to another loop. For example, a 10-bit system can employ two 5-bit SAR ADCs, allowing substantially less complex DACs.

⁶This is true in a differential MDAC.

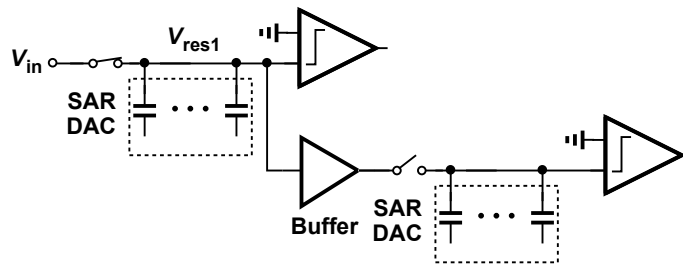


Figure 19.23 Cascaded and pipelined SAR stages.

The buffer in Fig. 19.23 can provide a voltage gain greater than unity so as to relax the noise and offset bounds of the second SAR comparator [2]. Of course, the buffer's noise and offset must be sufficiently small. Furthermore, the buffer's gain error introduces DNL and INL, as mentioned in previous sections. The design in [2] employs calibration to ensure that the maximum output of the buffer/amplifier matches the full scale of the second SAR stage. Note that this full scale is not equal to that of the first SAR ADC unless the buffer's gain is chosen to be 2^n , where n is the resolution of the first loop.

Example 19.8

Explain how the offsets in Fig. 19.23 affect the performance.

Solution

The residue generated by the first stage is corrupted by the first comparator's offset, V_{OS1} , and the buffer's, V_{OSb} . That of the second comparator, V_{OS2} , has a similar effect but it is divided by the buffer's gain, A_b . Thus, $V_{OS1} + V_{OSb} + V_{OS2}/A_b$ must be maintained well below 1 LSB.

An interesting variant of the above architecture is described in [3]. Illustrated in Fig. 19.24, the ADC consists of two pipelined SAR loops and a residue amplifier. However, each SAR stage relies on *different* comparators for different cycles. For example, Comp_1 is engaged for the first cycle, Comp_2 for the second, etc. That is, after Comp_1 is clocked and its decision is conveyed to the DAC, it does not participate in the conversion until the remaining SAR cycles are completed. This approach allows the comparators to have a longer reset time and even perform offset cancellation after each cycle.

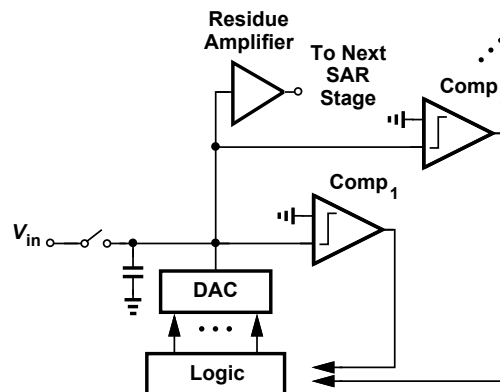


Figure 19.24 Use of multiple comparators in a SAR loop.

The foregoing idea faces three issues. First, the offset of the comparators must be canceled. Second, it incurs an area and complexity penalty due to the larger number of comparators and their offset cancellation

networks. Third, the input capacitances of the comparators load the DAC and introduce gain error, a serious issue in view of the pipelined structure. This error can be calibrated along with the gain of the residue amplifier.

19.4.4 Hybrid Example 4

From our analysis of pipelined ADCs in Chapter 18, we have learned that the op-amp open-loop gain, A_0 , must be sufficiently high to achieve an acceptably low gain error for the MDAC. For example, a 10-bit system requires $A_0 > 3,800$, a severe challenge in op amp design.

A key point coming to our aid here is that the effect of the finite op-amp gain manifests itself at the MDAC's virtual ground. Specifically, for an output of V_1 , the virtual ground voltage, V_X , is equal to $V_1/(-A_0)$. We then wonder whether it is possible to sense V_X independently and use this information to improve the MDAC's accuracy [Fig. 19.25(a)]. For simplicity, we assume the left plate of C_1 swings to zero. We must then decide how the sensor's result should correct the residue. The critical point here is that the total charge at X is constant and must be simply *redistributed* so that it mostly resides on C_2 .

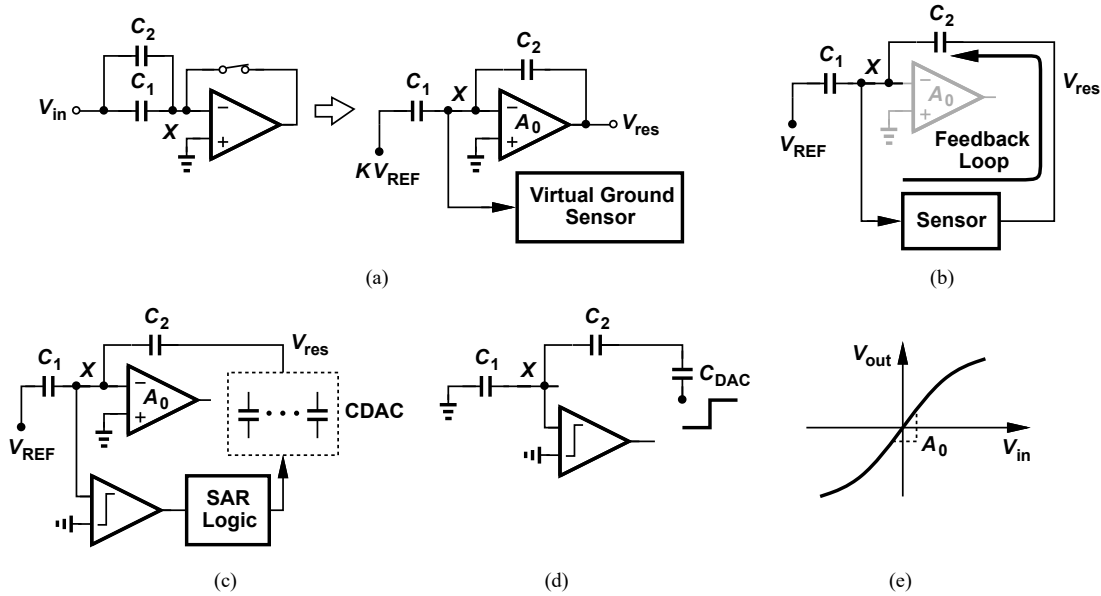


Figure 19.25 (a) MDAC with a mechanism for sensing the virtual ground voltage, (b) feedback loop driven by sensor for correcting the gain error, (c) use of a SAR loop for sensing and correcting the gain error, (d) a section of the circuit for loop gain calculation, and (e) op-amp characteristic.

We can envision an extra phase, depicted in Fig. 19.25(b), in which C_2 is disengaged from the op amp output and is driven by the sensor so that the new feedback loop drives V_X toward zero. This means that the sensor must provide a high loop gain, but without a high-gain amplifier! Fortunately, SAR ADCs exhibit a high loop gain (Chapter 14), with great ability to enforce a virtual ground at their comparator's input. We therefore construct the arrangement shown in Fig. 19.25(c), where the comparator measures V_X and drives the SAR logic, the capacitor DAC, and C_2 so as to force V_X toward zero in successive cycles [4]. Note that the total charge at X remains unchanged, migrating from the right plate of C_1 to the left plate of C_2 . If $V_X \rightarrow 0$, then V_{res} is highly accurate. The work in [4] achieves a resolution of 12 bits with an op-amp gain of only 20 dB.

This hybrid system merits several remarks. First, since the total charge at X does not change, bringing V_X to zero causes the charge on C_2 to be equal to that in the sampling mode, yielding $V_{res} = 2V_{in} - KV_{REF}$. That is, the nearly infinite gain provided by the comparator drives V_X toward zero even though the CDAC experiences heavy loading by the MDAC capacitors. As shown in Fig. 19.25(d), the loop gain is reduced by the voltage division resulting from C_1 , C_2 , and C_{DAC} . Second, each conversion period must accommodate

the SAR loop cycles, making this architecture suited to low speeds. Third, both the offset and noise of the comparator corrupt the final value of V_X and must be well below 1 LSB of the overall ADC.

It is interesting to note that the SAR loop in Fig. 19.25(c) also suppresses the op amp nonlinearity [4]. This point can be seen from the op amp characteristic depicted in Fig. 19.25(e): since (static) nonlinearity is equivalent to variation of slope and hence gain, any mechanism that makes the residue independent of the op-amp gain also removes nonlinearity. This is an important consideration as low-gain op amps typically exhibit high nonlinearity as well.

Example 19.9

Given the long SAR convergence time in Fig. 19.25(c), a student reasons that one can simply design a slow, low-power, high-gain op amp and employ a conventional pipeline to obtain similar results. Is the student's point valid?

Solution

It is valid to some extent. However, a low-power op amp inevitably suffers from high noise. A low-noise comparator, on the other hand, is typically more power-efficient.

References

- [1]. M. Brandolini, et al, "A 5 GS/s 150 mW 10 b SHA-less pipelined/SAR hybrid ADC for direct-sampling systems in 28 nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 50, pp. 2922-2934, Dec. 2015.
- [2]. F. van der Goes, et al, "A 1.5 mW 68 dB SNDR 80 Ms/s 2x interleaved pipelined SAR ADC in 28 nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 49, pp. 2835-2845, Dec. 2014.
- [3]. L. Kull, et al, "A 10b 1.5GS/s pipelined-SAR ADC with background second-stage common-mode regulation and offset calibration in 14nm CMOS FinFET," *ISSCC Digest*, Feb. 2017.
- [4]. K. Yoshioka, et al, "A 0.7V 12b 160MS/s 12.8fJ/conv-step pipelined-SAR ADC in 28nm CMOS with digital amplifier technique," *ISSCC Digest*, Feb. 2017.

Problems

- 19.1. For the sake of simplicity, an engineer uses only one comparator in the flash stage of Fig. 19.1(a). Redraw the diagrams of Figs. 19.1(b), (c), and (d) for this case.
- 19.2. Suppose the comparator in the previous problem exhibits a small offset, V_{OS} . If the SAR stage is designed for a resolution of 3 bits, plot the residue and the overall input-output characteristic.
- 19.3. Is it possible to incorporate overlap and redundancy in Problem 19.1 so as to accommodate the offset of the single comparator? Explain.
- 19.4. We design the hybrid ADC of Fig. 19.1(a) for a resolution of 6 bits and add 1 bit of redundancy to allow comparator offsets in the flash stage. Determine the maximum tolerable comparator offset if the flash ADC has a resolution of 3 bits or 4 bits (and hence the SAR loop has a resolution of 4 bits or 3 bits, respectively).
- 19.5. Suppose the capacitor DAC in Fig. 19.1 exhibits an offset, V_{OS} . That is, its input-output characteristic is shifted up by this amount. Explain how the offset affects the overall ADC characteristic. Can overlap and redundancy correct for this effect?
- 19.6. Repeat the analysis depicted in Fig. 19.7(e) if the comparator offset, V_{OS} , has the opposite sign.
- 19.7. Construct the residue plot of Fig. 19.7(b) if the flash stage employs only one comparator.
- 19.8. Construct the residue plot of Fig. 19.11(e) if only the lower comparator in Fig. 19.11(c) incurs an offset.
- 19.9. Suppose both $+V_{REF}/4$ and $-V_{REF}/4$ in Fig. 19.11(c) are shifted up by ΔV . Explain what happens to the residue plot.
- 19.10. Consider two cascaded 1.5-bit MDACs, which ideally exhibit the residue behavior shown in Fig. 19.14(d). Explain how this plot changes if a comparator in the second stage exhibits an offset equal to V_{OS} .
- 19.11. Repeat the analysis depicted in Fig. 19.16 if $V_{in} = -3.2$ LSB or $+2.5$ LSB.
- 19.12. A 1.5-bit MDAC incorporates two comparators based on the topology shown in Fig. 19.17. Explain what happens to the residue plot if $C_{1a}/(C_{1a} + C_{1b}) = 1/4 + \epsilon$, where ϵ is an error resulting from the mismatch between the two capacitors.
- 19.13. Suppose the hybrid architecture of Fig. 19.20(d) employs a 2-bit SAR loop and a 1.5-bit MDAC as the first stage of the pipelined ADC. Plot the MDAC residue versus V_{in} if the buffer has an offset of V_{OS} .
- 19.14. Repeat the previous problem if only the SAR comparator incurs an offset of V_{OS} .
- 19.15. Repeat Problem 19.13 if only the op amp has an offset of V_{OS} .
- 19.16. Consider the hybrid architecture of Fig. 19.22(a). Explain what happens to the V_{res1} plot if C_F is slightly greater than its nominal value.
- 19.17. In the hybrid architecture of Fig. 19.22(a), the op amp exhibits an offset of V_{OS} . Explain what happens to the residue plot. Can redundancy and overlap deal with this offset?
- 19.18. Suppose we wish to implement pipelining in the hybrid architecture of Fig. 19.23. This requires that the residue produced by the first SAR stage be sampled by the second. Draw a detailed timing diagram for the two stages and explain how sampling and SAR conversion proceed in each. Assume each stage requires one clock cycle for sampling and N clock cycles for conversion.
- 19.19. The buffer in Fig. 19.23 exhibits offset and gain error. Explain whether redundancy and overlap can deal with those imperfections.
- 19.20. Repeat the previous problem if the two comparators in Fig. 19.23 incur offset voltages.

Interleaved ADCs

In our study of ADCs, we have seen trade-offs between speed and power consumption and methods of raising the conversion rate. Another approach to increasing the speed and easing its trade-off with power draws upon the concept of “time interleaving,” or simply “interleaving.” Applicable to all Nyquist-rate architectures, this idea affords a considerable boost in the performance and is widely used today.

In this chapter, we begin with the basic interleaving concept and its advantages, examining its behavior in both time and frequency domains. We then address the problem of interchannel mismatches. Next, we deal with interleaving techniques and study various mismatch calibration methods. Finally, we deal with multiphase clock generation for interleaved ADCs. The reader is encouraged to review Chapter 3 before proceeding here.

20.1 General Considerations

20.1.1 Basic Idea

An ADC spends part of the clock cycle, T_{CK} , on sampling and the other on conversion (quantization). Suppose, for now, that each mode lasts $T_{CK}/2$ seconds [Fig. 20.1(a)]. We call this sampler-quantizer cascade a “channel.” Let us add another channel that operates with the opposite phase of the clock [Fig. 20.1(b)]. When Channel 1 is in the sampling mode (between $t = 0$ and $t = T_{CK}/2$), Channel 2 holds and digitizes V_Y and vice versa. That is, by interleaving the two channels, the system samples and quantizes V_{in} *twice* per clock cycle, generating D_{out1} and D_{out2} . Called an interleaved ADC, the overall circuit, in principle, achieves twice the speed at the cost of twice the power and chip area. The “interleaving factor,” M , is equal to 2 here, allowing a twofold increase in the analog input signal bandwidth.

It is helpful to visualize the operation of the above two-channel ADC as follows. Channel 1 produces “odd-numbered” samples of the input, depicted by empty dots in Fig. 20.2(a). Similarly, Channel 2 generates “even-numbered” samples, shown by solid dots. Equivalently, the system multiplies V_{in} by two trains of impulses [Fig. 20.2(b)], one offset with respect to the other by $T_{CK}/2$ seconds. Noting that $D_{out1} \equiv y_1(t)$ and $D_{out2} \equiv y_2(t)$, the reader naturally asks how we should “combine” D_{out1} and D_{out2} so as to produce a single digital output representing V_{in} . We address this question below, but it seems that *adding* $y_1(t)$ and $y_2(t)$ yields the complete signal.

For our subsequent derivations, we recall three points from the theory of signals. First, for a general function, $g(t)$, we have $g(t)\delta(t - T_0) = g(T_0)\delta(t - T_0)$. This “sifting” property holds in the frequency domain as well: $W(f)\delta(f - f_0) = W(f_0)\delta(f - f_0)$. Second, a time shift translates to an exponential factor in the Fourier transform, i.e., $x(t - T_0)$ yields $X(f) \exp(-j2\pi f T_0)$. Third, convolution of a signal with an impulse (in the time or frequency domain) creates a shift, e.g., $W(f) \star \delta(f - f_0) = W(f - f_0)$.

We can gain further insights into interleaving by assuming that the ADC input, $x(t)$, is a random signal and displays a spectrum from $-f_{in}$ to $+f_{in}$ [Fig. 20.3(a)]. We expect that the two-channel ADC samples $x(t)$ properly (i.e., satisfies Nyquist’s sampling theorem) if the clock frequency, f_{CK} , exceeds f_{in} (why?).

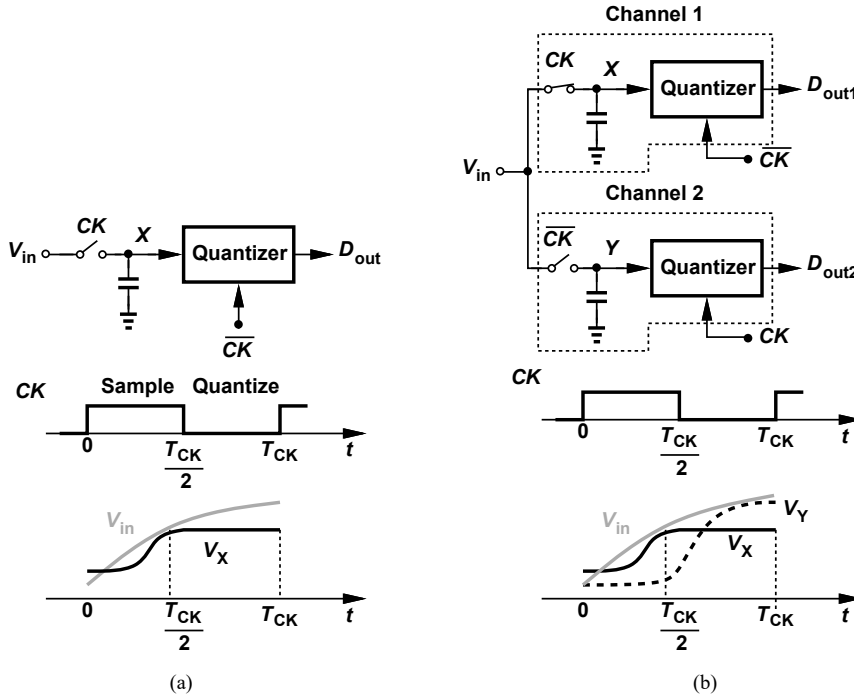


Figure 20.1 (a) Single-channel ADC, and (b) two time-interleaved channels.

From Chapter 3, impulse sampling in the two channels gives outputs of the form:

$$y_1(t) = x(t) \sum_n \delta(t - nT_{CK}), \quad (20.1)$$

$$y_2(t) = x(t) \sum_n \delta\left(t - nT_{CK} - \frac{T_{CK}}{2}\right). \quad (20.2)$$

In the frequency domain, the spectrum of $x(t)$ is convolved with a train of impulses; consequently, replicas of $X(f)$ are shifted to integer multiples of f_{CK} [Fig. 20.3(b)]. We have¹

$$Y_1(f) = \frac{1}{T_{CK}} \sum_n X(f - nf_{CK}), \quad (20.3)$$

$$Y_2(f) = \frac{1}{T_{CK}} \sum_n X(f - nf_{CK}) \exp[-j\pi(nf_{CK})T_{CK}]. \quad (20.4)$$

The exponential factor is equal to $+1$ for even n and -1 for odd n . If f_{CK} is slightly greater than f_{in} , the resulting spectra appear as shown in Fig. 20.3(b). Violating Nyquist's theorem, each channel exhibits heavy aliasing, but if we add $Y_1(f)$ and $Y_2(f)$, the aliased components are removed.

The foregoing studies suggest that $y_1(t)$ and $y_2(t)$ in Fig. 20.2(b) should be *added* so as to produce the overall digital output [Fig. 20.4(a)]. However, sketching their equivalent digital values as a function of time [Fig. 20.4(b)] reveals that addition in the digital domain does not provide the correct output because D_{out1} and D_{out2} do not behave as impulses. We instead *multiplex* the two data streams. From another perspective, multiplexing discrete-time signals [such as those in Fig. 20.2(b)] is equivalent to adding them.

In practice, the multiplexing operation is unnecessary because the digital processor following the ADC prefers to operate with lower data rates.

¹For a random signal, the spectrum is obtained by taking the Fourier transform of its autocorrelation rather than the Fourier transform of the signal itself. We do not make this distinction here.

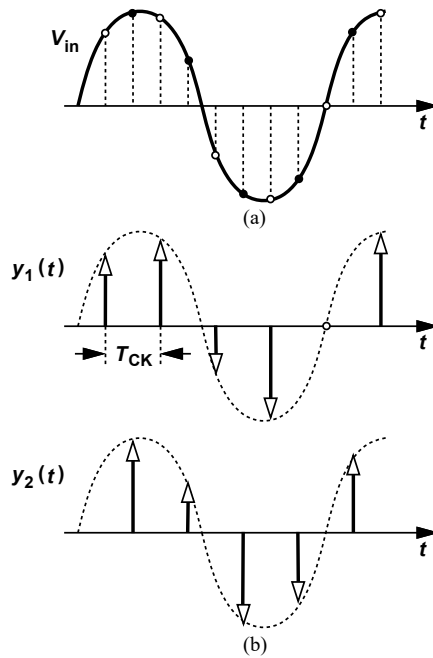


Figure 20.2 (a) Ideal interleaved sampling, and (b) odd and even channel outputs.

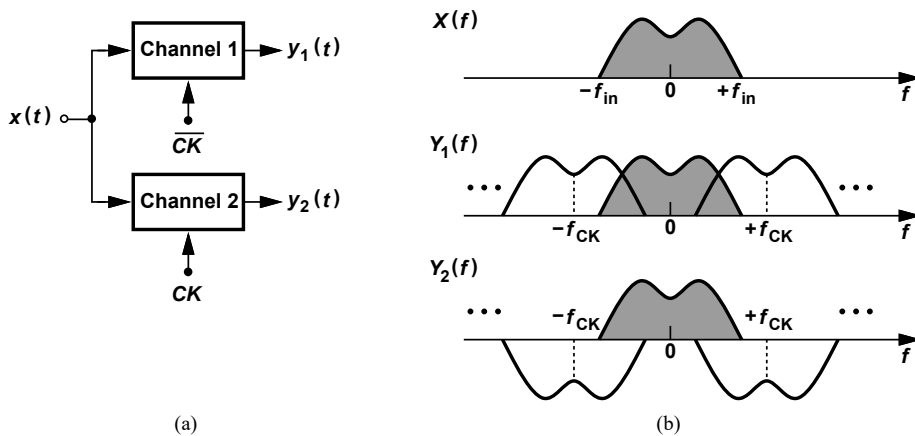


Figure 20.3 (a) Two interleaved channels clocked by CK and $\overline{CK}$, and (b) input and output spectra.

Example 20.1

Construct a four-channel interleaved ADC.

Solution

Figure 20.5(a) depicts a simple extension of the idea. The four channels can be driven by the “nonoverlapping” clocks shown in Fig. 20.5(b). Each channel samples V_{in} for $T_{CK}/4$ seconds and digitizes it for $3T_{CK}/4$ seconds. Thus, D_{out1} , D_{out2} , $\dots$ correspond to $V_{in}(t_1)$, $V_{in}(t_2)$, $\dots$, respectively.

The clocking scheme in Fig. 20.5(b) requires that the samplers fully acquire V_{in} in one quarter of the clock cycle, a challenging constraint at very high speeds. Alternatively, we can opt for the *overlapping* clock waveforms shown in Fig. 20.5(c), where each channel samples for $T_{CK}/2$ seconds. Of course, each channel now has less time for conversion. Also, during the overlap time, *two* sampling capacitors load the analog

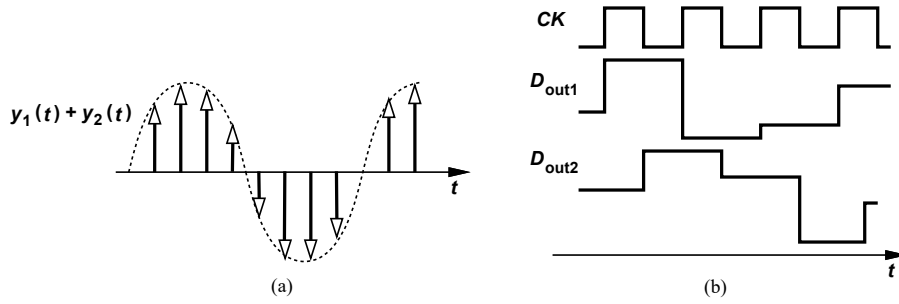


Figure 20.4 (a) Addition of the discrete-time samples produced by two interleaved channels, and (b) actual output data sequences.

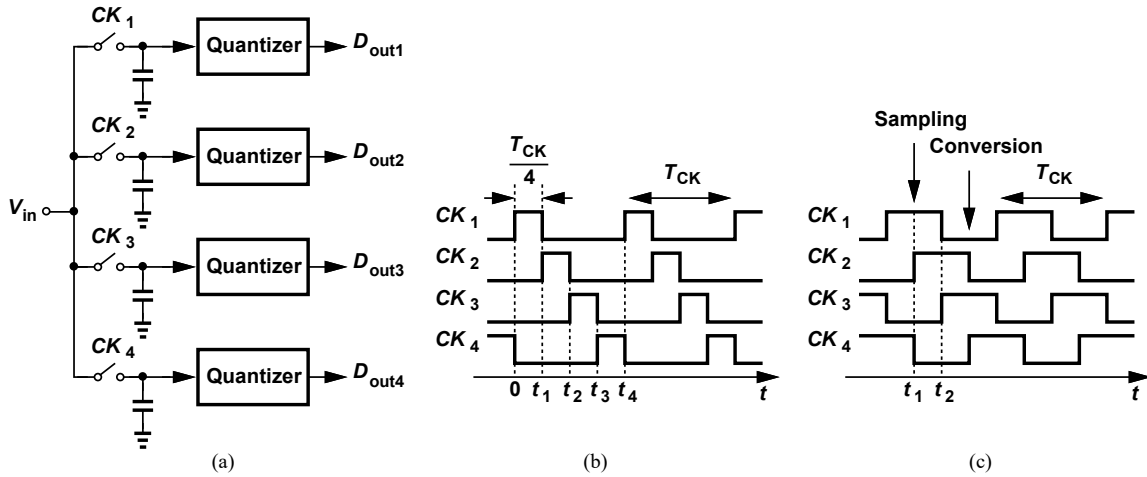


Figure 20.5 (a) Four interleaved channels, (b) nonoverlapping clocks, and (c) overlapping clocks.

input. For example, both Channel 1 and Channel 2 are in the sampling mode from t_1 to t_2 . This issue proves problematic in the design of a buffer that would precede the ADC. We delve into these issues later in this chapter.

20.1.2 Advantages of Interleaving

While originally conceived to increase the conversion rate [1], interleaving offers several other advantages as well.

Improved Speed-Power Trade-Off For a given desired resolution, an ADC's speed trades with its power consumption. But a key point is that this trade-off becomes *nonlinear* as the conversion rate exceeds a certain limit (Fig. 20.6). This is because, for a given technology node, each circuit function faces some speed limitation that cannot be improved much even if it draws a great deal of power. For example, a CMOS inverter with a unity fanout exhibits a certain delay (e.g., 5–6 ps in 28-nm technology) that is relatively independent of its power consumption. Similarly, a StrongArm comparator (Chapter 5) takes some time to respond regardless of how much power we provide to it. In fact, as explained in Chapter 6, a SAR loop entails a minimum delay that is given by the technology node and places an upper bound on the conversion speed. For this reason, the figure of merit of ADCs (Chapter 12) tends to degrade at higher rates even though it accounts for the speed.

Interleaving favorably changes the power-speed trade-off. Suppose, for example, we wish to develop a 10-bit 10-GHz ADC. We first design a single-channel converter and determine how fast it can run and yet

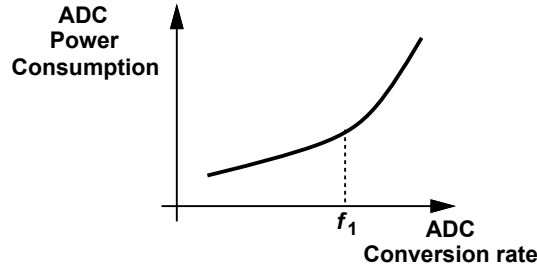


Figure 20.6 Conceptual power-speed trade-off.

maintain a linear power-speed trade-off (i.e., up to sampling rate f_1 in Fig. 20.6). If, for example, $f_1 = 1$ GHz, then we must interleave 10 channels. Each still consumes optimal power. Of course, the ADC chip area increases by the same factor. Also, the generation and distribution of 10 clock phases draw some power (Section 20.6).

Lower Metastability Errors Time interleaving can dramatically reduce the probability of metastable states in ADCs. Let us assume that a single-channel ADC running at a certain rate, f_S , employs a comparator having a regeneration time of τ_{reg} . As shown in Chapter 6, the probability of a metastable state is given by $\alpha \exp(-T_1/\tau_{reg})$, where α is a proportionality factor and T_1 is the nominal time allocated for the comparator's decision.

Now, we instead interleave M such ADCs, assuming that the overall system still samples at f_S but the allowable comparator decision time can be increased to MT_1 . Each channel therefore exhibits an error rate of

$$P_e = \alpha \exp \frac{-MT_1}{\tau_{reg}}. \quad (20.5)$$

The exponential dependence upon MT_1 reveals a considerable drop in metastable states. For example, if $T_1/\tau_{reg} = 10$ and $\alpha = 1$, then P_e falls from 4.5×10^{-5} to 2.1×10^{-9} if M goes from 1 to 2.

20.1.3 Disadvantages of Interleaving

In addition to the direct area penalty, interleaving entails three other issues as well. First, the multiple nonoverlapping or overlapping clock phases described earlier must be generated and distributed across all of the channels, a daunting challenge at high speeds. Second, interleaving raises the ADC input capacitance. From Example 20.1, we note that overlapping clock phases cause $M/2$ samples to load the input simultaneously. One can select nonoverlapping phases to avoid this effect, but at the cost of a shorter sampling time. In addition, the longer routing of the analog input signal across M channels leads to a greater input capacitance. Third, mismatches among the channels' offsets, gains, and phases corrupt the signal, often requiring calibration. We deal with this issue extensively in subsequent sections.

20.2 Effect of Mismatches

Recall from Fig. 20.3(b) that each interleaved channel can exhibit heavy aliasing due to sampling the input at a clock frequency lower than the Nyquist rate. Nevertheless, the addition of $Y_1(f)$ and $Y_2(f)$ removes the aliased replicas centered around $\pm f_{CK}$ if these components match *exactly*. Otherwise, the cancelation is not perfect and the desired signal is corrupted.

Mismatches among the offsets, gains, and phases of interleaved channels can be studied in the time or frequency domains with sinusoidal or random inputs. Each of these four approaches provides its own insights.

Most of our treatment focuses on two-channel systems. We denote the sampling rate per channel by f_{CK} and the overall sampling rate by f_S . In a two-channel ADC, for example, $f_S = 2f_{CK}$.

20.2.1 Offset Mismatch

We have seen in previous chapters that an ADC's characteristic can shift up or down due to certain imperfections. For example, if two front-end sampling switches that sense differential signals suffer from a threshold mismatch, they inject different amounts of charge onto the sampling capacitors. The resulting offset is typically benign in single-channel designs but not in interleaved converters. Since the dc offset is random, it assumes different values in different channels, leading to offset *mismatch*.

To study this phenomenon, we set the analog input to zero but keep the offsets [Fig. 20.7(a)]. The channels digitize their own offsets and yield the overall output shown in Fig. 20.7(b). Since V_{OS1} repeats every T_{CK} seconds, it can be viewed as a train of impulses in the time domain, yielding a first harmonic at f_{CK} in the frequency domain. Similarly, the periodic behavior of V_{OS2} gives a component at f_{CK} but with the *opposite* sign because this train is offset by $T_{CK}/2$ with respect to the first. The net result is therefore expressed as $(V_{OS1} - V_{OS2}) \cos(2\pi f_{CK} t)$, where we have excluded components at $f = 0$ and beyond $\pm f_S$. Figure 20.7(c) depicts the resulting spectrum between $-f_S$ and $+f_S$. Pure sinusoids are also called “tones.” If a tone's frequency is not a harmonic of the desired input or the clock, it is also called a “spur.”

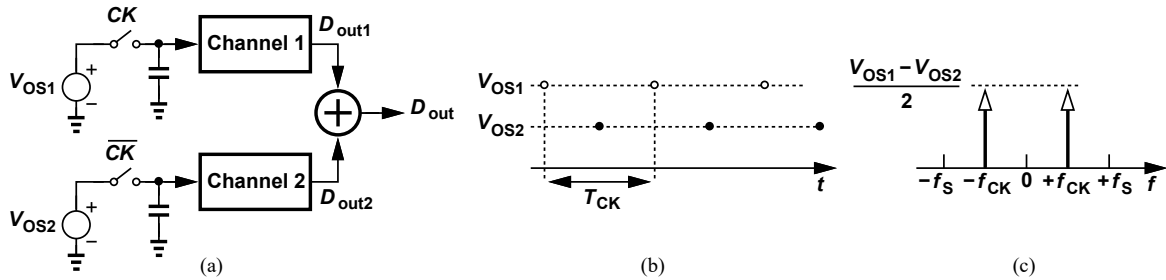


Figure 20.7 (a) Two interleaved channels with different offsets, (b) time-domain outputs, and (c) resulting spectrum.

Example 20.2

The circuit of Fig. 20.7(a) receives a random signal such as that shown in Fig. 20.3(a). Construct the spectrum of the final output.

Solution

As noted in Fig. 20.3(b), the aliased replicas centered around $\pm f_{CK}$ cancel each other. We thus arrive at the spectrum shown in Fig. 20.8, observing that the tone arising from offset mismatch appears near the edge of the band.² Is this tone objectionable? It is possible that the digital signal processor following the ADC further

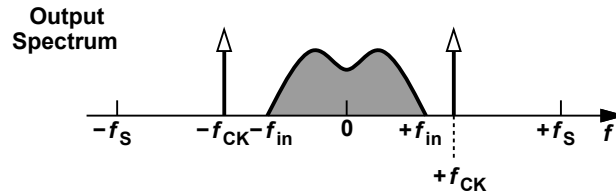


Figure 20.8 Output spectrum including tones due to offset mismatch.

corrupts the signal by the tone. The situation is more severe for greater interleaving factors, as studied below.

²We have ignored the impulse at $f = 0$.

It is instructive to repeat the above analysis for four interleaved channels having offsets equal to V_{OS1} , V_{OS2} , V_{OS3} , and V_{OS4} . The channels sample at a frequency of f_{CK} and the overall conversion rate is $f_S = 4f_{CK}$. As illustrated in Fig. 20.9(a), a fixed pattern of V_{OS1} , V_{OS2} , V_{OS3} , and V_{OS4} repeats every T_{CK} seconds. This corresponds to four sinusoids at f_{CK} but shifted by $T_{CK}/4$ ($\equiv \pi/2$), $T_{CK}/2$ ($\equiv \pi$), and

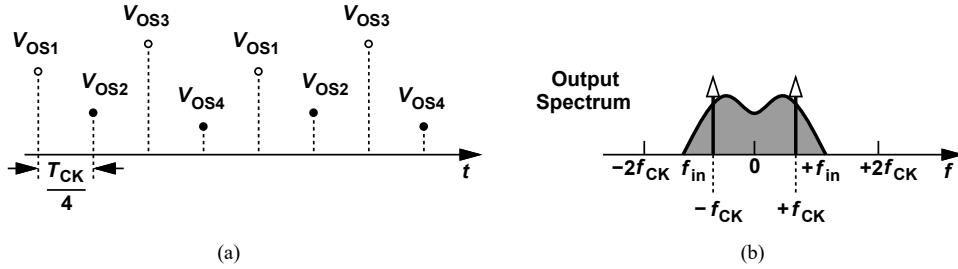


Figure 20.9 (a) Outputs of four channels with offsets, and (b) resulting spectrum.

$3T_{CK}/4$ ($\equiv 3\pi/2$). We then write the equivalent final output as

$$V_{eq}(t) = V_{OS1} \cos(2\pi f_{CK}t) + V_{OS2} \cos\left(2\pi f_{CK}t - \frac{\pi}{2}\right) \quad (20.6)$$

$$+ V_{OS3} \cos(2\pi f_{CK}t - \pi) + V_{OS4} \cos\left(2\pi f_{CK}t - \frac{3\pi}{2}\right), \quad (20.7)$$

where higher harmonics are ignored. The first and third terms simply subtract and so do the second and fourth terms:

$$V_{eq}(t) = (V_{OS1} - V_{OS3}) \cos(2\pi f_{CK}t) \quad (20.8)$$

$$+ (V_{OS2} - V_{OS4}) \sin(2\pi f_{CK}t). \quad (20.9)$$

The resulting tone therefore has an amplitude of $\sqrt{(V_{OS1} - V_{OS3})^2 + (V_{OS2} - V_{OS4})^2}$ (why?) and still a frequency of f_{CK} , but the input signal bandwidth can extend to $2f_{CK}$ because of the interleaving factor of 4. Figure 20.9(b) depicts the output spectrum in the presence of a random signal, revealing that the tone now appears *within* the input band, possibly creating corruption. For this reason, we typically ensure that the tone amplitude is well below 1 LSB. Methods of canceling these offsets are described in Section 20.5.3.

We should remark that dc offsets have become undesirable even in single-channel ADCs. Owing to today's low supply voltages and hence the limited full scale, the presence of offset requires that the input swing be reduced (Fig. 20.10), thereby decreasing the ADC dynamic range. We therefore strive to maintain an offset of no more than a few LSB even in single-channel ADCs.

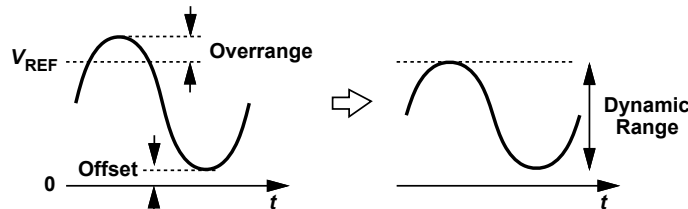


Figure 20.10 Effect of offset on dynamic range.

20.2.2 Gain Mismatch

Gain mismatch between interleaved channels can be visualized as shown in Fig. 20.11. For a mismatch of $\alpha \ll 1$, the odd samples and even samples, respectively emerge as

$$y_1(t) = \sum_n x(t) \delta(t - kT_{CK}), \quad (20.10)$$

$$y_2(t) = \sum_n (1 + \alpha)x(t) \delta\left(t - kT_{CK} - \frac{T_{CK}}{2}\right). \quad (20.11)$$

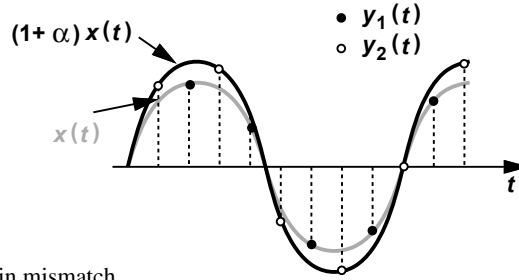


Figure 20.11 Illustration of gain mismatch.

The cancelation of the aliased replicas in Fig. 20.3(b) requires that the two interleaved channels exhibit exactly equal gains. Suppose the two channels display a voltage gain mismatch of α . Then, as shown in Fig. 20.12, the desired signal is corrupted by the residual aliased components. To quantify this phenomenon, we

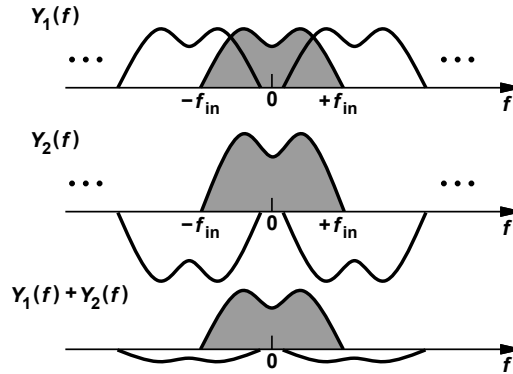


Figure 20.12 Effect of gain mismatch in the frequency domain.

first compute the power of $x(t)$ by integrating its spectrum from $-f_{in}$ to $+f_{in}$. We denote this power by P_x . Since both $Y_1(f)$ and $Y_2(f)$ carry the input spectrum with *the same phase*, their sum yields a signal power of approximately $4P_x$ (why?). On the other hand, the residual aliased power is equal to $\alpha^2 P_x$. The signal-to-noise ratio can therefore be expressed as

$$\text{SNR} \approx \frac{4}{\alpha^2}. \quad (20.12)$$

For example, if $\alpha = 0.1\%$ (roughly commensurate with a resolution of 10 bits), we have $\text{SNR} = 66$ dB.

One can also repeat the analysis for a sinusoidal input, $x(t) = A \cos(2\pi f_{in} t)$, so as to gain more intuition. We begin with Fig. 20.13(a), where the spectra of the input and the impulse train are shown.³ We assume that

³The Fourier transform of the input is given by $(A/2)[\delta(f + f_{in}) + \delta(f - f_{in})]$ (why?).

f_{CK} is slightly greater than f_{in} . The input spectrum thus shifts to the left and to the right by integer multiples of f_{CK} , yielding $Y_1(f)$ and $Y_2(f)$ [Fig. 20.13(b)].⁴ We observe the aliased components at $-f_{CK} + f_{in}$ and $+f_{CK} - f_{in}$. The impulses in $Y_2(f)$ are smaller (or larger) due to the gain error. In this illustration, $Y_1(f)$ represents the Fourier transform of a voltage quantity, and the amplitude of its components is denoted by $A/2$. Those in $Y_2(f)$ are multiplied by $1 + \alpha$. The components at $\pm f_{in}$ add in phase, yielding an amplitude of $(2 + \alpha)A/2$, and those at $\pm(f_{CK} - f_{in})$ partially cancel, producing $+\alpha A/2$. The ratio of the corresponding powers is thus equal to

$$\text{SNR} = \frac{(2 + \alpha)^2 A^2}{\alpha^2 A^2} \approx \frac{4}{\alpha^2}. \quad (20.13)$$

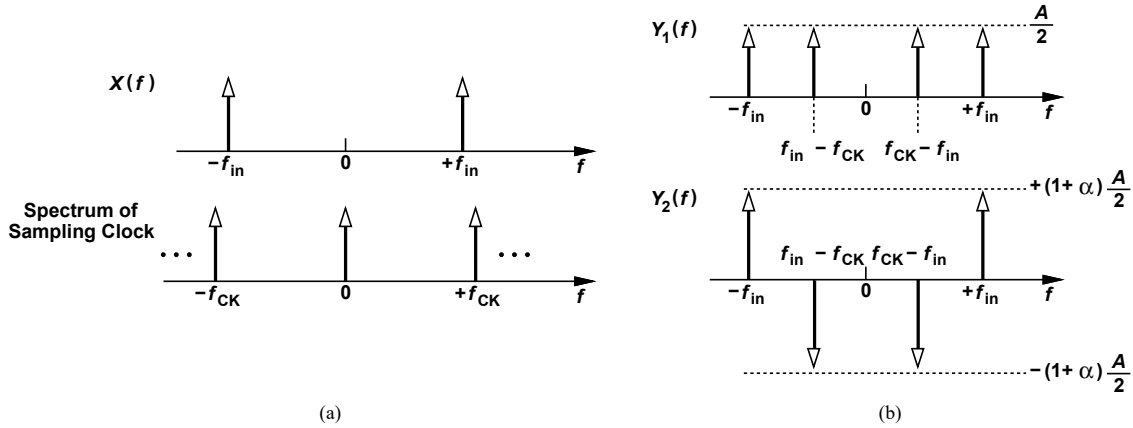


Figure 20.13 Effect of gain mismatch for a sinusoidal input.

Example 20.3

Are the aliased components at $\pm(f_{CK} - f_{in})$ in $Y_1(f)$ in Fig. 20.13(b) in phase with the desired signal?

Solution

Yes, they are. This can be seen by writing

$$y_1(t) = A \cos(2\pi f_{in} t) \sum_n \delta(t - nT_{CK}) \quad (20.14)$$

and hence

$$Y_1(f) = \frac{A}{2} [\delta(f + f_{in}) + \delta(f - f_{in})] * \left[\frac{1}{T_{CK}} \sum_n \delta(f - nf_{CK}) \right]. \quad (20.15)$$

We recognize that all of the impulses have the same phase and the convolution operation does not alter this phase.

20.2.3 Phase Mismatch

Also known as “timing mismatch” or “timing skew,” this phenomenon occurs because the sampling clock edges in different interleaved channels do not have exactly the desired phase difference. Consider, for example, a two-channel system wherein the clock duty cycle slightly deviates from a nominal value of 50%

⁴All of the components in $Y_1(f)$ and $Y_2(f)$ are divided by T_{CK} , which we disregard for simplicity.

(Fig. 20.14).⁵ Assuming that the odd and even samples are taken on the rising and falling edges of CK , respectively, we observe that the even samples are shifted from their ideal instances by $\Delta T = t_2 - t_1$.

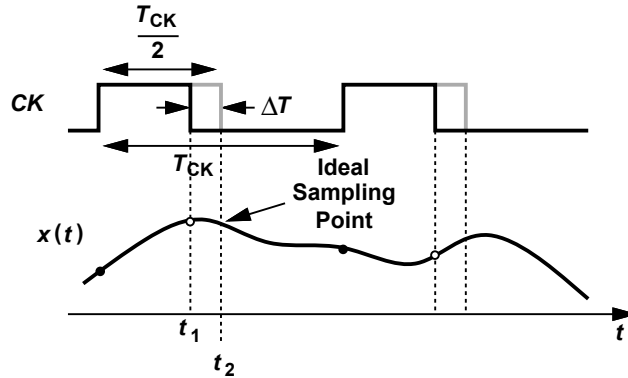


Figure 20.14 Illustration of phase mismatch.

It is instructive to identify the circuit blocks that contribute to phase mismatch. Shown in Fig. 20.15 is a four-phase clock generation and distribution network consisting of an oscillator, a $\div 2$ stage that delivers overlapping quadrature (I and Q) phases, buffers, long interconnects distributing these phases across the chip, a nonoverlapping clock generator, and a bootstrapped sampling switch. We recognize that all of the devices from the $\div 2$ circuit to the sampling action of M_1 introduce timing mismatches among the channels. How

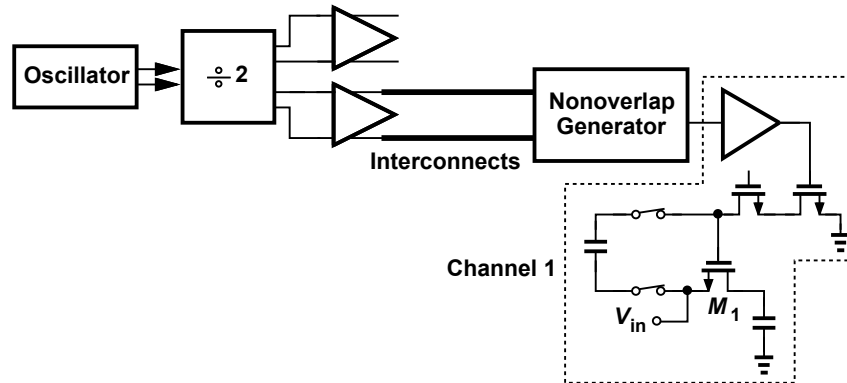


Figure 20.15 Sources of phase mismatch in a four-channel system.

does phase mismatch corrupt the signal? The digital (discrete-time) processor following the ADC knows nothing about the mismatch and assumes that all data samples arrive at equal time intervals. That is, it thinks that the ideal sample at $t = t_2$ in Fig. 20.14 occurs at $t = t_1$. The resulting amplitude error for even samples in Fig. 20.14 is computed from Fig. 20.16:

$$\Delta x \approx \Delta T \frac{dx}{dt}. \quad (20.16)$$

Alternatively, we can write

$$x(t + \Delta T) \approx x(t) + \Delta T \frac{dx}{dt}. \quad (20.17)$$

We make three remarks. First, we pretend that the sample occurs at the ideal instant, $t = t_2$, but with a signal-dependent amplitude error. Second, the mismatch ΔT varies randomly from one chip to another but not with

⁵The “duty cycle” is defined as the time during which the clock is high divided by the period.

time. That is, for a given ADC chip, ΔT is constant and must not be confused with clock jitter (Chapter 3). Third, the error is proportional to the derivative of $x(t)$. Let us return to the time-domain equations for $y_1(t)$

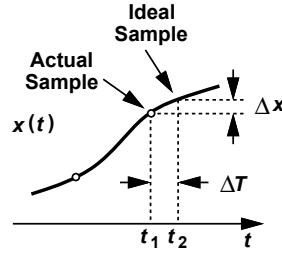


Figure 20.16 Amplitude error due to phase mismatch.

and $y_2(t)$ in Section 20.1.1 and see how they change in the presence of phase mismatch. Equation (20.1) remains the same but Eq. (20.2) is revised as follows:

$$y_2(t) = x(t) \sum_n \delta \left(t - nT_{CK} - \frac{T_{CK}}{2} - \Delta T \right) \quad (20.18)$$

$$= \sum_n x \left(nT_{CK} + \frac{T_{CK}}{2} + \Delta T \right) \delta \left(t - nT_{CK} - \frac{T_{CK}}{2} - \Delta T \right), \quad (20.19)$$

where we have used the property $x(t)\delta(t - t_a) = x(t_a)\delta(t - t_a)$. From (20.17), we have

$$y_2(t) \approx \sum_n x \left(nT_{CK} + \frac{T_{CK}}{2} \right) \delta \left(t - nT_{CK} - \frac{T_{CK}}{2} \right) \quad (20.20)$$

$$+ \sum_n \Delta T x' \left(nT_{CK} + \frac{T_{CK}}{2} \right) \delta \left(t - nT_{CK} - \frac{T_{CK}}{2} \right), \quad (20.21)$$

where $x'(\cdot) = dx/dt$. Note again that the impulses now appear at their ideal time instants. The second summation in (20.21) represents the error: the *derivative* of the signal is sampled by the even channel and scaled by a factor of ΔT . We denote this summation by $y_e(t)$ and study it for sinusoidal or random inputs.

Case of Sinusoidal Input To understand the nature of the corruption, we first assume a sinusoidal input, $x(t) = A \cos \omega_{in} t$, where $\omega_{in} = 2\pi f_{in}$. We have $\Delta T dx/dt = -A\Delta T \omega_{in} \sin \omega_{in} t$, obtaining the corresponding spectrum shown in Fig. 20.17(a).⁶ This spectrum is now shifted to the right and to the left by integer multiples of f_{CK} and produces the spurs depicted in Fig. 20.17(b).

These results merit a few remarks. As expected, the amplitude of the spurs is proportional to ΔT , but also proportional to f_{in} . Thus, the spurs are greatest if the ADC operates near the Nyquist rate. (More fundamentally, the spurs are proportional to the maximum input slew rate, $2\pi A f_{in}$.) Second, from Eqs. (20.3) and (20.4), the reader can show that the desired impulses in $Y_1(f)$ and $Y_2(f)$ have an area of $(A/2)/T_{CK}$ and add in phase, yielding A/T_{CK} . Each spur exhibits an area of $\pi \Delta T A f_{in}/T_{CK}$. We can therefore define a signal-to-noise ratio reflecting this corruption:

$$\text{SNR} \approx \frac{\left(\frac{A}{T_{CK}} \right)^2}{\left(\frac{\pi \Delta T A f_{in}}{T_{CK}} \right)^2} \quad (20.22)$$

$$\approx \frac{1}{\pi^2 \Delta T^2 f_{in}^2}. \quad (20.23)$$

⁶The Fourier transform of $A \sin(2\pi f_{in} t)$ is given by $(A/2)[+j\delta(f + f_{in}) - j\delta(f - f_{in})]$.

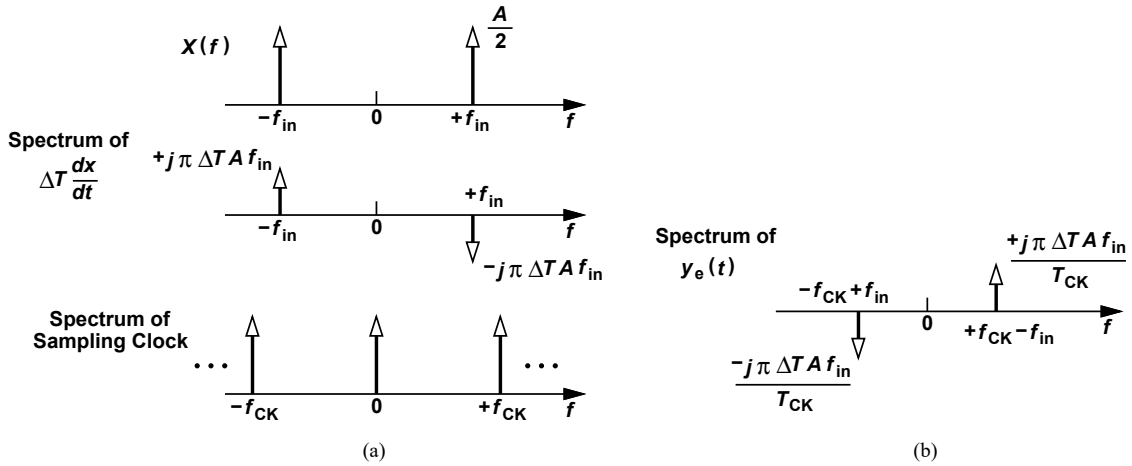


Figure 20.17 (a) Input, clock, and error spectra, and (b) spectrum of error.

For example, a two-channel interleaved ADC digitizing a 5-GHz input provides an SNR less than 60 dB if $\Delta T = 64$ fs, a very tight constraint.

Example 20.4

A student writes $x(t) = A \cos \omega_{in} t$ and $\Delta T dx/dt = -\Delta T A (2\pi f_{in}) \sin \omega_{in} t$. The student then reasons that the signal power is equal to $A^2/2$ and the spur power equal to $[\Delta T A (2\pi f_{in})]^2/2$, concluding that

$$\text{SNR} = \frac{A^2}{\Delta T^2 A^2 (4\pi^2 f_{in}^2)} = \frac{1}{4\pi^2 \Delta T^2 f_{in}^2}. \quad (20.24)$$

Explain the flaw in this calculation.

Solution

This calculation fails to consider that the timing mismatch corrupts only the *even* samples. For this reason, we examined the *output* of the overall ADC and did not refer the amplitude error to the input. If referred to the input, the spur power is equal to $\pi^2 \Delta T^2 A^2 f_{in}^2/2$ so that $\text{SNR} = (\pi^2 \Delta T^2 f_{in}^2)^{-1}$.

Example 20.5

How much phase mismatch is acceptable for a two-channel N -bit ADC?

Solution

The tolerable mismatch must be quantified by an SNR *penalty*. Neglecting other imperfections and assuming $x(t) = A \cos \omega_{in} t$, we write the signal power as $A^2/2$ and the spur power as $\pi^2 \Delta T^2 A^2 f_{in}^2/2$. This value must be added to the quantization noise so as to obtain the overall SNR:

$$\text{SNR} = \frac{A^2/2}{\frac{1}{12} \left(\frac{2A}{2^N} \right)^2 + \frac{\pi^2 \Delta T^2 A^2 f_{in}^2}{2}} \quad (20.25)$$

$$= \frac{1}{\frac{1}{3 \cdot 2^{2N}} + \pi^2 \Delta T^2 f_{in}^2}. \quad (20.26)$$

The reader recognizes the first term in the denominator as the inverse of the ideal SNR for an N -bit ADC. The “penalty” is defined as the ideal SNR divided Eq. (20.26). For example, if a 2-dB penalty can be tolerated

in a 10-bit system, and given that the ideal SNR is equal to $(3/2)2^{2N} = 1.573 \times 10^6$, we write the penalty as

$$10 \log \left[1.573 \times 10^6 \left(\frac{1}{1.573 \times 10^6} + \pi^2 \Delta T^2 f_{in}^2 \right) \right] = 2 \text{ dB}. \quad (20.27)$$

For $f_{in} = 5 \text{ GHz}$, as an example, we have $\Delta T \leq 39 \text{ fs}$.

Example 20.6

We have observed that both gain mismatch and phase mismatch generate spurs at $\pm(f_{CK} - f_{in})$. Can we distinguish between the two effects?

Solution

In principle, yes. We recognize that an input of the form $x(t) = A \cos \omega_{in} t$ leads to (1) gain-mismatch-induced spurs that are *in phase* with $x(t)$, and (2) phase-mismatch-induced spurs that are 90° out of phase with respect to $x(t)$. The latter point can be seen by writing $x(t + \Delta T) \approx x(t) + \Delta T dx/dt$, which is equal to $x(t) - \Delta T A \omega_{in} \sin \omega_{in} t$. Figure 20.18 illustrates the two sets of spurs in the frequency domain,⁷ emphasizing that the differentiation phenomenon associated with phase mismatch multiplies the spurs by $\pm j$ or, equivalently, rotates them by $\pm 90^\circ$. We can thus distinguish between the two if we examine the *phase* of the fast Fourier transform (FFT) of the output. As explained in Problem 20.6, we can also examine the spur amplitude as a function of the input frequency.

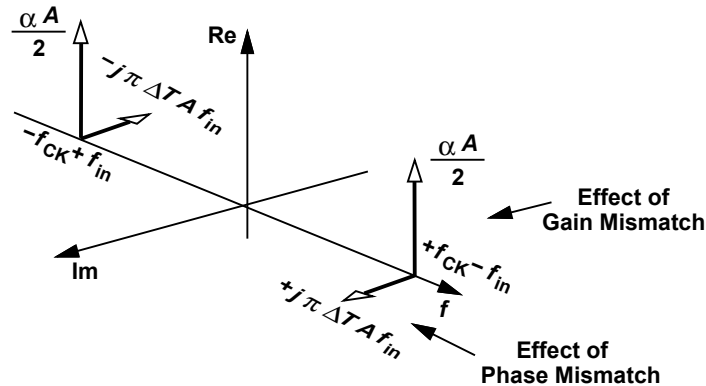


Figure 20.18 Illustration of tones due to gain and phase mismatches.

Case of Random Input Additional insights are obtained by studying the effect of phase mismatch on a random signal. For simplicity, we assume $x(t)$ displays a flat spectrum from $-f_{in}$ to $+f_{in}$ [Fig. 20.19(a)]. While our previous analyses have treated the “spectrum” and the Fourier transform interchangeably (which is not always correct), we must be more careful in this case. That is, $S_x(f)$ in Fig. 20.19(a) represents the actual spectrum (also called the “power spectral density”) of $x(t)$. From the approximation $x(t + \Delta T) \approx x(t) + \Delta T dx/dt$, we know that the corruption is proportional to the derivative of the signal. We must then decide how $\Delta T dx/dt$ affects the spectrum. We recognize that a linear system having a transfer function $H(s)$ and receiving $S_x(f)$ yields an output spectrum, $S_y(f)$, given by

$$S_y(f) = |H(s = j2\pi f)|^2 S_x(f). \quad (20.28)$$

⁷Note that all of these components must be divided by T_{CK} , which we disregard for simplicity.

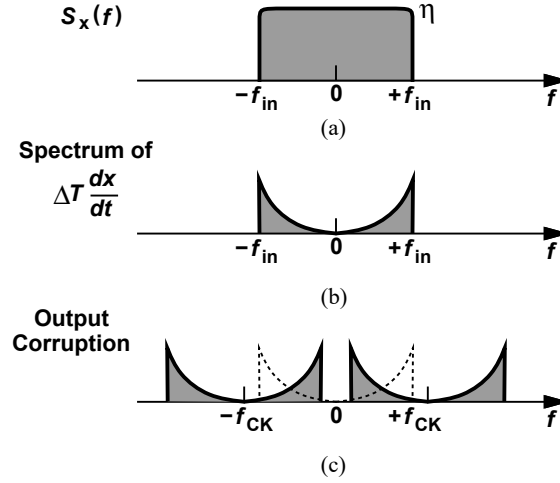


Figure 20.19 (a) Random input spectrum, (b) error spectrum, and (c) corruption due to phase mismatch.

For a differentiator, $H(s) = s$ and hence $\Delta T dx/dt$ “shapes” $S_x(f)$ as follows:

$$S_y(f) = |\Delta T j 2\pi f|^2 S_x(f) \quad (20.29)$$

$$= 4\Delta T^2 \pi^2 f^2 S_x(f). \quad (20.30)$$

That is, $S_x(f)$ is multiplied by a parabola [Fig. 20.19(b)]. Upon sampling, this spectrum shifts to the right and to the left [Fig. 20.19(c)]. In summary, we observe *two* corruption mechanisms here: (1) the shaped spectrum, $4\pi^2 f^2 \Delta T^2 S_x(f)$, simply adds to the desired input, and (2) the aliased copies of this spectrum also add to the signal. As shown in [2], the SNR in this case is given by

$$\text{SNR} = \frac{3}{2\pi^2 \Delta T^2 f_{in}^2}. \quad (20.31)$$

This result is $10 \log(3/2) = 1.76$ dB higher than that predicted by (20.23).

20.2.4 Bandwidth Mismatch

The analog signal paths in interleaved samplers can exhibit bandwidth mismatches. We show here that this imperfection can be lumped with the mismatches among the clock phases. Consider the two interleaved sampling circuits shown in Fig. 20.20, where ΔR denotes the switch resistance mismatch and leads to a

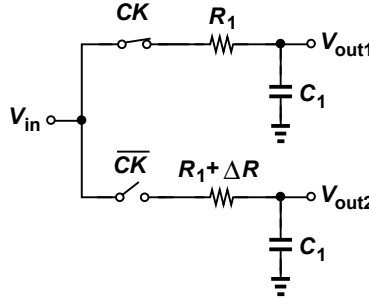


Figure 20.20 Interleaved samplers for bandwidth mismatch analysis.

bandwidth mismatch of $\Delta\omega = (R_1 C_1)^{-1} - [(R_1 + \Delta R) C_1]^{-1} \approx (\Delta R / R_1) (R_1 C_1)^{-1}$ (why?). For a sinusoidal input at a frequency of ω_{in} , the phase shifts through the two networks are slightly different, producing

an output mismatch given by

$$\Delta\phi = \tan^{-1} \frac{\omega_{in}}{\omega_0} - \tan^{-1} \frac{\omega_{in}}{\omega_0 - \Delta\omega}, \quad (20.32)$$

where $\omega_0 = 1/(R_1 C_1)$. Noting that $\tan^{-1} a - \tan^{-1} b = \tan^{-1}[(a - b)/(1 + ab)]$, and assuming $\Delta\omega \ll \omega_0$ and $\omega_{in}^2/\omega_0^2 \ll 1$, we have

$$|\Delta\phi| \approx \tan^{-1} \left(\frac{\Delta\omega}{\omega_0^2} \omega_{in} \right) \quad (20.33)$$

$$\approx \frac{\Delta\omega}{\omega_0^2} \omega_{in}. \quad (20.34)$$

This error can be viewed as a *time* shift if it is divided by 2π and multiplied by $T_{in} = 2\pi/\omega_{in}$:

$$\Delta T \approx \frac{\Delta\omega}{\omega_0^2} \quad (20.35)$$

$$\approx \Delta R \cdot C_1. \quad (20.36)$$

That is, the bandwidth mismatch translates to delay mismatch, as if the signal were sampled ΔT seconds later than desired in the second channel.⁸

20.3 Charge Sharing and Interchannel Crosstalk

Consider the four-channel front end shown in Fig. 20.21(a), where R_{buf} models the output resistance of the preceding buffer, and the clocks have an overlap time equal to $T_{CK}/4$. Capacitor C_1 begins to acquire V_{in} at $t = t_1$, but C_2 also switches into node X at $t = t_2$. That is, the sampling action on C_1 is disturbed from t_2 to t_3 , especially if C_2 inherits a voltage from the previous cycle that is far from the present value of V_{in} . Figure 20.21(b) illustrates how the initial voltage on C_2 at t_2 can prolong the settling of V_1 . For V_1 to represent V_{in} faithfully, the initial voltage on C_2 must vanish in $T_{CK}/4$ seconds. That is, it appears that C_1 has only this much time for acquiring V_{in} even though the overlap scheme allows $T_{CK}/2$ for sampling. Does this “interchannel crosstalk” mean that overlapping phases provide no advantage compared to nonoverlapping clocks?! The following examples delve into this point.

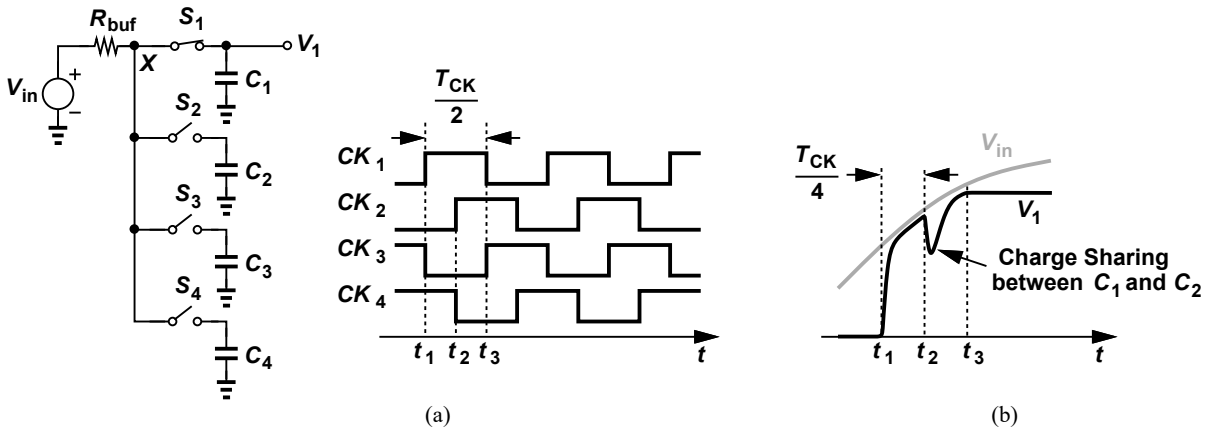


Figure 20.21 (a) A four-channel system, (b) effect of charge sharing.

⁸There is also a gain mismatch, but its effect is negligible.

Example 20.7

Consider the situation shown in Fig. 20.22, where C_1 and C_2 begin with zero initial conditions. Determine the acquisition error, ΔV_1 , at $t = t_3$. Neglect the on-resistance of the switches and assume $C_1 = C_2 = C$.

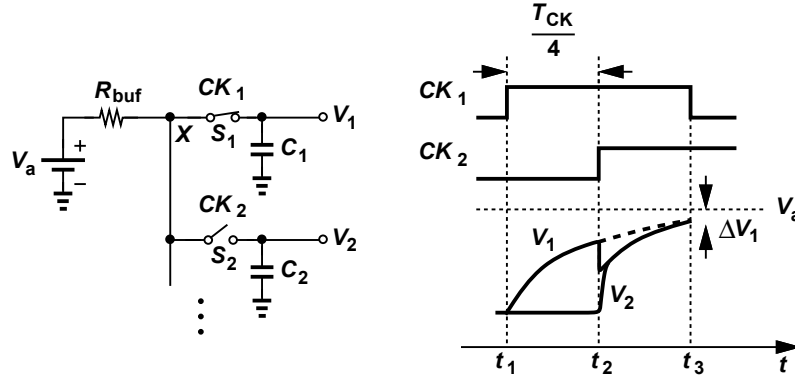


Figure 20.22 Effect of charge sharing on acquisition time.

Solution

From t_1 to t_2 , V_1 grows as $V_1 = V_a[1 - \exp(-t/\tau_1)]$, where $\tau_1 = R_{buf}C$. Thus,

$$V_1(t_2^-) = V_a \left(1 - \exp \frac{-T_{CK}}{4\tau_1} \right), \quad (20.37)$$

where t_2^- denotes the time immediately before t_2 . When S_2 turns on, charge sharing occurs between C_1 and C_2 , causing V_1 to fall by a factor of 2: $V_1(t_2^+) = V_1(t_2^-)/2$. Thereafter, the circuit reduces to a single RC section with $\tau_2 = 2R_{buf}C$. The charging action of C_1 and C_2 continues for $T_{CK}/4$ seconds, yielding:

$$V_1(t_3) = V_1(\infty) + [V_1(t_2^+) - V_1(\infty)] \exp \frac{-T_{CK}}{4\tau_2} \quad (20.38)$$

$$= V_a + \left[\frac{1}{2}V_a \left(1 - \exp \frac{-T_{CK}}{4\tau_1} \right) - V_a \right] \exp \frac{-T_{CK}}{4\tau_2}. \quad (20.39)$$

The settling error, ΔV_1 , is equal to $V_a - V_1(t_3)$:

$$\Delta V_1 = \frac{V_a}{2} \left(1 + \exp \frac{-T_{CK}}{4\tau_1} \right) \exp \frac{-T_{CK}}{4\tau_2}. \quad (20.40)$$

How does this error compare to that in a front end using *nonoverlapping* clocks? In such a case, the capacitor voltage is given by $V_a - V_a \exp[-T_{CK}/(4\tau_1)]$ at the end of the sampling phase, yielding an acquisition error of $\Delta V_2 = V_a \exp[-T_{CK}/(4\tau_1)]$. We then ask whether $\Delta V_1 > \Delta V_2$ or $\Delta V_1 < \Delta V_2$. Denoting $\exp[-T_{CK}/(4\tau_1)]$ by β and recognizing that $\tau_2 = 2\tau_1$, we write $\Delta V_1 = (V_a/2)(1 + \beta)\sqrt{\beta}$ and $\Delta V_2 = V_a\beta$. The reader can show that $\Delta V_1 > \Delta V_2$. Thus, overlapping phases do not improve the acquisition in this case!

The conclusion drawn by the above example goes against our original intuition that overlapping clocks would ease the sampling process. This calculation, however, is generally inaccurate because it neglects the on-resistance, R_{on} , of the switches. In the other extreme, if $R_{buf} \ll R_{on}$, we expect that the disturbance created by C_2 in Fig. 20.22 has only a slight effect on V_1 . (In the limit, $R_{buf} = 0$ and C_2 does not disturb V_1 .) Thus, overlapping phases prove useful only if $R_{buf} \ll R_{on}$. We describe a method of alleviating this effect in Section 20.4.2.

20.4 Interleaving Techniques

We have seen in Section 20.1.1 that interleaved samplers can be driven by overlapping or nonoverlapping clocks, facing a trade-off between the input capacitance and the acquisition time. We also learned in Section 20.2.3 that phase mismatches among the sampling clocks corrupt the signal. A number of techniques have been devised to alleviate these issues.

20.4.1 Two-Rank Interleaving

It is possible to employ a single front-end sampler that acquires the input at full rate⁹ and feeds its output to secondary interleaved channels. Illustrated in Fig. 20.23(a), such a “two-rank” arrangement samples V_{in} on C_F by means of the full-rate clock, CK_F , buffers the result, and applies it to the interleaved samplers. The latter *must* operate with nonoverlapping clocks because C_F holds the input for only half of the full-rate clock period. As shown in Fig. 20.23(b), CK_1 acquires the voltage held on C_F from t_1 to t_2 , CK_2 does the same from t_3 to t_4 , etc. We recognize that this approach eliminates the problem of phase mismatch as the secondary samplers sense *constant* voltages.¹⁰

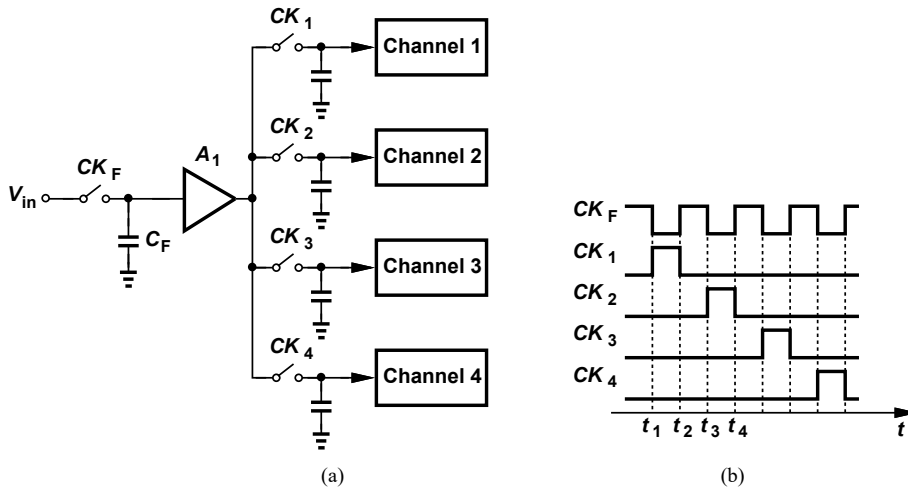


Figure 20.23 (a) Two-rank sampling, and (b) its clock waveforms.

The topology of Fig. 20.23(a) faces three drawbacks. First, buffer A_1 presents severe challenges: its linearity, speed, and noise must be commensurate with the overall ADC performance. For example, a source follower appears a natural choice here [Fig. 20.24(a)], but it (a) creates a level shift equal to V_{GS1} , limiting the voltage headroom for the secondary samplers, (b) suffers from a trade-off between the settling speed and power consumption,¹¹ (c) introduces noise of both M_1 and I_1 in the signal path, and (d) incurs nonlinearity even in the absence of body effect. The last point is evident from the I_D - V_{DS} characteristic of short-channel MOSFETs [Fig. 20.24(b)]: the slope *varies* in the saturation region, causing r_O to depend on V_{DS} . Since the follower’s gain without body effect is equal to

$$A_1 = \frac{r_{O1}}{r_{O1} + \frac{1}{g_{m1}}}, \quad (20.41)$$

⁹The term “full rate” refers to the overall ADC sampling frequency, f_S .

¹⁰We assume that the output of A_1 settles before the falling edge of CK_1 , etc.

¹¹A particularly acute issue when V_{out} must fall and the slew rate is limited to I_1/C_L .

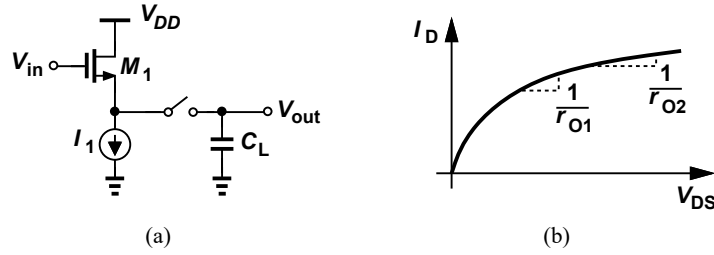


Figure 20.24 (a) Source follower driving a sampling capacitor, and (b) illustration of transistor's nonlinear output resistance.

we observe that A_1 depends on V_{DS} and hence produces nonlinearity. Of course, the dynamic current drawn by C_L also causes r_{O1} and g_{m1} to vary.

The second drawback of the two-rank sampling scheme of Fig. 20.23(a) is that the signal path accumulates kT/C noise twice, a serious issue if $A_1 \leq 1$. Thus, all of the sampling capacitors must be chosen greater than those in single-rank interleaved systems.

The third drawback is that the ADC now requires both a full-rate clock and, for example, quarter-rate clocks. Clock generation thus draws more power in this case.

Example 20.8

Is it possible to omit the buffer in Fig. 20.23(a)?

Solution

For low resolutions, it is [3]. As shown in Fig. 20.25, after V_{in} is sampled on C_F , switch S_1 in the first channel turns on. Now, C_F and C_1 share charge, yielding $V_1 = [C_F/(C_1 + C_F)]V_{in}$ for a zero initial condition on C_1 . The signal amplitude reaching the channels is thus attenuated. In the 6-bit ADC described by [3], $C_F = 100$ fF and $C_1 = 25$ fF, introducing a 20% swing reduction. For higher resolutions, we must also consider the junction capacitance nonlinearity at node X and kT/C noise.

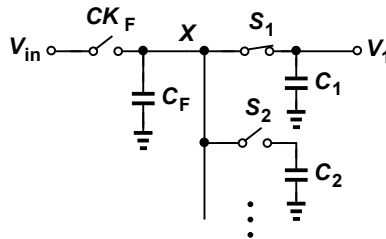


Figure 20.25 Two-rank sampling with no buffer.

20.4.2 Reduction of Interchannel Crosstalk

As explained in Section 20.3, interchannel crosstalk generally proves serious if overlapping clocks are used and a buffer precedes the ADC. We also note that this effect vanishes if only two clock phases, CK and $\overline{CK}$, are present because they naturally do not overlap. This observation leads to an interleaving scheme free from crosstalk [4].

Let us begin with the arrangement shown in Fig. 20.26(a), where buffer B_0 exhibits an output resistance equal to R_{buf0} . We recognize that C_1 and C_3 acquire with nonoverlapping phases, and so do C_2 and C_4 . We then decompose the input signal path into two, each containing its own buffer and driving nonoverlapping samplers [Fig. 20.26(b)] [4]. Here, no two capacitors are driven by a buffer simultaneously, thus avoiding crosstalk.

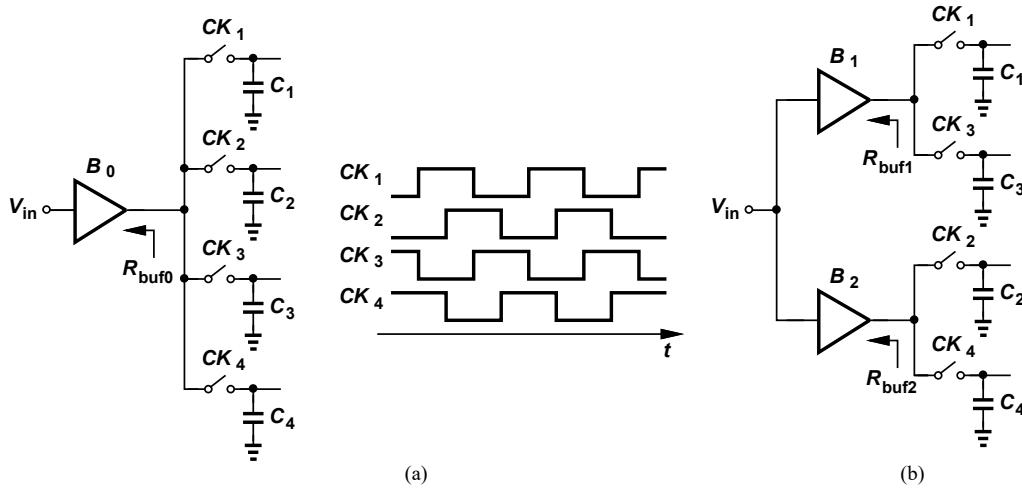


Figure 20.26 (a) Interleaved sampling with overlapping clocks, and (b) use of two buffers to provide isolation.

The topology of Fig. 20.26(b) entails three issues. First, the use of two buffers requires either higher power consumption or longer acquisition. If B_1 and B_2 are identical to B_0 , the power and the input capacitance are doubled. On the other hand, if B_0 is decomposed into two identical “slices,” then $R_{buf1} = R_{buf2} = 2R_{buf0}$, increasing the sampling time constant. For example, the source follower of Fig. 20.24(a) can be transformed to two buffers, each having half the transistor width and biased at $I_1/2$. The reader can readily show that the output resistance doubles.

Second, the offset, gain, and delay mismatches between B_1 and B_2 produce spurs. These mismatches cannot be directly lumped with those of the four interleaved channels because channels 1 and 3 share B_1 and channels 2 and 4 share B_2 . The reader is encouraged to study the effect of these mismatches.

20.5 Interchannel Mismatch Calibration

As we seek higher speeds and resolutions for ADCs, interleaving mismatches become more serious, necessitating calibration. Such a task consists of two operations, namely, detection and correction, each of which can be realized in the analog domain or the digital domain.

20.5.1 General Considerations

Let us first determine in which domain the mismatches should be detected (measured). Considering the generic architecture in Fig. 20.27(a), we note that the offset mismatch introduced by the sampling switches and the quantizers is difficult to sense in the analog domain. On the other hand, the digital outputs plotted in Fig. 20.27(b) reveal that this mismatch leads to unequal *average* values. We can then measure this error by subtracting the long-term averages of D_{out1} and D_{out2} (Section 20.5.3).

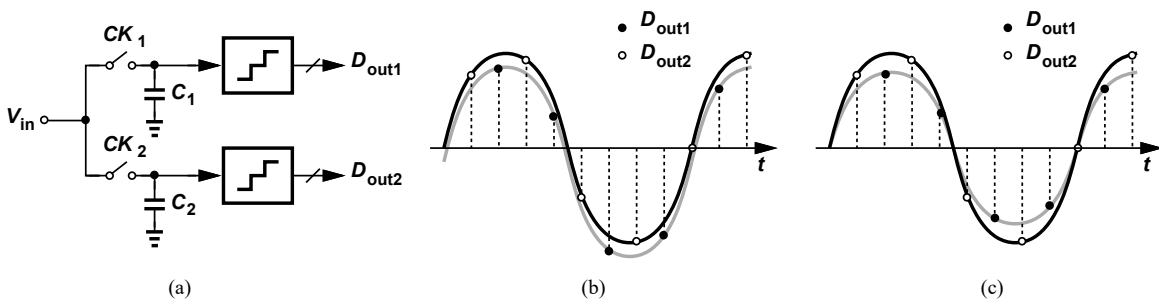


Figure 20.27 (a) Two interleaved channels, (b) effect of offset mismatch, and (c) effect of gain mismatch.

A similar method applies to gain mismatch detection as well. As illustrated in Fig. 20.27(c), we can compute the long-term powers of D_{out1} and D_{out2} and subtract the results (Section 20.5.4).

Example 20.9

Why can we not simply measure the amplitudes of D_{out1} and D_{out2} in Fig. 20.27(c)?

Solution

This is because a random input signal does not yield any “pattern” for the output amplitudes. In other words, if D_{out1} and D_{out2} vary randomly, only their average powers represent their swings meaningfully.

The measurement of clock phase mismatch in the analog domain is difficult. How does one sense this error in the digital domain? This task still proves challenging and is addressed in Section 20.5.5.

We must also ask in which domain the mismatches should be *corrected*. We begin with offset mismatch. Can we introduce offsets in the analog signal paths shown in Fig. 20.27(a)? Yes, we can. Depicted in Fig. 20.28 is an example where the sampling capacitor is decomposed into a fixed value, C_0 , and a simple DAC. By switching the bottom plates of the DAC capacitors from zero to V_{REF} , we can generate an offset at node X . Of course, the DAC must perform switching in every hold mode (why?) according to a fixed code stored in a register. The reader is encouraged to calculate the minimum voltage change created by the DAC at X . The principal advantage of this approach is that it avoids any loss of the ADC dynamic range due to the offset (Fig. 20.10).¹²

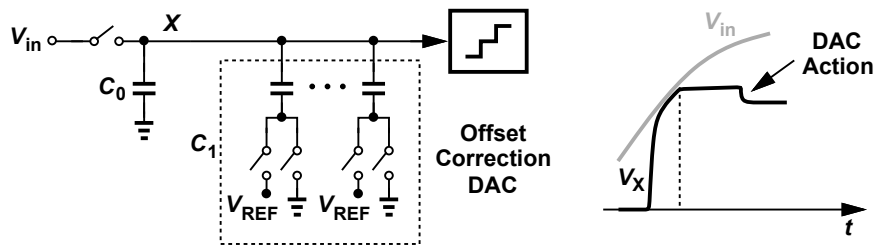


Figure 20.28 Correction of offset by a simple DAC.

Example 20.10

Each channel in an interleaved system incorporates a flip-around MDAC. Can the method of Fig. 20.28 be used in this case?

Solution

Yes, it can, but with a modification. As illustrated in Fig. 20.29, the DAC injects its charge into the virtual ground in the amplification mode, causing V_{res} to change.

Example 20.11

Repeat the previous example if each channel employs a SAR loop rather than an MDAC.

Solution

Figure 20.30 shows a possible arrangement: in addition to the SAR DAC, we have another that corrects the

¹²This point holds if we force the offset of each channel to *zero* rather than make the offsets of different channels equal.

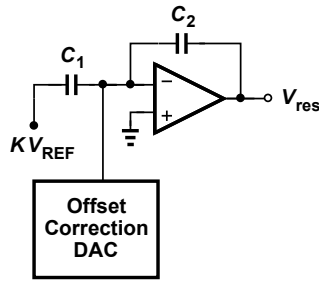


Figure 20.29 Correction of offset in an MDAC.

offset. However, the capacitance presented at X by the offset cancellation DAC raises the gain error (Chapter 15).

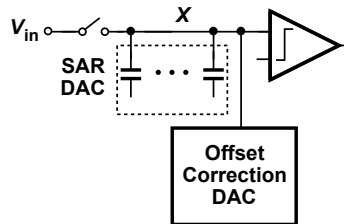


Figure 20.30 Correction of offset in a SAR environment.

If the dynamic range loss due to offsets is negligible, we opt for correction in the digital domain, where the output of each interleaved channel is simply shifted up or down (Section 20.5.3).

The correction of the gain mismatch can occur in the analog domain, but it tends to be somewhat complex. This error is often calibrated digitally. We elaborate on these points in Section 20.5.4.

For clock phase mismatch correction, we first consider the analog domain. As conceptually illustrated in Fig. 20.31, we can adjust the phase (delay) of each clock by means of a capacitor array. Though simple, this approach must still provide a sufficient *range* for the delay, i.e., it must (a) be able to cancel the *largest* expected mismatch, and (b) provide enough resolution so that the residual mismatch is acceptably small. The following example elaborates on these points.

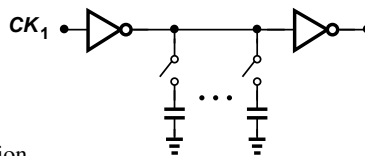


Figure 20.31 Simple clock phase correction.

Example 20.12

The 10-bit ADC studied in Example 20.5 dictates a clock phase mismatch of about 40 fs. If the uncorrected mismatch in a design is 2 ps, determine the delay adjustment resolution in Fig. 20.31.

Solution

We must reduce the 2-ps mismatch to 40 fs. If the clock delay in Fig. 20.31 is linearly proportional to the programmable capacitance, we require 50 units, i.e., slightly less than 6 bits of resolution for delay adjustment.

Dealing with phase mismatch in the digital domain is far more complex. Suppose two interleaved outputs, D_{out1} and D_{out2} , must undergo this type of correction. That is, one must be shifted by, say, 2 ps with respect to the other. How is that possible if each channel is clocked at, for example, 5 GHz? Considering the digital output sequence shown in Fig. 20.32, we wish to create a delayed version, $D_{out1,\Delta}$. Our task is therefore to compute $D_{out1,\Delta}$ from D_{out1} (and possibly the other channels' outputs). We know from our phase mismatch

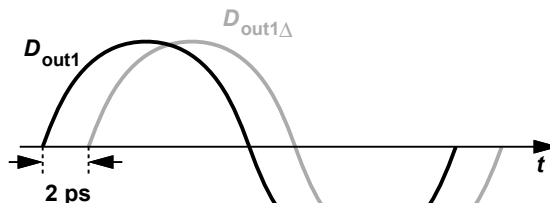


Figure 20.32 Digital outputs exhibiting timing mismatch.

analysis in Section 20.2.3 that these two are related by the slope of the input waveform. Unfortunately, the digital outputs of the interleaved system do not provide this slope information if the overall ADC runs at the Nyquist rate.¹³ It is possible to constrain the input bandwidth to slightly less than this rate (i.e., $f_{in} < f_{CK}$ for a two-channel converter) and employ a digital finite-impulse response (FIR) filter to generate $D_{out1,\Delta}$ [5]. The filter, however, typically requires a large number of taps, thus consuming a high power as it processes high-speed data. For this reason, correction in the analog domain is more desirable.

20.5.2 Foreground and Background Calibration

A system can perform calibration once after it is powered up and freeze the results in registers with no “update” thereafter. For example, using the techniques described in Chapter 6, we can cancel the offset of a comparator after power is applied and store the code in a register, making no changes afterwards. This procedure is called “foreground” calibration. Typically simpler, such an approach does not correct for drifts in the supply voltage or the temperature. For example, the delay mismatch between two CMOS inverters exhibits some temperature dependence, introducing an error after foreground calibration is completed. We surmise that one can interrupt the system periodically and repeat the calibration, but in many applications this is not possible.

“Background” calibration, on the other hand, runs continually (in the background) and updates the registers if supply and temperature variations affect the parameter of interest. This method requires that we *measure* this parameter without interrupting the system’s normal operation. For example, we must determine the dc offset of an ADC in the presence of a random input signal. This procedure therefore is generally more complex than foreground calibration.

Background calibration merits two remarks. First, the updates that it provides need not arrive at a high rate. If, for example, temperature drifts occur with a millisecond time constant, we simply measure the error of interest and update it every 100 μ s. This “compute-and-rest” method reduces the power consumed by the digital calibration engine.

Second, background calibration may fail if the input signal is absent or if it has an arbitrary spectrum or arbitrary amplitude statistics. We elaborate on this point in the subsequent sections but we must generally examine the calibration behavior for different types of signals.

20.5.3 Offset Calibration

As explained in Section 20.3.1, we prefer to detect offsets and their mismatches in the digital domain. Before delving into this matter, let us study the following example.

¹³We state without proof that Nyquist-rate operation produces consecutive samples that have zero correlation.

Example 20.13

An analog integrator senses an input of the form $V_0 \cos \omega_{in} t + V_{OS}$, where V_{OS} is an undesirable offset. Compute the integrator output, $V_{out}(t)$. Assume a zero initial condition.

Solution

We have

$$V_{out}(t) = \frac{V_0}{\omega_{in}} \sin \omega_{in} t + V_{OS} \times t, \quad (20.42)$$

observing that, given enough time, the integrated offset assumes an arbitrarily large magnitude and prevails over the integrated signal.¹⁴ That is, integration can reveal the offset.

To compute the long-term averages of the channels' output, we employ digital "integrators," also called "accumulators." Shown in Fig. 20.33(a) for two channels, each accumulator consists of a register placed in a positive-feedback loop. Suppose D_{out1} is constant and equal to $\alpha > 0$ and the corresponding register begins with a zero value. The register's content then increments by α every clock cycle, rising linearly with time [Fig. 20.33(b)]. As predicted by (20.42), this quantity can overwhelm the integrated signal, thereby serving as a measure of the offset.

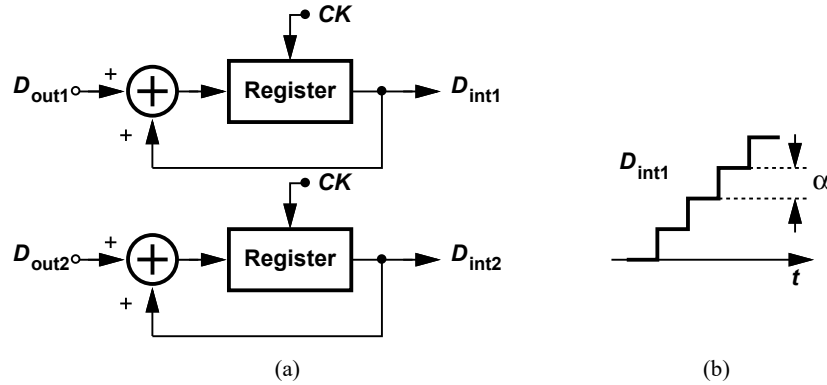


Figure 20.33 (a) Computation of output averages using accumulators, and (b) accumulator waveform example.

How do we utilize the integrated offsets generated in Fig. 20.33(a)? We simply use each channel's result to cancel its own offset. This is accomplished as depicted in Fig. 20.34, where D_{int1} is subtracted from D_{out1} to yield the final output, D_{Y1} [6]. We recognize that D_{int1} is bounded only if the offset at the integrator input is

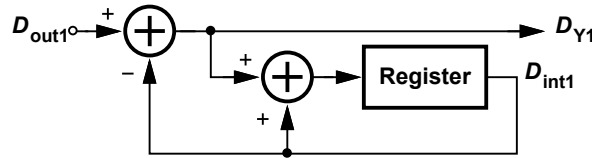


Figure 20.34 Correction of offset in digital domain.

zero. Thus, D_{int1} adjusts itself so that $D_{out1} - D_{int1}$ guarantees a zero offset. This topology performs background calibration, raising concern as to whether it converges for different types of signals. More specifically, we ask, does the feedback loop in Fig. 20.34 also cancel low-frequency *signal* components? The following example illustrates this point.

¹⁴Note that the dimension of the right-hand side in (20.42) is incorrect because we have assumed a unity coefficient for the integrator.

Example 20.14

The foregoing offset calibration technique assumes that V_{Ost} in Eq. (20.42) remains much greater than the integrated signal, $(V_0/\omega_{in}) \sin \omega_{in}t$. Is this generally true?

Solution

No, it is not. For a sufficiently low input frequency, V_0/ω_{in} can well exceed V_{Ost} because the total integration time is *finite*. We note that the accumulator's register has a finite length, b , thus accommodating a maximum value of $11 \cdots 1 = 2^b - 1$. From another perspective, a very slowly-varying input can be confused with dc offsets and hence disturb the value of D_{int1} in Fig. 20.34. Fortunately, the spectrum of practical signals does not extend to arbitrarily low frequencies. We must therefore choose the register length according to the period of the *slowest* input component that the ADC digitizes.

20.5.4 Gain Calibration

We have seen in Section 20.2.2 that the gain of each interleaved channel can be determined by computing its average output power, P_{avg} . Recall that P_{avg} of a signal, $x(t)$, is defined as

$$P_{avg} = \frac{1}{T} \int_0^T x^2(t) dt, \quad (20.43)$$

where T is chosen long enough to capture the slowest signal components. (In the exact definition, $T \rightarrow \infty$.) For a digital implementation, we wish to avoid the squaring function as it requires a multiplier. This is possible by recognizing that a full-wave rectifier approximates a squarer [Fig. 20.35(a)]. We thus construct the cascade shown in Fig. 20.35(b), where the rectified signal is integrated by an accumulator.

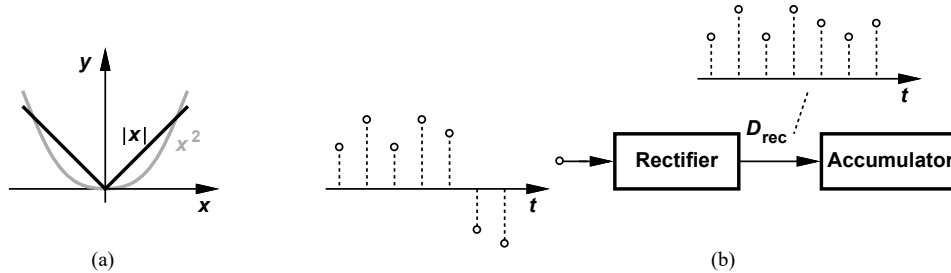


Figure 20.35 (a) Approximation of a parabola by the absolute value function, and (b) use of a rectifier to obtain the absolute value.

Example 20.15

The digital output of typical ADCs does not have negative values. That is, D_{out} simply varies from $00 \cdots 0$ to $11 \cdots 1$ as the differential input goes from its most negative to its most positive extreme. Explain how the rectification of Fig. 20.35(b) must proceed in this case.

Solution

As shown in Fig. 20.36, we compute the average value of the output, subtract it from the signal, and then rectify the result.

We must now determine how the result obtained in Fig. 20.35(b) is utilized to perform gain *correction*. Shown in Fig. 20.37, a two-channel system applies one of the outputs to a variable-gain stage (a multiplier), producing βD_{out1} , where β is a scaling factor. This signal and D_{out2} are rectified, subtracted, and integrated.

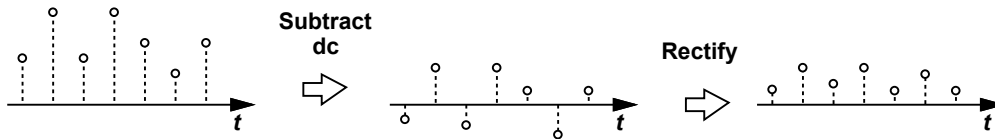


Figure 20.36 Processing of the digital output to obtain the signal average.

In the steady state, the accumulator input must have a zero average so that its output does not go to $\pm\infty$, causing βD_{out1} and D_{out2} to have the same average power. These two signals therefore incur a negligibly small gain mismatch. Note that this is possible only if the dc offset mismatch between the two channels is already removed (why?).

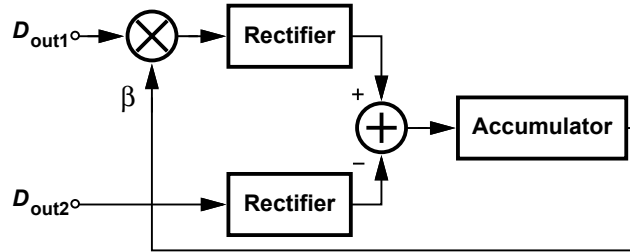


Figure 20.37 Correction of gain error.

It is instructive to study the limitations of this background calibration technique. First, if the ADC receives no input signal for some time, the system in Fig. 20.37 provides no correction and the accumulator can contain any value. If the ADC now begins to receive a signal again, the loop requires a finite time (given by the accumulator's length) to converge, failing to correct the gain error in the meantime. Second, the digital multiplier in Fig. 20.37 can draw substantial power because it processes the high-speed output of one channel. For this reason, one may opt for gain correction in the analog domain. For example, [7] utilizes the gain detection results to drive a DAC that adjusts the gain.

20.5.5 Phase Calibration

The *detection* of phase mismatches (timing skews) among interleaved channels poses the greatest challenge and has been investigated extensively. When comparing these techniques, we must ask the following questions about each. (1) Does it operate properly with *random* signals? (Some methods are tested with only sinusoids.) (2) Does it impose limitations on the input spectrum? (For example, a technique may not converge if part of the spectrum is vacant.) (3) How much high-speed digital computation (specifically, how many adders and multipliers) does it require? (4) Does it employ an extra channel for calibration? (5) Does it add a random signal to the input (thus limiting the dynamic range)?

It is helpful to bear in mind that the purpose of detection is to generate a dc term proportional to the mismatch of interest. We also wish this average to be *zero* for a zero phase error. In this spirit, we surmise that “phase detectors,” commonly used in phase-locked loops and RF applications, can provide such a dc output. For example, a multiplier (mixer) sensing two periodic signals $A_1 \cos \omega_1 t$ and $A_2 \cos(\omega_1 t + \theta)$ produces an output of the form $(A_1 A_2 / 2) \cos(2\omega_1 t + \theta) + (A_1 A_2 / 2) \cos \theta$, where the second term yields the phase error information. We begin with this principle, realize that it fails, and modify our approach until we arrive at a solution.

Let us attempt to multiply the digital outputs of two interleaved channels and view the average of the result as the phase mismatch. Unfortunately, this approach fails, at least in some cases. Suppose the main ADC input is expressed as $x(t) = A \cos(2\pi f_{in} t)$ and recall from Section 20.2.3 that the phase mismatch leads to an error given by $\Delta T dx/dt = -\Delta T A (2\pi f_{in}) \sin(2\pi f_{in} t)$. Upon sampling, the clock translates this component by f_{CK} . We first examine the output spectra of the two channels. The first channel carries components at $\pm f_{in}$,

$+f_{CK} \pm f_{in}$, $-f_{CK} \pm f_{in}$, etc. [Fig. 20.38(a)], all of which have a cosine form and are hence considered to have a real value.¹⁵ The second channel, on the other hand, carries the mismatch-induced sine term as well, whose Fourier transform contains imaginary impulses translated by the clock [Fig. 20.38(b)].¹⁶ That is, the imaginary spurs also land at $\pm(f_{CK} - f_{in})$. We must carry on the analysis one more step to see why this method fails.

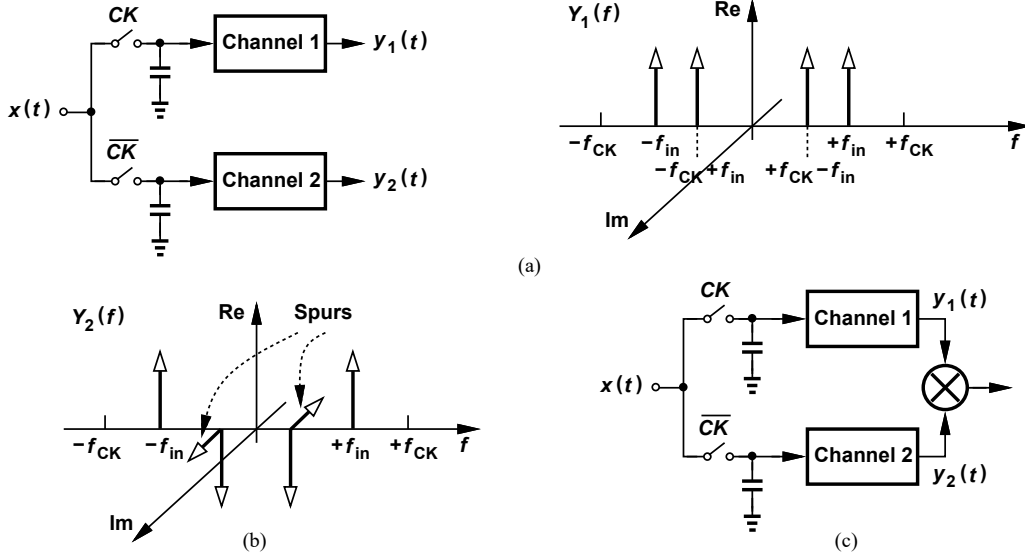


Figure 20.38 (a) Interleaved channels with phase mismatch, and (b) resulting spurs in output spectrum.

We now ask whether multiplying $y_1(t)$ by $y_2(t)$ generates a dc term proportional to ΔT [Fig. 20.38(c)]. The time-domain multiplication equivalently convolves the impulses in Fig. 20.38(a) with those in Fig. 20.38(b). A dc term at the multiplier output would be equivalent to an impulse at $f = 0$, making the impulses at $\pm(f_{CK} - f_{in})$ in Fig. 20.38(a) and the spurs at the same frequencies in Fig. 20.38(b) candidates for this purpose. However, since the former represent cosines and the latter, sines, their time-domain product, $\cos[2\pi(f_{CK} - f_{in})t] \sin[2\pi(f_{CK} - f_{in})t]$, exhibits a zero average. Thus, $y_1(t) \times y_2(t)$ does not yield information regarding the phase mismatch.

We describe one phase calibration technique below and refer the reader to the literature for others [11]–[15]. Let us continue with our multiplication-based method but form *two* products: an odd sample multiplied by the next even sample, and an even sample multiplied by the next odd sample. As illustrated in Fig. 20.39(a), the time difference between an odd sample and the next even sample is, for example, greater than $T_{CK}/2$, thereby skewing the long-term average of their product. Similarly, the time difference between an even sample and the next odd sample skews their product's average in the opposite direction. The difference between the two averages is thus proportional to ΔT . This intuition can also be proved mathematically [2].

In order to implement the foregoing detection method, we recognize from Fig. 20.39(a) that two products of the form $y_2[k-1]y_1[k-1]$ and $y_1[k]y_2[k-1]$ are necessary. The digital realization therefore emerges as shown in Fig. 20.39(b), where the z^{-1} blocks denote a one-cycle delay. Note that $P_{1,2} = y_1[k]y_2[k-1]$, $P_{2,1} = y_2[k-1]y_1[k-1]$, and $D_{\Delta T}$ denotes the average of $P_{1,2} - P_{2,1}$. We study this technique in detail in Problem 20.15.

We test our detection technique for a two-channel ADC by behavioral simulations where the analog input is in the form of white noise but with a bandwidth limited to f_{CK} . Such an input emulates a random signal. Figure 20.40 plots $D_{\Delta T}$ in Fig. 20.39(b) as a function of ΔT , revealing that $D_{\Delta T}$ varies monotonically and

¹⁵The Fourier transform of $A \cos(2\pi f_1 t)$ is equal to $(A/2)\delta(f - f_1) + (A/2)\delta(f + f_1)$.

¹⁶The Fourier transform of $A \sin(2\pi f_1 t)$ is equal to $-j(A/2)\delta(f - f_1) + j(A/2)\delta(f + f_1)$.

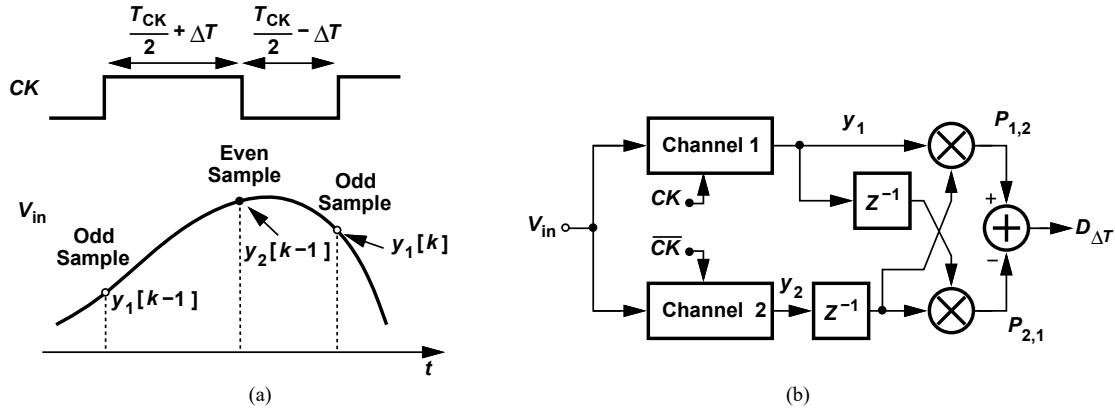


Figure 20.39 (a) Odd and even samples with phase mismatch, and (b) phase mismatch detection.

crosses zero at $\Delta T = 0$. As with other background calibration schemes, this approach fails in the absence of an input signal.

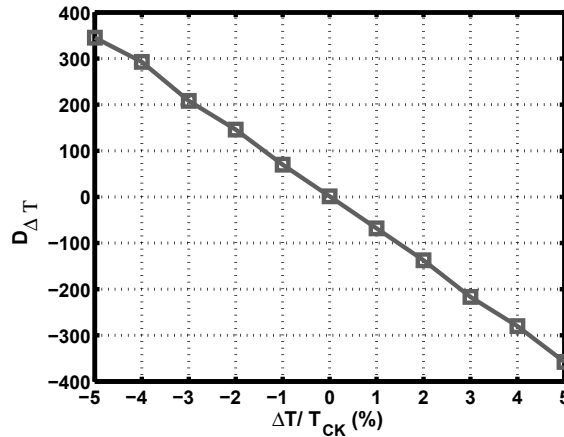


Figure 20.40 Dependence of output upon phase mismatch (arbitrary vertical scale).

The digital hardware in Fig. 20.39(b) can draw high power but it need not run at the full clock rate. That is, the logic can be enabled every $100 \mu s$, for example, so as to provide updates reflecting supply and temperature drifts.

Example 20.16

How do we extend the detection method of Fig. 20.39(b) to four channels?

Solution

We consider the first channel as the “reference” and compute the timing mismatches between Channels 1 and 2, Channels 1 and 3, and Channels 1 and 4. Shown in Fig. 20.41, the system provides three averages that can be used for correction.

Let us now combine this detection technique with the simple analog phase correction scheme shown in Fig. 20.31 so as to form a complete calibration loop. Illustrated in Fig. 20.42, the system stores $D_{\Delta T}$ in a register and adjusts the delay introduced in the path of $\overline{CK}$. The overall loop thus drives the phase mismatch toward zero by virtue of negative feedback. In the steady state, the register and hence the programmable delay [also called a variable-delay line (VDL)] toggle by 1 LSB.

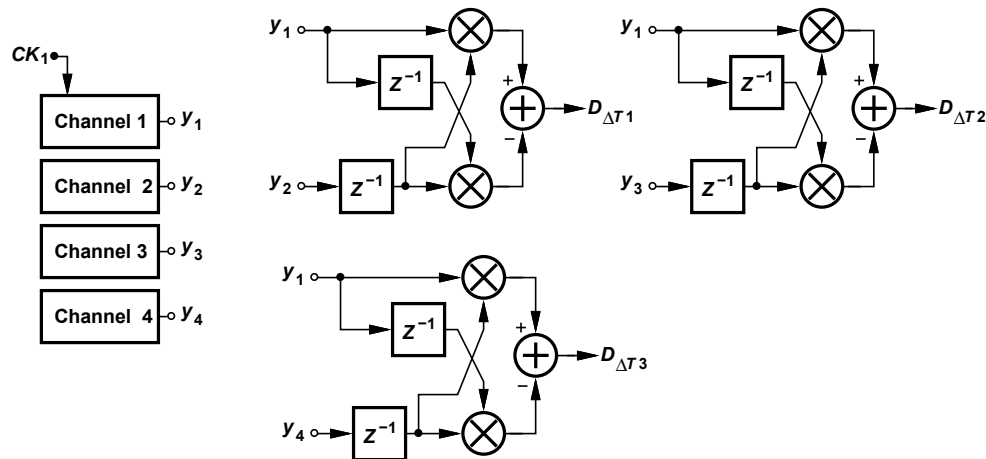


Figure 20.41 Phase error measurement for a four-channel system.

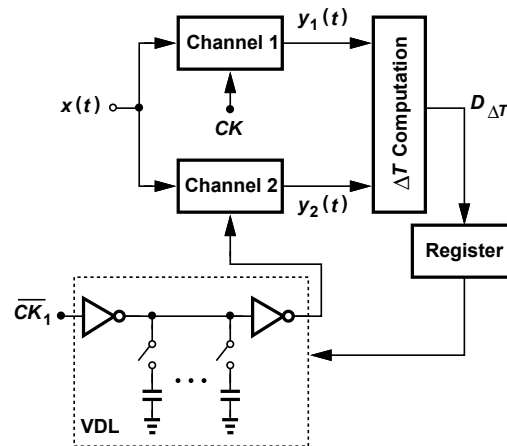


Figure 20.42 Overall phase mismatch detection and correction system.

Example 20.17

How do we extend the loop of Fig. 20.42 to the four-channel system shown in Fig. 20.41?

Solution

Channels 2 to 4 in Fig. 20.41 must incorporate VDLs. First, $D_{\Delta T 1}$ controls the VDL in Channel 2, minimizing the mismatch between the first two channels. Next, $D_{\Delta T 2}$ controls Channel 3, etc. By calibrating Channels 2 to 4 against Channel 1 (rather than Channel 3 against Channel 2, etc.), we avoid accumulation of errors.

Example 20.18

How do we determine the necessary resolution of the VDL in Fig. 20.42?

Solution

After the layout of the channels and their clock distribution network is completed, we extract the parasitics in the clock paths. We then simulate this network and estimate the random and deterministic phase mismatches.

Suppose the maximum error is ΔT_{max} .¹⁷ The total delay correction range must therefore exceed this value. The minimum delay step (the LSB), on the other hand, is dictated by the SNR penalty that we can tolerate (Example 20.5).

The arrangement depicted in Fig. 20.42 faces three issues. First, the VDL provides only “positive” delay, failing to correct “negative” timing mismatches. Second, the VDL creates a certain minimum delay (equal to that of two inverters) even if all of the capacitors are switched out. That is, the VDL introduces its own delay mismatch between CK and $\overline{CK}$. To overcome this difficulty, we turn to “differential” delay control. Illustrated in Fig. 20.43, the idea is to insert VDLs in both CK and $\overline{CK}$ paths and adjust them in *opposite* directions. In the absence of any mismatches, both delays are set to their mid-values (corresponding to a $100 \cdots 0$ code), and to correct mismatches, one increments or the other decrements. Note that differential operation inevitably raises the VDL-induced jitter by a factor of $\sqrt{2}$ because the two VDLs generate uncorrelated jitter.

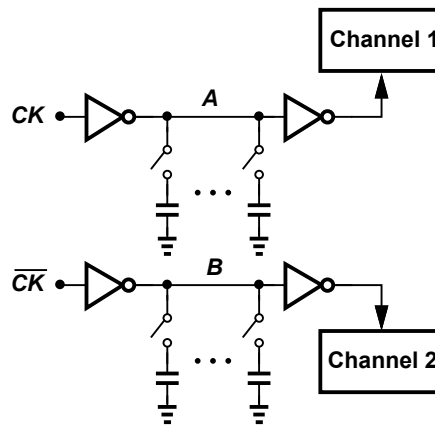


Figure 20.43 Differential adjustment of clock phases.

Example 20.19

How do we incorporate differential delay control in Fig. 20.41?

Solution

We begin with Channels 1 and 2 and calibrate them as in Fig. 20.43. Next, we calibrate Channel 3 against Channel 1 but we adjust only the VDL in Channel 3. A similar procedure is followed for Channel 4. This multistep calibration can proceed in the background after the initial setup.

The third issue relates to the complexity of VDLs. Suppose the maximum delay mismatch is 2 ps and the desired LSB is 20 fs, demanding 100 unit capacitors in each VDL. To appreciate the challenges here, suppose the inverters in Fig. 20.43 incorporate $(W/L)_N = 1 \mu\text{m}/28 \text{ nm}$ and $(W/L)_P = 2 \mu\text{m}/28 \text{ nm}$. The capacitance at A and B is thus around 5 fF before the switches are added and the delay of the first inverter in the chain is about 10 ps. To change this delay by 2 ps, we must add 1 fF of programmable capacitance to these nodes, with an LSB value of $1 \text{ fF}/100 = 10 \text{ aF}$. It is difficult to implement such a small unit capacitor.

Example 20.20

A student reasons that a small metal-9 square, e.g., $0.5 \mu\text{m}$ on each side, exhibits a capacitance on the order

¹⁷For a Gaussian distribution, we consider the 3σ value to be roughly the maximum.

of 10 aF to the substrate. Can we rely on such a structure for a unit capacitor in VDL design?

Solution

The difficulty with small capacitor geometries is that they can incur a substantial error due to their fringe capacitances. Consider the three nominally equal units shown in Fig. 20.44(a), where C_{12} and C_{23} arise from the lateral electric field lines between the metal-9 squares. If only S_1 is on, the capacitance presented to A assumes the equivalent value depicted in Fig. 20.44(b) and is roughly equal to $C_1 + C_{12}$ if the fringe components are much less than the main capacitors.¹⁸ On the other hand, if both S_1 and S_2 are on, the programmable network reduces to that in Fig. 20.44(c), providing an equivalent value of about $C_1 + C_2 + C_{23}$. That is, the total capacitance does not double. In reality, the situation is more complex as the units are arranged in a matrix, thereby incurring fringe components on four sides.

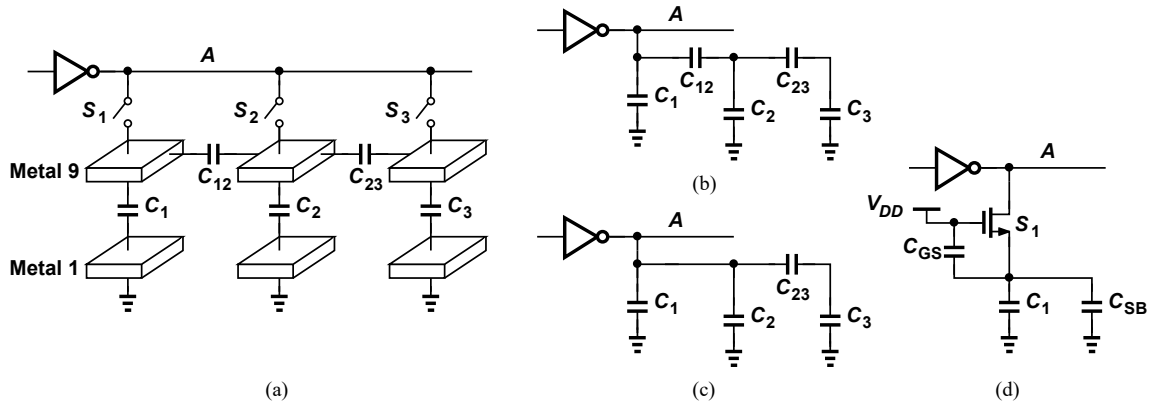


Figure 20.44 (a) Delay control network, (b) equivalent circuit when C_1 is connected to A , (c) equivalent circuit when C_1 and C_2 are connected to A , and (d) effect of switch parasitics.

Another issue here is that the capacitances associated with the switches far exceed 10 aF. For example, when S_1 turns on [Fig. 20.44(d)], its C_{GS} and C_{SB} load node A by several femtofarads, making the LSB much larger than desired.

The foregoing issues motivate us to seek other methods of delay adjustment. We can turn to “starved” inverters instead. As shown in Fig. 20.45(a), one of the transistors in the clock path is degenerated by a variable resistance, M_3 , so that the output falling (or rising) transition time can be adjusted. The delay associated with this edge is therefore controlled. Unfortunately, however, the delay becomes a very nonlinear function of V_{cont} if this voltage is near the threshold of M_3 [Fig. 20.45(b)] and hence M_1 is heavily degenerated. This effect is alleviated by placing a constant resistor in parallel with transistor M_3 , thus keeping M_1 partially on [Fig. 20.45(c)] [8].

We must still address one question: how is V_{cont} generated? This can be accomplished by a resistor-ladder DAC controlled by the register in Fig. 20.42. Illustrated in Fig. 20.45(d), the resulting circuit adjusts the delay monotonically. The ladder end voltages, $V_{DD}/4$ and $3V_{DD}/4$, provide a reasonable range for controlling the strength of M_3 . For our previous example of 100 delay steps, the ladder must incorporate as many resistor segments, but it does not suffer from the inaccuracies described in Example 20.20.

Another method of controlling a starved inverter involves decomposing M_3 in Fig. 20.45(c) into the required number of units and controlling them by a thermometer code. Figure 20.46(a) shows such an arrangement; the degeneration resistance in the source of M_1 monotonically varies from R_{on4} when the thermometer code is equal to zero to $R_{on4} \lfloor (R_u/K) \rfloor$ when it is equal to $11 \cdots 1$.

¹⁸This may not be true in practice.

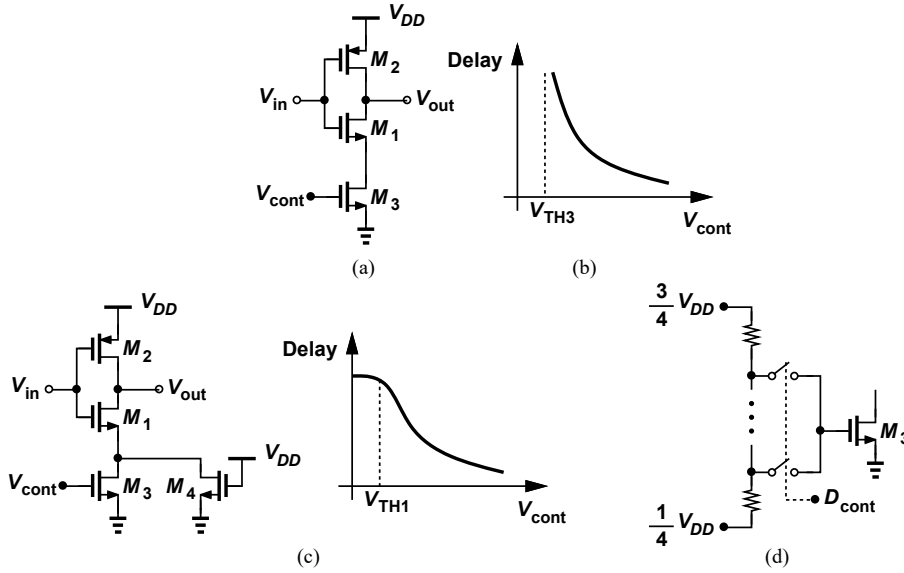


Figure 20.45 (a) Starved inverter, (b) its delay characteristic, (c) addition of M_4 to linearize the delay and the resulting characteristic, and (d) fine delay adjustment by a ladder.

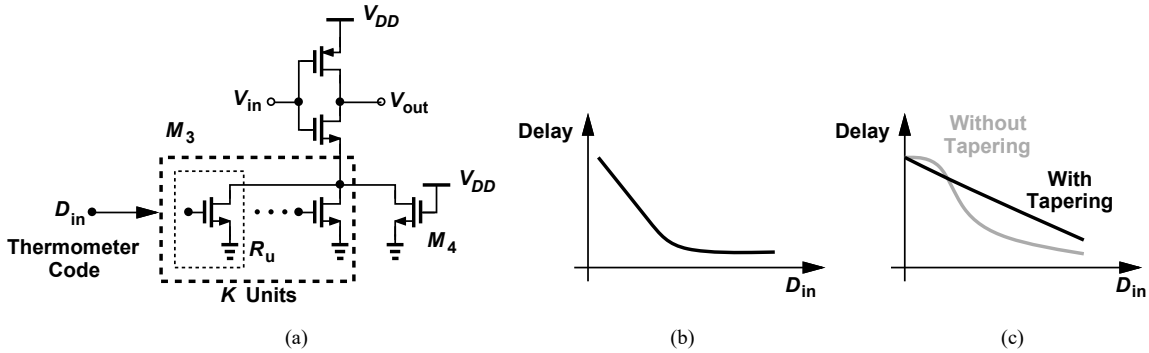


Figure 20.46 (a) Digital control of the delay, (b) corresponding characteristic, and (c) effect of tapering.

The programmable delay scheme of Fig. 20.46(a) entails two issues. First, with a lower bound imposed by the process on the unit transistor width (e.g., $W_{min} \approx 120$ nm in 28-nm technology), the sum of the widths in Fig. 20.46(a) far exceeds W_4 , making the delay only a weak function of D_{in} for higher values of D_{in} [Fig. 20.46(b)] (why?). To make the characteristic more linear, we can select a greater length for the transistors so that their overall equivalent W/L is the same as that of M_4 in Fig. 20.46(b). Alternatively, we can taper the unit transistor lengths down from the LSB side to the MSB side so that, as D_{in} rises, the transistors that are activated are increasingly stronger. As shown in Fig. 20.46(c), tapering leads to a more uniform delay variation.

The second issue related to Fig. 20.46(a) is the complexity associated with the programmable-delay network for 100 delay steps. It is possible to simplify the design by adding a coarse binary-weighted capacitor array. Depicted in Fig. 20.47(a) is an example wherein a 3-bit structure controls the delay of the first stage. The starved inverter need only incorporate about $100/8 \approx 12$ unit transistors. The calibration loop in Fig. 20.42 first adjusts the capacitor array so as to minimize the error and then begins to control the starved inverter. The overall characteristic is shown in Fig. 20.47(b).

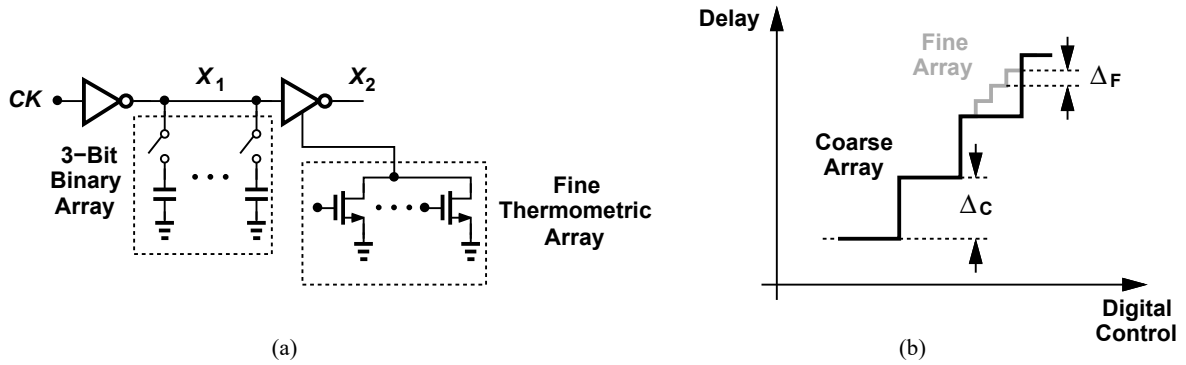


Figure 20.47 (a) Coarse and fine delay control, and (b) its characteristics.

Example 20.21

It is possible that, in Fig. 20.47(b), the LSB delay provided by the coarse array, Δ_c , is *not* equal to eight times that of the fine array, Δ_f . Explain what happens if $\Delta_c > 8\Delta_f$ or $\Delta_c < 8\Delta_f$. Which case is preferable?

Solution

If $\Delta_c > 8\Delta_f$, then the fine array cannot completely span the coarse LSB [Fig. 20.48(a)], leaving a “dead zone” in every segment. That is, the delay range from ΔT_1 to ΔT_2 cannot be generated in fine steps. If $\Delta_c < 8\Delta_f$, then the characteristic becomes nonmonotonic [Fig. 20.48(b)]. This effect is not detrimental here as it simply means that a given delay value, ΔT_1 , can be generated by either D_1 or D_2 . This situation is thus preferable. However, a single negative-feedback loop potentially fails to operate properly in this case because the delay drop at D_0 changes the polarity of the slope and hence the polarity of feedback.

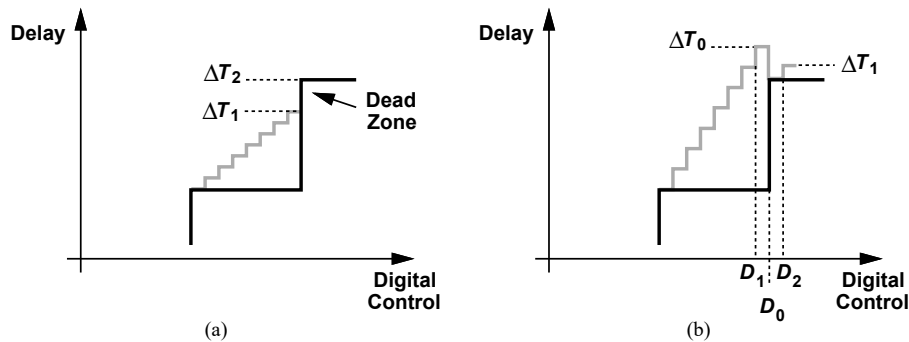


Figure 20.48 Effect of fine steps being (a) less, or (b) greater than nominal.

Nevertheless, we can modify the search algorithm to avoid this issue. Rather than allow the loop in Fig. 20.42 to decide the digital value controlling the delay, we begin with a zero digital input in Fig. 20.48(b) and increment it in steps equal to the fine LSB. After each step, we measure the timing mismatch indicator, $D_{\Delta T}$ in Fig. 20.42, and continue to increment if $D_{\Delta T}$ has not changed its sign. (Recall from Fig. 20.40 that $D_{\Delta T}$ crosses zero at $\Delta T = 0$.) The search stops after this sign change occurs, freezing the data in the register.

The reader may realize that the nonmonotonicity in Fig. 20.48(b) still proves problematic in *background* calibration. Since the feedback loop must now make small adjustments to the delay, it can fail if the digital code and the delay happen to be equal to D_0 and ΔT_0 , respectively. When the loop increments D_{in} by 1 LSB, the delay *falls*, an undesirable phenomenon. For this reason, additional precautions must be taken to ensure that the digital input remains a few LSB away from D_0 .

20.6 Clock Generation for Interleaved ADCs

As explained in Section 20.1.1, interleaving by a factor of M requires M overlapping or nonoverlapping clock phases, where M is typically a power of 2. In this section, we study a number of circuit topologies that serve this purpose. We begin with $M = 4$.

Clock generation can begin with overlapping phases, in which case a chain of $\div 2$ stages proves useful. Let us first consider the arrangement shown in Fig. 20.49, where a flipflop (FF) consisting of two latches is placed in a negative-feedback loop so as to divide the clock frequency by 2. Outputs X_1 and Y_1 change on the rising edge of CK , and X_2 and Y_2 on its falling edge [Fig. 20.49(b)]. We also call X_1 and Y_1 the “in-phase” outputs and X_2 and Y_2 the “quadrature-phase” outputs, sometimes denoting them by I , $\bar{I}$, Q , and $\bar{Q}$, respectively. This circuit provides four overlapping clocks. Note that the frequency of CK is equal to one-half of the overall sampling rate.

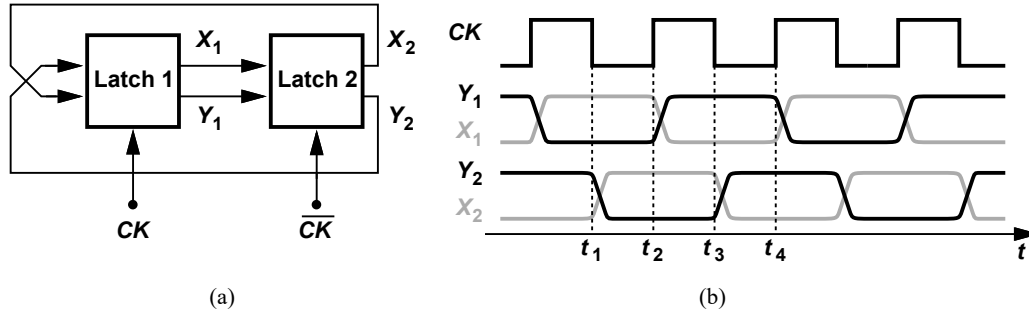


Figure 20.49 (a) Two latches operating as a frequency divider, and (b) their output waveforms.

The latches in Fig. 20.49(a) can be realized by various logic styles. For clock frequencies up to about 30 GHz in 28-nm CMOS technology, clocked CMOS (C^2 MOS) implementations prove feasible. Shown in Fig. 20.50(a) is a C^2 MOS latch, where D_{in} travels to the output when CK is high, and the output is stored on the parasitic capacitance at N when CK goes low. We call this design style “unratioed” logic as the circuit operates correctly (but not necessarily optimally) for any ratio between the NMOS and PMOS strengths. Unfortunately, however, the circuit does not provide the complementary waveforms necessary in Fig. 20.49(b).¹⁹

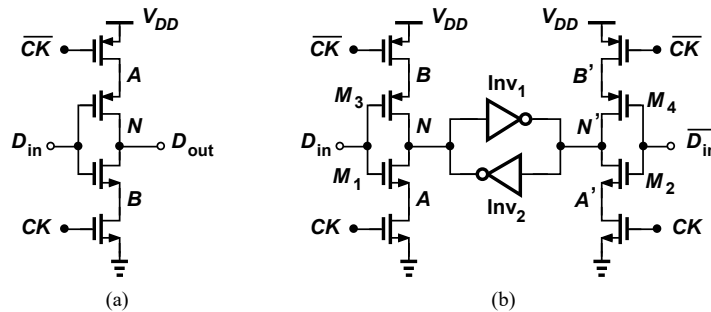


Figure 20.50 (a) C^2 MOS latch, and (b) its complementary form with cross-coupled inverters.

The above C^2 MOS latch can be duplicated to allow differential operation. The result is shown in Fig. 20.50(b); two of these latches can form a $\div 2$ circuit. Inverters Inv_1 and Inv_2 are necessary in a $\div 2$ loop to

¹⁹This topology also suffers (a) from charge sharing between N and A or B , a relaxed issue in a $\div 2$ environment, and (b) output voltage corruption due to device leakage currents, a serious issue at frequencies below 1 GHz.

allow proper operation [9]. This modification leads to “ratioed” logic; i.e., the main latch transistors must be able to overcome those of Inv_1 and Inv_2 for a state change. The following example illustrates this point.

Example 20.22

Suppose CK is low in Fig. 20.50(b) and the two inverters hold a ZERO at N (and a ONE at N'). Now, CK goes high while D_{in} is low. Explain how the transistor dimensions must be chosen to ensure proper switching.

Solution

Shown in Fig. 20.51 is the critical part of the latch, revealing that the series combination of two PMOS devices must pull N high while the NMOS transistor within the inverter attempts to keep it at a low level. For this reason, we typically select the widths of the inverter transistors to be about 25% of those in the main branches.

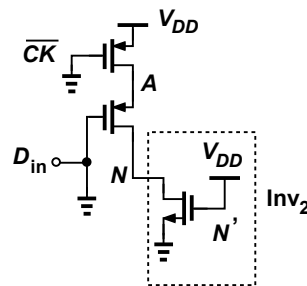


Figure 20.51 Fight between C^2 MOS latch and inverter pull-down device.

It is interesting to note that, if either D_{in} or $\overline{D_{in}}$ in Fig. 20.50(b) is low, only one of M_1 and M_2 conducts. Thus, A and A' can be shorted to each other and so can B and B' . That is, the number of clocked devices can be reduced from four to two [10].

To convert the waveforms of Fig. 20.49(b) to nonoverlapping phases, we first observe that X_1 and X_2 are simultaneously high between t_1 and t_2 . Thus, an AND operation can generate an output with a 25% duty cycle. Similarly, $X_1 \cdot Y_2$, $Y_1 \cdot X_2$, and $Y_1 \cdot Y_2$ provide the other phases (Fig. 20.52).

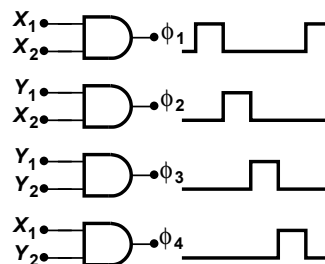


Figure 20.52 Generation of nonoverlapping phases.

Example 20.23

The floor plan of a four-channel interleaved ADC can be designed as in Fig. 20.53(a) or Fig. 20.53(b). Explain which one is preferable.

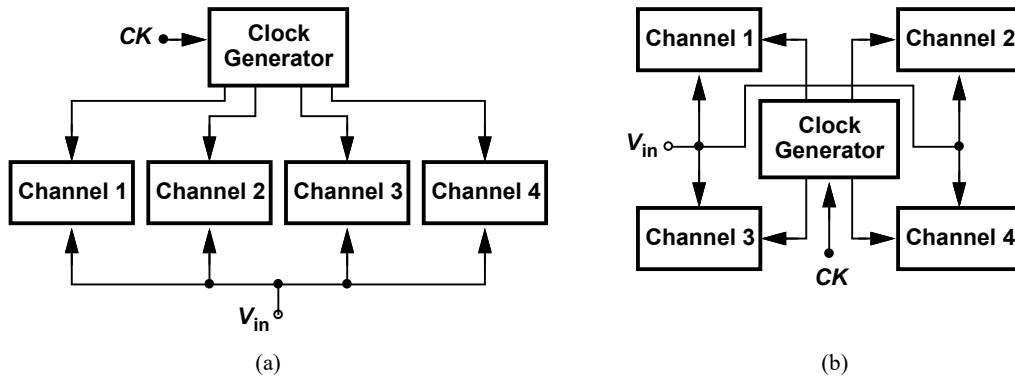


Figure 20.53 (a) Linear, and (b) quadrant-based floor plans.

Solution

In Fig. 20.53(a), the clock lines traveling to Channels 1 and 4 are longer than those reaching the other two, thus incurring an additional delay. In Fig. 20.53(b), on the other hand, all four phases are distributed symmetrically, but V_{in} now suffers from different delays. If V_{in} is distributed from the center of the floor plan, this arrangement is more desirable. This scheme, however, does not lend itself to eight channels.

As the clock propagates through the $\div 2$ circuit of Fig. 20.49(a) and the AND gates of Fig. 20.52, it accumulates jitter and phase mismatch. These effects can be reduced if the final outputs are retimed by the main clock. The AND gates readily lend themselves to such a method if they accept three inputs [Fig. 20.54(a)]. Let us assume a finite delay for the divider and sketch the waveforms as shown in Fig. 20.54(b). Note that the falling edge of ϕ_1 is determined by the rising edge of CK and is thus free from jitter and mismatch in X_1 . When ϕ_1 finishes sampling and falls, the signal is sampled properly.

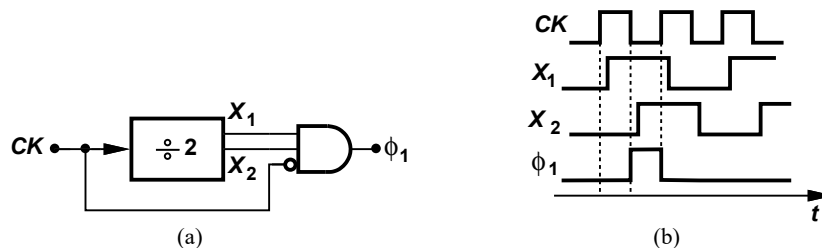


Figure 20.54 Use of input clock to retime divider outputs.

The divider used in our approach faces an important issue. When the circuit of Fig. 20.49 is powered up, it is possible that X_1 is 90° ahead of X_2 or 90° behind (Fig. 20.55). Suppose that sampling occurs on the falling edges. We observe that, if X_1 and X_2 begin with zero initial conditions, then their falling edges are $T_{CK}/4$ seconds apart. On the other hand, if X_1 begins at a high level, the time difference increases to $3T_{CK}/4$. This ambiguity may “scramble” the sampling points, and the four channels’ outputs may not emerge in the nominal order. This issue can be resolved by adding a reset mechanism to the latches or detecting the sequence of the clock phases and accordingly reordering the channels’ outputs.

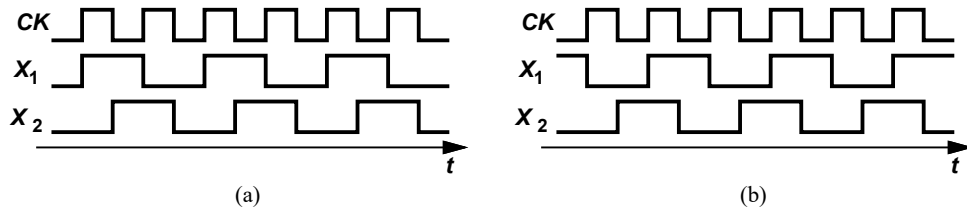


Figure 20.55 Different clock sequences resulting from divider initial conditions.

We now develop a clock generator for an eight-channel ADC. In this case, too, we begin with a clock frequency equal to one-half of the overall sampling rate and apply it to a chain of $\div 2$ stages [Fig. 20.56(a)]. As with the circuit of Fig. 20.49(a), each divider produces complementary (differential) in-phase (I) and quadrature (Q) waveforms. Suppose the I_1 output toggles on the rising edge of the input, and the Q_1 output on the falling edge. We thus arrive at I_2 , Q_2 , I_3 , and Q_3 in Fig. 20.56(b). Along with their complements, these overlapping clocks can drive eight channels.

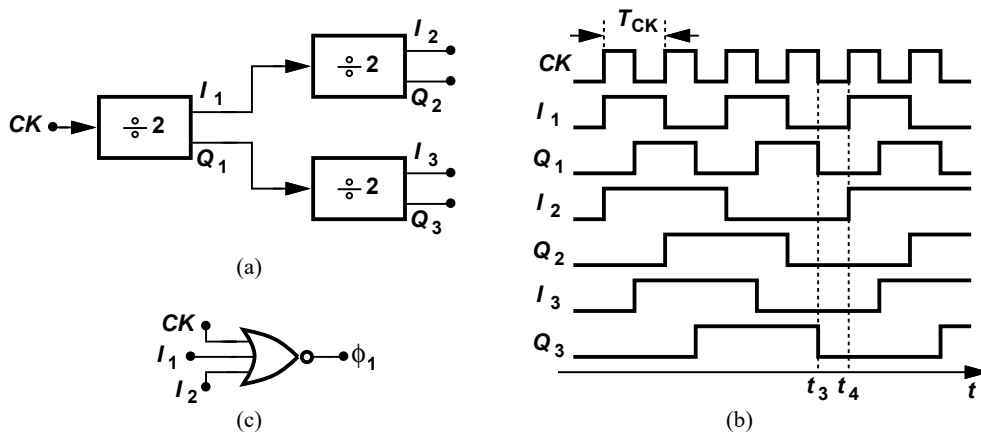


Figure 20.56 (a) Clock generator for an eight-channel ADC, (b) its waveforms, and (c) use of the main clock for retiming.

In order to generate eight even nonoverlapping phases, we employ simple logic. This is accomplished as shown in Fig. 20.56(b), where CK , I_1 , and I_2 are NORed. The output is high only if all three inputs are low, e.g., from t_3 to t_4 in Fig. 20.56(b). This event occurs once every $4T_{CK}$ seconds (why?), yielding a duty cycle of 12.5%. Similar NOR operations can generate the other seven phases.

Example 20.24

Suppose the dividers in Fig. 20.56(a) exhibit jitter or phase mismatches. Explain how ϕ_1 in Fig. 20.56(c) is corrupted by these effects.

Solution

We observe from Fig. 20.57 that uncertainties associated with the rising edges of I_0 and I_1 manifest themselves in the falling edge of ϕ_1 . But, as seen in the next example, the actual situation is more promising.

Example 20.25

Suppose the $\div 2$ stages in Fig. 20.56(a) exhibit some delay, ΔT_{div} . Explain how the waveforms change.

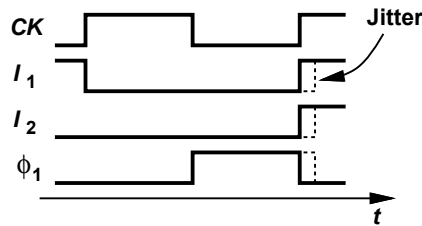


Figure 20.57 Effect of jitter in I_0 and I_1 .

Solution

As illustrated in Fig. 20.58, I_1 and I_2 are shifted by ΔT_{div} and $2\Delta T_{div}$, respectively. The ϕ_1 waveform remains unchanged. Interestingly, if the rising edge of I_1 experiences an uncertainty (even jitter) less than ΔT_{div} , it has no effect on the falling edge of ϕ_1 (why?). For I_2 , this margin is equal to $2\Delta T_{div}$. Thus, *both* edges of ϕ_1 are free from errors in this case.

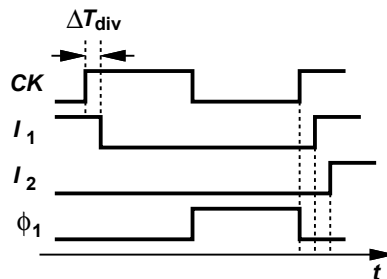


Figure 20.58 Effect of divider delay.

References

- [1]. W. C. Black and D. A. Hodges, "Time-interleaved converter arrays," *IEEE Journal of Solid-State Circuits*, vol. 15, pp. 1022-1029, Dec. 1980.
- [2]. B. Razavi, "Design considerations for interleaved ADCs," *IEEE Journal of Solid-State Circuits*, vol. 49, pp. 1806-1817, Aug. 2013.
- [3]. S. Le Tual, et al, "A 20GHz-BW 6b 10GS/s 32mW time-interleaved SAR ADC with master T/H in 28nm UTBB FDSOI technology," *ISSCC Dig.*, Feb. 2014.
- [4]. A. Varzaghani, et al, "A 10.3-GS/s, 6-bit flash ADC for 10G Ethernet applications," *IEEE Journal of Solid-State Circuits*, vol. 48, pp. 3038-3048, Dec. 2013.
- [5]. S. M. Jamal et al, "A 10-b 120-Msample/s time-interleaved analog-to-digital converter with digital background calibration," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 1618-1627, Dec. 2002.
- [6]. D. Fu, et al, "A digital background calibration technique for time-interleaved analog-to-digital converters," *IEEE Journal of Solid-State Circuits*, vol. 33, pp. 1904-1911, Dec. 1998.
- [7]. E. Janssen, et al, "An 11b 3.6GS/s time-interleaved SAR ADC in 65nm CMOS," *ISSCC Dig.*, Feb. 2013.
- [8]. H. Wei and B. Razavi, "An 8 Bit 4 GS/s 120 mW CMOS ADC," *IEEE Journal of Solid-State Circuits*, vol. 49, pp. 1751-1761, Aug. 2014.
- [9]. B. Razavi, *Design of CMOS Phase-Locked Loops*, Cambridge University Press, 2020.
- [10]. Y. Chang, et al, "An 80-Gb/s 40-mW wireline PAM4 transmitter," *IEEE Journal of Solid-State Circuits*, vol. 53, pp. 2214-2226, Aug. 2018.
- [11]. A. Haftbaradaran and K. W. Martin, "A sample-time error compensation technique for time-interleaved ADC systems," *Proceedings of CICC*, pp. 341-344, Sept. 2007.
- [12]. J. Elbornsson, F. Gustafsson, and J.-E. Eklund, "Blind equalization of time errors in a time-interleaved ADC system," *IEEE Transactions on Signal Processing*, vol. 53, pp. 1413-1424, April 2005.

- [13]. S. Huang and C. Levy, "Adaptive blind calibration of timing offset and gain mismatch for two-channel time-interleaved ADCs," *IEEE Transactions on Circuits and Systems I*, vol. 53, pp. 1278-1288, June 2006.
- [14]. V. Divi and G. W. Wornell, "Blind calibration of timing skew in time-interleaved analog-to-digital converters," *IEEE Journal on Selected Topics in Signal Processing*, vol. 3, pp. 509-522, June 2009.
- [15]. C.-Y. Wang and J.-T. Wu, "A background timing-skew calibration technique for time-interleaved analog-to-digital converters," *IEEE Transactions on Circuits and Systems II*, vol. 53, pp. 299-303, April 2006.

Problems

- 20.1.** In the interleaved system of Fig. 20.3(a), we halve the clock frequency while the input signal bandwidth remains unchanged. Redraw the spectra shown in Fig. 20.3(b).
- 20.2.** Consider the four-channel ADC shown in Fig. 20.5(a). Express the outputs in the time domain (as sequences of impulses). Note that CK_j is delayed by $T_{CK}/4$ with respect to CK_{j-1} .
- 20.3.** Using the results of the previous problem, sketch the output spectra of the four channels and explain what happens if they are summed.
- 20.4.** We observe in Fig. 20.7 that $V_{OS1} = V_{OS2}$ removes the tones from the output spectrum. What conditions are necessary for this to occur in the four-channel case depicted in Fig. 20.9?
- 20.5.** Suppose we are characterizing a two-channel interleaved ADC in the laboratory. We observe an unwanted tone in the output spectrum. Explain how we can determine whether the tone arises from offset mismatch or gain mismatch.
- 20.6.** Repeat the previous problem if we wish to know whether the tone arises from gain mismatch or phase mismatch. Assume we do not have access to the phase of the output FFT. (Hint: Only one type produces a tone amplitude proportional to f_{in} .)
- 20.7.** A two-channel interleaved ADC must achieve an SNR of 60 dB only due to bandwidth mismatch. (That is, quantization noise and other imperfections are neglected.) Using the model shown in Fig. 20.20, determine the maximum tolerable $\Delta R/R_1$ if the sinusoidal input frequency is equal to $\omega_{in} = 0.2/(R_1 C_1)$. (Hint: Use the SNR expression for phase mismatch.)
- 20.8.** Looking at Fig. 20.21(a), student A postulates that C_1 samples kT/C_1 when S_1 turns off, C_2 samples kT/C_2 when S_2 turns off, etc. Student B, on the other hand, believes that charge sharing between the capacitors also affects their kT/C noise. Who is right? For simplicity, assume the time constants of the circuit are much less than the acquisition period.
- 20.9.** The jitter of which clock(s) in Fig. 20.23(a) is critical?
- 20.10.** Suppose buffers B_1 and B_2 in Fig. 20.26(b) exhibits input-referred offsets equal to V_{OS1} and V_{OS2} , respectively. Setting V_{in} to zero, sketch the outputs of the four channels in the time domain and determine the tones resulting from this offset mismatch.
- 20.11.** Repeat the previous problem if B_1 has a gain of 1 and B_2 a gain of $1 + \alpha$. Assume $V_{in} = V_0 \cos \omega_0 t$.
- 20.12.** The input-referred offset of the ADC in Fig. 20.28 has a standard deviation of $\sigma = 5$ mV. We wish to reduce the offset from $\pm 3\sigma$ to 0.5 mV. Assuming a fully-differential ADC, determine the number of unit capacitors necessary for the DAC.
- 20.13.** A two-channel interleaved ADC digitizes a dc input. In such a situation, does the offset calibration method of Fig. 20.33(a) operate properly?
- 20.14.** Consider a two-channel interleaved ADC whose offset-mismatch is not corrected. Explain whether the gain calibration method of Fig. 20.12 operates properly in the presence of offset mismatch.
- 20.15.** In this problem, we study the phase mismatch detection of Fig. 20.39(a) for a sinusoidal input. As shown in Fig. 20.59, we have $V_{in} = V_0 \cos \omega_0 t$.

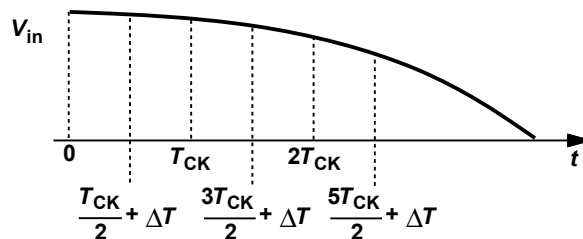


Figure 20.59 Phase mismatch example.

(a) Express the product of the first two samples as

$$P_{12} = V_0 \cos \left(\omega_0 \frac{T_{CK}}{2} + \omega_0 \Delta T \right) V_0 \cos(\omega_0 T_{CK}) \quad (20.44)$$

and convert the product to a sum of two cosines.

(b) Repeat part (a) for the product of samples 2 and 3:

$$P_{23} = V_0 \cos(\omega_0 T_{CK}) V_0 \cos\left(\frac{3}{2}\omega_0 T_{CK} + \omega_0 \Delta T\right). \quad (20.45)$$

(c) Now, calculate $P_{12} - P_{23}$ and express the difference between the cosines as a product of two sines. You then obtain two terms of the form $\sin(\omega_0 T_{CK}/2) \sin(2\omega_0 T_{CK} + \omega_0 \Delta T)$ and $\sin(\omega_0 \Delta T) \sin(\omega_0 T_{CK}/2)$.

(d) Repeat the previous parts for P_{34} and P_{45} , and compute $P_{34} - P_{45}$, arriving at two terms of the form $\sin(\omega_0 T_{CK}/2) \sin(4\omega_0 T_{CK} + \omega_0 \Delta T) + \sin(\omega_0 \Delta T) \sin(\omega_0 T_{CK}/2)$.

(e) We note that as we accumulate (integrate) these results, i.e., as we form $(P_{12} - P_{23}) + (P_{34} - P_{45})$, the first term averages to zero because $\sin(2\omega_0 T_{CK} + \omega_0 \Delta T)$, $\sin(4\omega_0 T_{CK} + \omega_0 \Delta T)$, etc., correspond to samples of the input every $2T_{CK}$ seconds. The second term, $\sin(\omega_0 \Delta T) \sin(\omega_0 T_{CK}/2)$, on the other hand, accumulates, yielding a dc value proportional to the timing mismatch because $\sin(\omega_0 \Delta T) \approx \omega_0 \Delta T$ if $\omega_0 \Delta T \ll 1$ rad.

20.16. Suppose the two VDLs in Fig. 20.43 are tied to the same supply voltage, V_{DD} , and this voltage contains noise. As V_{DD} fluctuates, the delays of the two chains increase or decrease together. Can we say that this “common-mode” jitter does not corrupt the samples taken by each channel?

20.17. In Fig. 20.45(c), what happens to the delay characteristic if M_4 is excessively wide or narrow?

20.18. Explain what happens to the four clock phases in Fig. 20.49(b) if CK does not have a 50% duty cycle.

20.19. Extend the floor plans of Fig. 20.53 to eight and 16 channels.

20.20. Suppose the input clock in Fig. 20.56(b) does not have a 50% duty cycle. Sketch all of the output waveforms in this case and explain which ones experience phase mismatches.

Introduction to Oversampling ADCs

The Nyquist-rate ADCs studied in previous chapters assume that the analog signal bandwidth can be as great as half of the sampling rate. In other words, the latter is chosen just high enough to satisfy Nyquist's criterion. In “oversampling” converters, on the other hand, the sampling rate is well above twice the input signal bandwidth, for example, by a factor of 8 or 16. Such a choice offers certain advantages that are absent in Nyquist-rate architectures, making oversampling particularly attractive for higher resolutions (typically greater than 12 bits).

This chapter deals with the fundamentals of oversampling ADCs. We first revisit the phenomenon of quantization noise and examine its spectrum. Next, we introduce the concept of “noise shaping” and describe how it can be realized in an oversampling environment. This development leads to the principle of $\Delta\Sigma$ modulation. Finally, we study first-order and second-order architectures.

21.1 Quantization Noise Revisited

Recall from Chapter 12 that the approximation of a continuous-amplitude signal by a discrete-amplitude counterpart leads to quantization noise. We have seen that the noise amplitude exhibits a relatively flat distribution between $-\Delta/2$ and $+\Delta/2$, where Δ denotes the quantizer's analog LSB value. But we have not studied the spectrum of this noise.

Let us begin at a fundamental level and apply a sinusoid, $x(t) = A \sin \omega_{in} t$, to a quantizer [Fig. 21.1(a)], e.g., a flash ADC. We can obtain the output waveform by one-to-one mapping of V_{in} through the quantizer's characteristic [Fig. 21.1(b)]. We now ask, does $y(t)$ contain *noise*, i.e., any random components? The answer is no: since $y(t)$ is periodic, Fourier tells us that it consists of only *harmonics*. Thus, it appears that quantization creates no noise! We must ponder this point further.

Suppose a sinusoid drives a 1-bit differential quantizer (i.e., a comparator). As shown in Fig. 21.2(a), the output is a square wave toggling between -1 and $+1$ and exhibits only odd harmonics with a peak amplitude of $(4/\pi)(1/n)$, where n denotes the harmonic order [Fig. 21.2(b)]. We can also examine $x(t) - y(t)$ [Fig. 21.2(c)], recognizing that this waveform represents the quantization error and contains the harmonics (except for the fundamental).

We next repeat the foregoing study for a three-level quantizer. The output waveform is shown in Fig. 21.3(a) and the error is plotted in Fig. 21.3(b). Similarly, Figs. 21.4(a) and 21.4(b) depict the results for a seven-level quantizer. We observe two trends as the quantizer resolution increases. First, the output more closely resembles the input, revealing that low-order harmonics (e.g., the third and the fifth) become smaller. Second, the error waveform, $x(t) - y(t)$, displays a greater number of fluctuations in one input cycle. This effect suggests that high-order harmonics become *larger*. We thus expect that quantizers having a resolution of roughly 4 bits or higher produce a relatively *uniform* set of harmonics—and up to very high orders (Fig. 21.5). The spectrum in Fig. 21.5 still suggests that quantization produces no *noise*! But recall that an ADC must sample the signal as well so as to create a discrete-time output. Before exploring the implications, let us prove that the *order* of quantization and sampling does not affect an ADC's output.

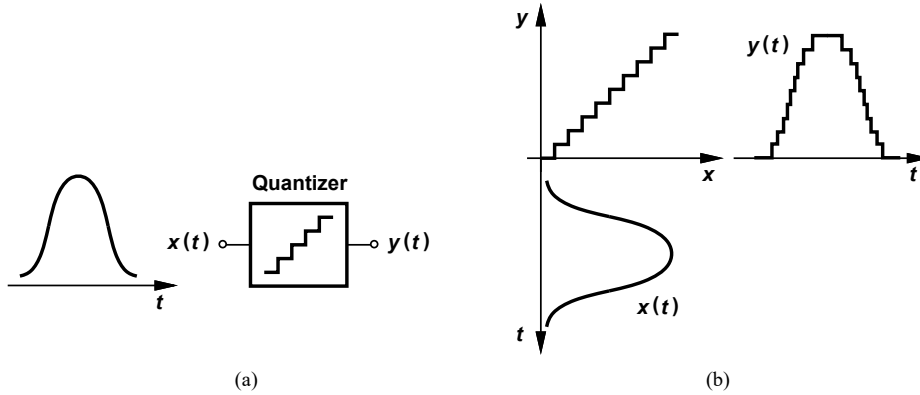


Figure 21.1 (a) A quantizer sensing a sine wave, and (b) computation of the output from the quantizer characteristic.

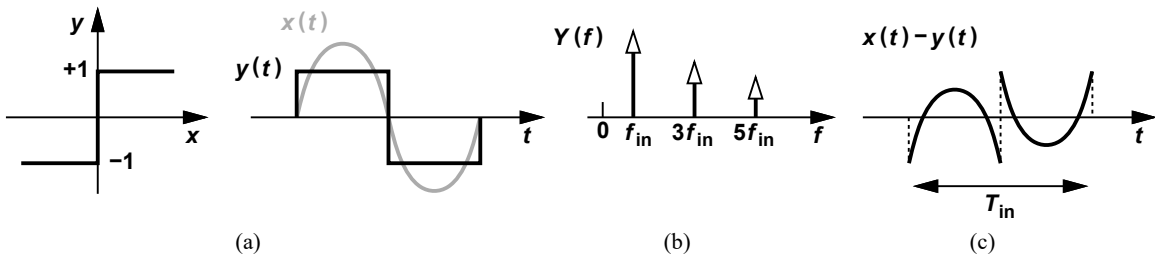


Figure 21.2 (a) A 1-bit quantizer and its input and output waveforms, (b) output spectrum, and (c) quantizer error waveform.

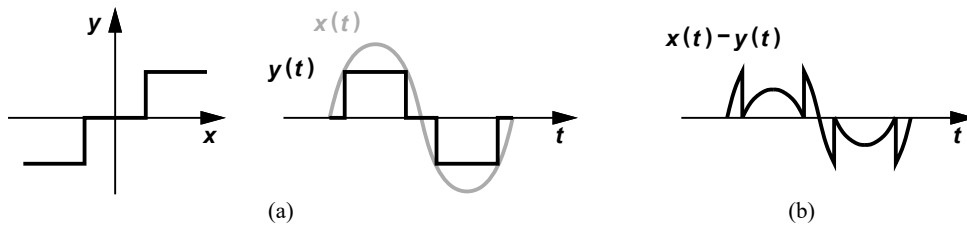


Figure 21.3 (a) A three-level quantizer and its input and output waveforms, and (b) quantizer error waveform.

Example 21.1

An analog signal is applied to a sampler and a quantizer. Prove that the two operations can be swapped.

Solution

As shown in Fig. 21.6(a), a sampled signal driving a quantizer generates discrete-amplitude components at T_S , $2T_S$, etc. Figure 21.6(b) illustrates that a quantized signal applied to a sampler leads to the same output. Thus, the order is unimportant—but only for our conceptual study here.

Suppose the quantizer output spectrum of Fig. 21.5 is subjected to sampling at a rate slightly greater than $2f_{in}$, e.g., $f_S = 2f_{in} + \Delta f$. As a result, all of the components shift up and down by integer multiples of $2f_{in} + \Delta f$. The fundamental thus maps to f_{in} , $2f_{in} + \Delta f \pm f_{in}$, $2(2f_{in} + \Delta f) \pm f_{in}$, etc. [Fig. 21.7(a)]. The third harmonic lands at $3f_{in}$, $2f_{in} + \Delta f \pm 3f_{in}$, $2(2f_{in} + \Delta f) \pm 3f_{in}$, etc. [Fig. 21.7(b)].

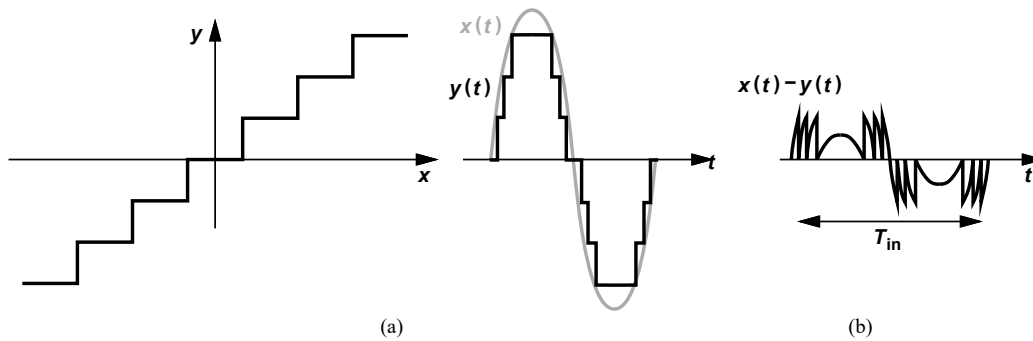


Figure 21.4 (a) A seven-level quantizer and its input and output waveforms, and (b) quantizer error waveform.

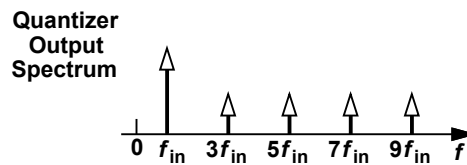


Figure 21.5 Output spectrum of a quantizer.

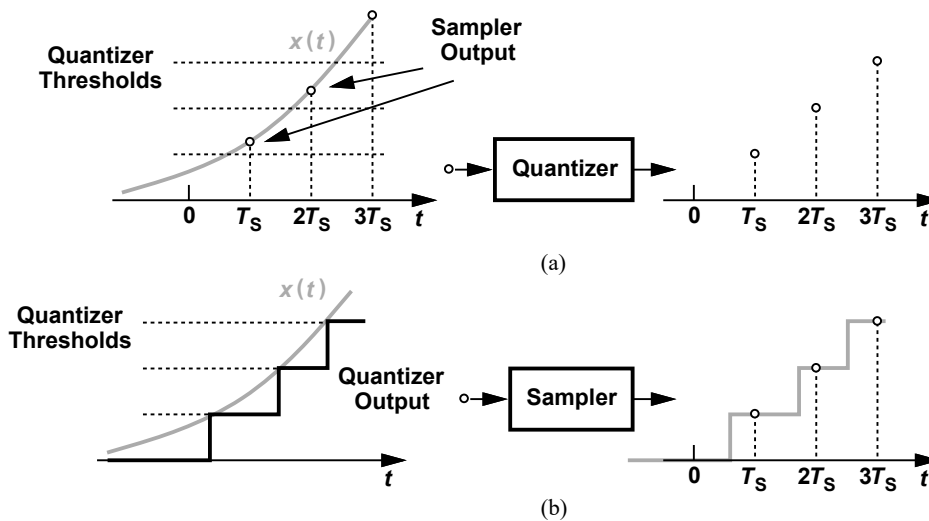


Figure 21.6 (a) Sampling followed by quantization, and (b) quantization followed by sampling.

Example 21.2

Determine some more components that fall below f_{in} in Fig. 21.7(b).

Solution

The m -th harmonic of the input and the n -th harmonic of the clock can generate a frequency component less than f_{in} :

$$n(2f_{in} + \Delta f) - mf_{in} < f_{in}. \quad (21.1)$$

For example, $2(2f_{in} + \Delta f) - 5f_{in} = -f_{in} + 2\Delta f$, $3(2f_{in} + \Delta f) - 7f_{in} = -f_{in} + 3\Delta f$, etc. Shown in Fig. 21.8, the resulting spectrum exhibits a large number of aliased components between zero and f_{in} .

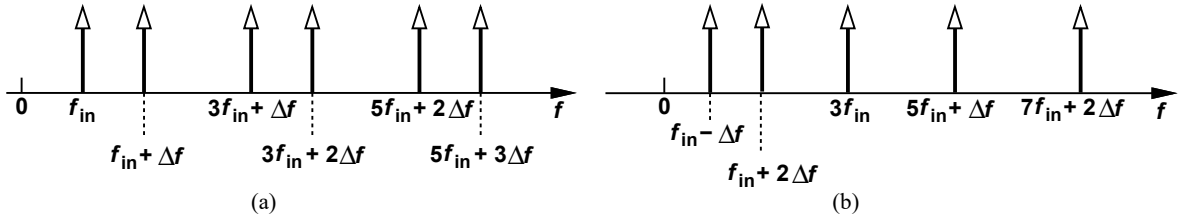


Figure 21.7 Spectrum of sampled output of a quantizer showing the effect on (a) the first, and (b) the third harmonic.

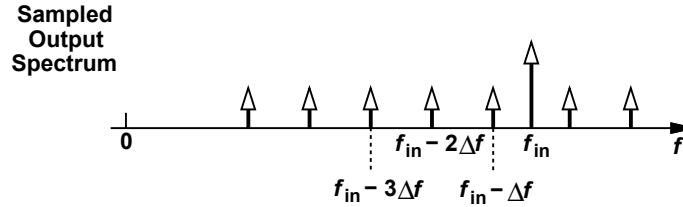


Figure 21.8 Additional frequency components due to sampling.

The large number of closely-spaced aliased components in Fig. 21.8 can be approximated by a continuous spectrum (Fig. 21.9). It is in this spirit that we say the quantizer produces “noise;” we consider the area under the spectrum from $f = 0$ to $f_{in} \approx f_S/2$ to be the power of the noise that corrupts the signal.

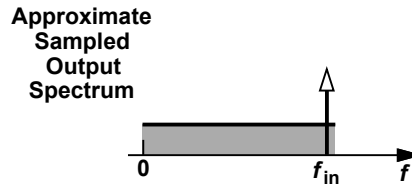


Figure 21.9 Approximation of spurs by a continuous spectrum.

We accept, without proof, three approximations for quantization noise: (1) it is uncorrelated with the analog input signal;¹ (2) its spectrum is flat (like white noise); and (3) its total power integrated from $-f_S/2$ to $+f_S/2$ is equal to $\Delta^2/12$, where Δ is the quantizer’s analog LSB value (Fig. 21.10). It follows that the spectral density is equal to $\Delta^2/(12f_S)$.

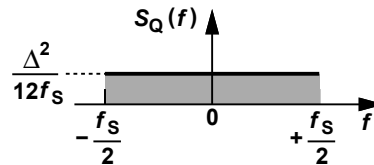


Figure 21.10 Quantization noise spectrum.

An interesting phenomenon emerging from this study is that *increasing* the sampling rate *reduces* the integrated quantization noise if the analog input bandwidth remains constant. Illustrated in Fig. 21.11, this trend suggests that doubling f_S lowers the “in-band” quantization noise (Q-noise) power by 3 dB. As explained

¹That is, the two bear no resemblance.

below, oversampling converters do employ a higher sampling rate than that required by Nyquist's theorem, but they also rely on “noise shaping” to further reduce Q-noise.

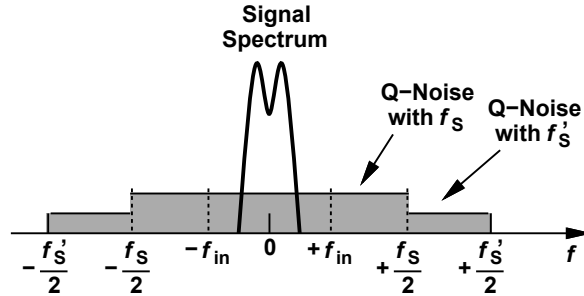


Figure 21.11 Effect of increasing the sampling rate on in-band noise.

21.2 Basic Concepts

21.2.1 Q-Noise Model

We have seen in the previous section that Q-noise is uncorrelated with the analog input signal. We can therefore view this corruption as an additive term and model simple quantizers as shown in Fig. 21.12. We have $y(t) = x(t) + q(t)$, where $q(t)$ is the Q-noise. Note that the coefficient of $x(t)$ signifies a unity gain for the quantizer. We assume $q(t)$ has a flat spectrum from $-f_S/2$ to $+f_S/2$, and write

$$S_Y(f) = S_X(f) + S_Q(f), \quad (21.2)$$

where $S(f)$ denotes the spectrum. The reader can visualize the quantizer simply as a flash stage for now.

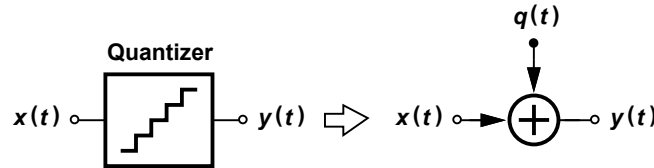


Figure 21.12 Simple quantizer model.

21.2.2 Noise in a Feedback System

Consider the negative-feedback system shown in Fig. 21.13(a), where a noise component, $n(t)$, is injected “near” the output port. How does $n(t)$ corrupt $y(t)$? We note that a negative-feedback system attempts to make the feedback signal a faithful replica of the input. Thus, $y(t)$ tracks $x(t)$ closely, as if $n(t)$ were absent. This is true so long as the system maintains a high loop gain.

If $x(t)$ and $n(t)$ in Fig. 21.13(a) are uncorrelated, we can determine their output contributions by superposition. Setting $x(t)$ to zero, we arrive at the loop depicted in Fig. 21.13(b) (why?) and hence

$$\frac{Y(s)}{N(s)} = \frac{1}{1 + H(s)}. \quad (21.3)$$

We call $1/[1 + H(s)]$ the “noise transfer function” (NTF) and denote it by $H_N(s)$. Similarly, the “signal transfer function” (STF) is given by

$$\frac{Y(s)}{X(s)} = \frac{H(s)}{1 + H(s)}. \quad (21.4)$$

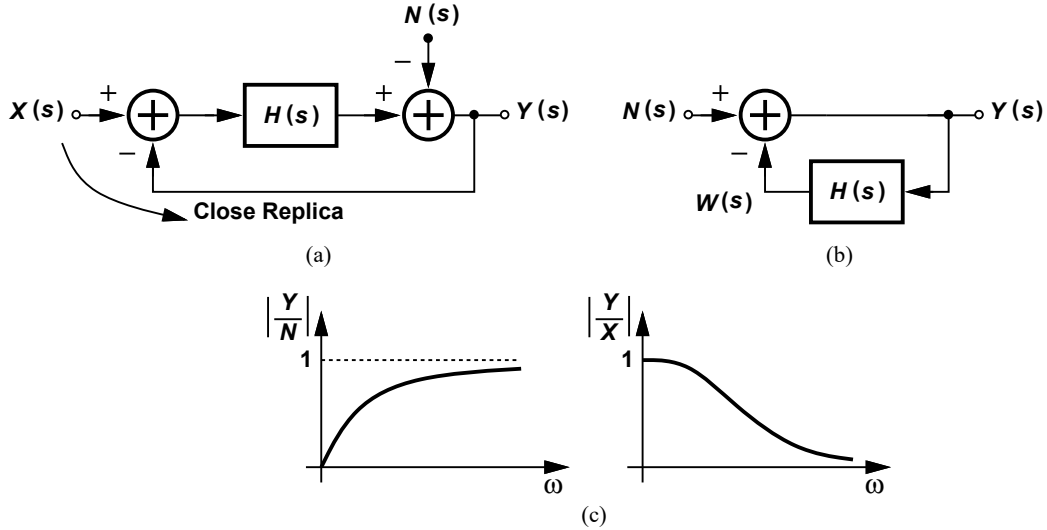


Figure 21.13 (a) Effect of noise injected “near” the output of a feedback loop, (b) equivalent block diagram, and (c) responses from the noise source and from the main input to the output.

The NTF plays a central role in our studies and merits some thought. Suppose, for example, that $H(s)$ is an integrator, i.e., $H(s) = 1/s$. Then,

$$H_N(s) = \frac{Y(s)}{N(s)} = \frac{s}{s+1}. \quad (21.5)$$

The signal transfer function emerges as $1/(s+1)$. Interestingly, the noise experiences a *high-pass* response, and the signal a *low-pass* response [Fig. 21.13(c)].

Example 21.3

Explain intuitively why the foregoing NTF is a high-pass function.

Solution

Considering the loop in the time domain and setting $x(t)$ to zero, we first assume that $n(t)$ in Fig. 21.13(b) varies slowly. The integrator, $H(s)$, thus provides a high gain, ensuring that $w(t)$ tracks $n(t)$. In other words, the loop *opposes* the effect of $n(t)$ by generating a feedback component that cancels it. As a result, $y(t) = n(t) - w(t) \approx 0$. Note that the output of $H(s)$ in Fig. 21.13(a) is approximately equal to $-n(t)$ in this case (why?).

Now, we allow $n(t)$ to vary at a high rate, recognizing that the loop in Fig. 21.13(b) exhibits less gain. Consequently, $w(t)$ does not follow $n(t)$ closely. In the limit, the integrator gain is zero and $y(t) = n(t)$.

With the NTF known, one can obtain the output noise spectrum of the feedback system shown in Fig. 21.13(a). We know that the output spectrum of a linear, time-invariant system is equal to the input spectrum multiplied by the magnitude squared of the transfer function:

$$S_Y(f) = S_N(f) |H_N(j2\pi f)|^2 \quad (21.6)$$

$$= S_N(f) \frac{(2\pi f)^2}{1 + (2\pi f)^2}. \quad (21.7)$$

We say $|H_N(j2\pi f)|$ “shapes” $S_N(f)$. Illustrated in Fig. 21.14 is this effect for the case of a flat $S_N(f)$. Of great interest to us is the fact that the noise is *suppressed* at low frequencies. In our studies, we will draw two-sided spectra (from $f = -\infty$ to $f = +\infty$) or one-sided spectra (from $f = 0$ to $f = \infty$).

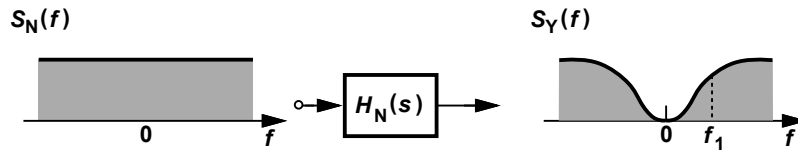


Figure 21.14 Shaping of noise spectrum.

In summary, a noise component injected “near” the output of a negative-feedback system is shaped by a transfer function equal to $1/[1 + H(s)]$ (for a unity feedback factor). This principle forms the foundation of our studies in this chapter.

21.2.3 Simple Noise-Shaping ADC

The previous sections have revealed two important points: (1) a quantizer’s noise simply adds to the analog signal, and (2) the noise spectrum can be shaped (and suppressed) by feedback. Combining these two ideas leads to noise-shaping ADCs.

Let us embed a quantizer in a feedback loop [Fig. 21.15(a)]. As in the previous section, the forward path includes an (analog) integrator. Moreover, to subtract the quantizer’s digital output from the analog input, we must insert a DAC in the feedback path. Assuming a unity gain for the quantizer and the DAC, we draw the equivalent circuit in Fig. 21.15(b). It follows that the quantization noise experiences an NTF given by

$$H_N(s) = \frac{s}{1 + s} \quad (21.8)$$

and is therefore shaped as in Fig. 21.14. That is, the *digital* output, $y(t)$, inherits less Q-noise at low frequencies. The loop in Fig. 21.15(a) exemplifies a “noise-shaping” ADC.

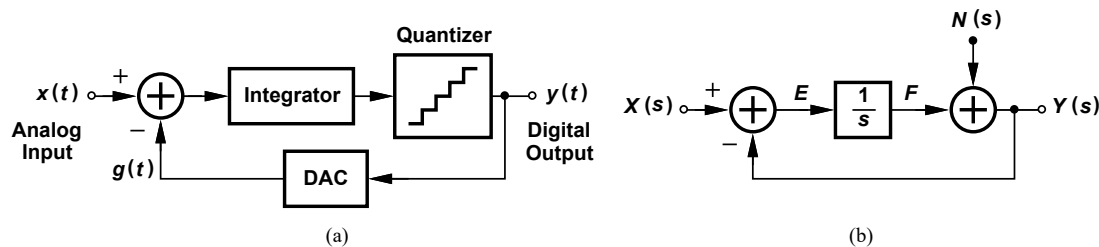


Figure 21.15 (a) A quantizer embedded in a loop containing an integrator, and (b) equivalent block diagram.

Another perspective gives more insights. As any well-designed negative-feedback system, the loop in Fig. 21.15(a) attempts to make $g(t)$ a faithful replica of $x(t)$, thereby suppressing the imperfections introduced by the forward path. At low frequencies, the high loop gain provided by the integrator ensures this replication, but at high frequencies, the loop gain drops and the output is corrupted by Q-noise to a greater extent. We hope that the total noise in the signal band of interest is small, a point to which we return later.

Example 21.4

A student reasons that the integrator in Fig. 21.15(a) can be replaced with a broadband high-gain amplifier (Fig. 21.16) so that the feedback loop keeps $g(t)$ a good replica of $x(t)$ even in the presence of *high*-frequency noise. Is this approach preferable?

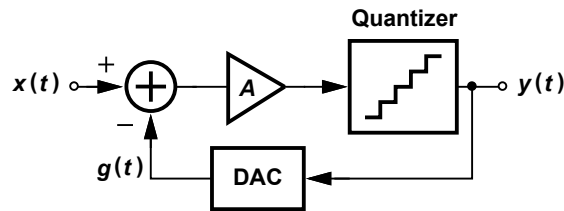


Figure 21.16 A quantizer embedded in a loop containing an amplifier.

Solution

No, it is not. The key point here is that the integrator provides a *very* high gain at low frequencies, an effect that would be difficult to create by means of several gain stages (within A) while maintaining loop stability.² That is, for a practical value of A , the total noise in the signal band is higher here.

A number of questions may come to the reader's mind at this point. (1) What does the noise-shaping loop do to the analog input signal, $x(t)$? (2) How do we deal with the *large* output noise in Fig. 21.14, e.g., beyond $f = f_1$? (3) Where does *sampling* occur in this ADC?

As computed in the previous section, the input signal experiences a low-pass response given by $1/(1+s)$. Summarized in Fig. 21.17(a), the two responses cross at a frequency equal to $1/(2\pi)$; that is, the input signal bandwidth, f_0 , must not exceed this frequency.³ However, if we wish to minimize the in-band Q-noise, we must limit f_0 to well below this value [Fig. 21.17(b)]. As explained later, we integrate the shaped Q-noise from $-f_0$ to $+f_0$ and ensure that the resulting power is small enough to meet the desired resolution.

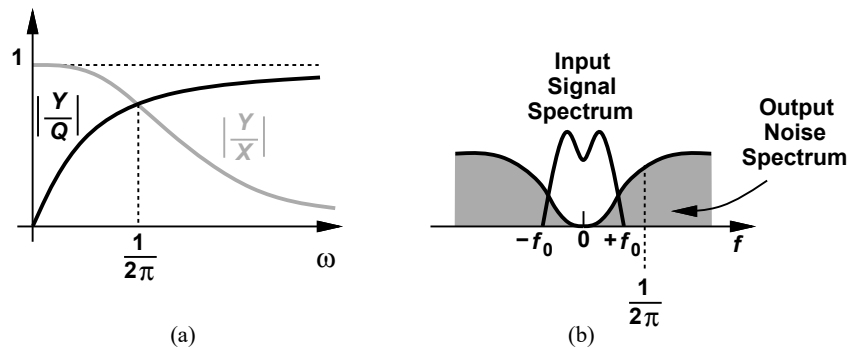


Figure 21.17 (a) Summary of transfer functions, and (b) illustration of Q-noise shaping and its effect on signal.

To answer the second question, we surmise that the digital output of the quantizer in Fig. 21.15(a) must be filtered such that it retains the signal and removes the high-frequency Q-noise. Returning to Fig. 21.17(b), we expect a corner frequency of about f_0 for this low-pass filter. Digital filters can provide a sharp cut-off (Chapter 24).

For the third question, we recognize that sampling can occur before the input summer, before the quantizer, or within the quantizer (as in a flash stage). We study these possibilities as we develop the circuit details.

Example 21.5

Compute the signal and noise components that appear at different nodes in Fig. 21.15(b).

²Recall from analog design that a multi-stage amplifier exhibits multiple poles and hence significant phase shift.

³The frequency $1/(2\pi)$ is a normalized value because the integrator is modeled by $1/s$. It will be converted to an actual value in the next section.

Solution

We have

$$Y(s) = \frac{1}{s+1}X(s) + \frac{s}{s+1}N(s) \quad (21.9)$$

and hence

$$E(s) = X(s) - Y(s) \quad (21.10)$$

$$= \frac{s}{s+1}X(s) - \frac{s}{s+1}N(s). \quad (21.11)$$

That is, node E carries high-pass-filtered versions of X and N . Also,

$$F(s) = \frac{1}{s+1}X(s) - \frac{1}{s+1}N(s), \quad (21.12)$$

revealing low-pass-filtered components at the integrator output.

The feedback loop of Fig. 21.15(a) is called a “ $\Delta\Sigma$ modulator” (DSM) as it first computes the difference $x(t) - g(t)$ (hence the term “Delta”) and then sums it with the previous value held by the integrator (hence the term “Sigma”). Most oversampling ADCs employ $\Delta\Sigma$ modulation, but we have not yet seen the relation between the two concepts.

Example 21.6

A student examining the model shown in Fig. 21.15(b) wonders why the quantizer is assumed to have a unity gain. Given that the input is an analog quantity and the output a digital one, what do we exactly mean by “gain” here?

Solution

An intuitive view point is to consider the cascade of the quantizer and the DAC as one block that senses an analog input and generates an analog output. If the input and output full scales are equal, the gain is unity.

21.2.4 Noise Shaping in Time Domain

The notion of injecting noise “near” the output of a feedback system is intuitively appealing in the frequency domain as it leads to spectral shaping. We can gain additional insights if we examine this phenomenon in the time domain as well. We hereafter denote the quantizer’s noise by $q(t)$.

Let us begin our study by analyzing the time-domain behavior of the $\Delta\Sigma$ modulator model shown in Fig. 21.18(a). The subtractor output, $x(t) - y(t)$, is integrated and summed with $q(t)$, yielding

$$y(t) = \int [x(t) - y(t)]dt + q(t). \quad (21.13)$$

Differentiating both sides and rearranging the terms, we have

$$\frac{dy}{dt} + y(t) = x(t) + \frac{dq}{dt}. \quad (21.14)$$

The reader may recognize the left-hand side as the representation of a first-order system. [Recall the form $RCdy/dt + y(t)$ for an RC section.] That is, both $x(t)$ and dq/dt are subjected to a low-pass response of the

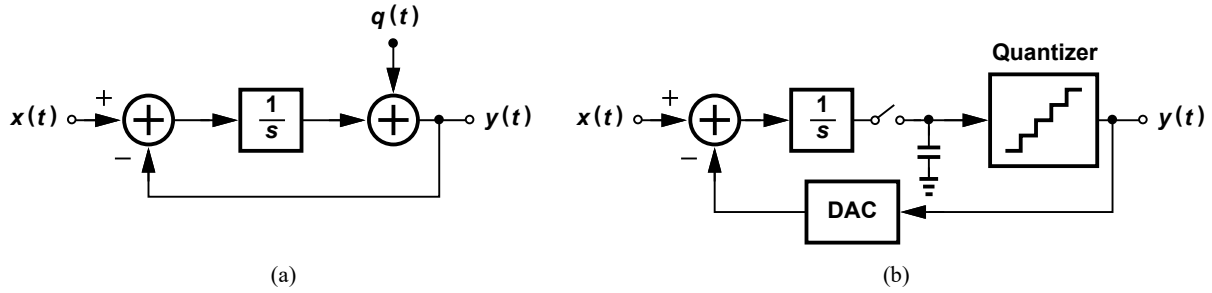


Figure 21.18 (a) A loop containing an integrator, and (b) addition of sampler, quantizer, and DAC.

form $1/(s+1)$. But what does dq/dt signify? Since a differentiator's transfer function is given by $G(s) = s$, we conclude that $q(t)$ experiences a transfer function given by the product of s and $1/(s+1)$, as predicted by (21.5).

The differentiation effect incurred by $q(t)$ directly translates to noise shaping, creating $S_Q(f)(2\pi f)^2$ at the output.⁴ That is, the shaped noise spectrum follows a parabolic shape. This holds for frequencies well below the low-pass corner, $f = 1/(2\pi)$, in Fig. 21.17(a). For the remainder of our study, we do need to assume that sampling occurs somewhere in the $\Delta\Sigma$ loop of Fig. 21.18(a), for example, at the quantizer input [Fig. 21.18(b)].

The foregoing thoughts allow us to construct $\Delta\Sigma$ modulators at the circuit level. In the next section, we realize one using a discrete-time (DT) integrator, which also performs signal sampling. In Section 21.4, we repeat our effort with a continuous-time (CT) integrator.

21.3 Discrete-Time $\Delta\Sigma$ Modulator

21.3.1 Discrete-Time Integrator

Basic Topology We are familiar with the continuous-time integrator shown in Fig. 21.19(a), where R_1 converts V_{in} to a current and C_1 integrates the result. To convert this structure to a discrete-time configuration, we focus on R_1 . Maxwell recognized that a resistor tied between two nodes, A and B , can be approximated by a grounded capacitor that periodically switches between the two [Fig. 21.19(b)]. In one-half of the clock cycle, C_2 draws charge from V_A , and in the next half, it delivers the charge to V_B . We can say this network serves as a DT approximation of R_{AB} . Each time C_2 flips from V_B to V_A , it receives charge in the amount of $C_2(V_A - V_B)$. That is, the average current flowing from V_A to V_B is equal to $C_2(V_A - V_B)/T_{CK}$, where $T_{CK} = 1/f_{CK}$ denotes the clock cycle. The equivalent resistance is given by V_{AB} divided by this average current:

$$R_{eq} = \frac{1}{f_{CK}C_2}. \quad (21.15)$$

Let us now replace R_1 in the integrator with its DT counterpart, as depicted in Fig. 21.19(c). This circuit indeed integrates its input: first, C_2 collects charge from V_{in} , and then it passes *all* of it to C_1 (why?). Since C_1 is *not* reset, it continues to accumulate the input samples.⁵

Example 21.7

If V_{in} in Fig. 21.19(c) is constant and equal to V_a , plot V_{out} as a function of time. Assume $V_{out}(0^-) = 0$.

⁴Recall that the output spectrum of a system is given by the input spectrum multiplied by the magnitude-squared of the transfer function.

⁵One can view C_1 and the op amp as a reservoir and C_2 as a bucket that receives water from the input and pours it into the reservoir.

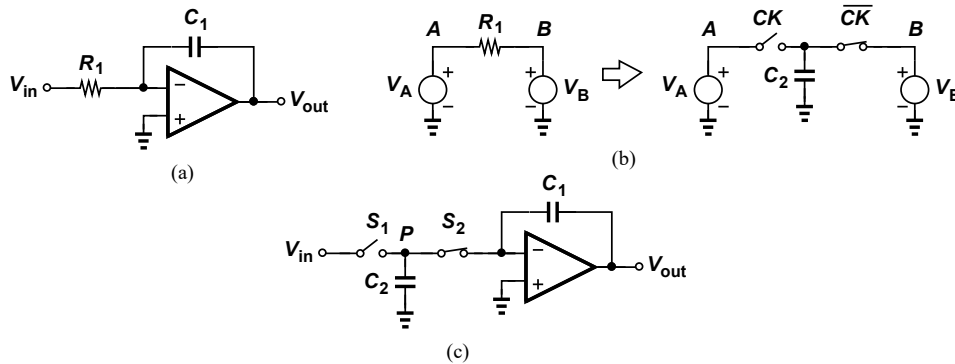


Figure 21.19 (a) Simple continuous-time integrator, (b) approximation of a resistor by a switched capacitor, and (c) discrete-time integrator.

Solution

Capacitor C_2 receives charge in the amount of $C_2 V_a$ and deposits it on C_1 , creating an output change equal to $-V_a(C_2/C_1)$. Thus, V_{out} grows toward $-\infty$ (Fig. 21.20), with a slope equal to $-(C_2/C_1)/T_{CK}$. We say the integrator inverts the input because a positive V_a leads to negative jumps at the output.

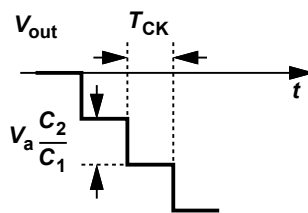


Figure 21.20 Output of a discrete-time integrator in response to a constant input.

It is instructive to examine the integrator's waveforms for a general input, noting that the circuit also performs sampling. As illustrated in Fig. 21.21, while CK is high, V_P in Fig. 21.19(c) tracks V_{in} , and V_{out} holds a certain value. When CK goes low, $V_P = V_{in}(t_2)$, and the charge on C_2 is equal to $V_{in}(t_2)C_2$. Since C_2 is switched to the virtual ground of the integrator, this charge moves to C_1 , creating a change equal to $-V_{in}(t_2)C_2/C_1$. The integrator output is thus valid around $t = t_3$, one-half of T_{CK} later than t_2 .

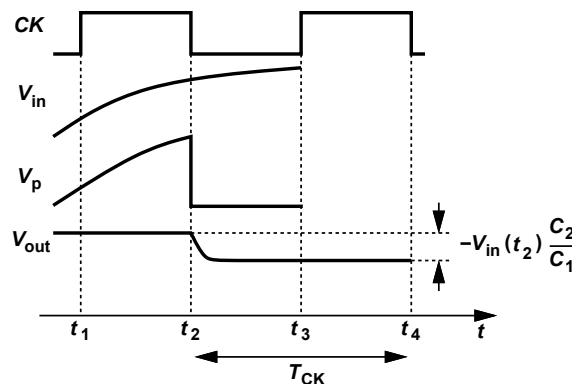


Figure 21.21 Integrator's waveforms.

We now wish to express the integrator output as a discrete-time function of its input. Each new value of V_{out} is equal to the previous value minus V_{in} . In Fig. 21.21, for example, we have $V_{out}(t_4) = V_{out}(t_2) - (C_2/C_1)V_{in}(t_2)$. But why do we focus on $V_{out}(t_4)$ rather than on $V_{out}(t_3)$? From the waveforms of Fig. 21.21, we surmise that the sampled level of V_{in} at t_2 is available in V_{out} at t_3 , i.e., it experiences a delay equal to $T_{CK}/2$. As explained in Section 21.3.2, however, the integrator output is sensed and processed at $t = t_4$. That is, the effective delay is equal to T_{CK} . We then have

$$V_{out}(nT_{CK}) = V_{out}[(n-1)T_{CK}] - \frac{C_2}{C_1}V_{in}[(n-1)T_{CK}]. \quad (21.16)$$

The factor C_2/C_1 is called the “gain” of the integrator. In discrete-time signal processing, we drop T_{CK} and simply write

$$V_{out}[n] = V_{out}[n-1] - \frac{C_2}{C_1}V_{in}[n-1]. \quad (21.17)$$

It is helpful to articulate this operation as follows: the present output value is equal to the difference between the values of V_{out} and $(C_2/C_1)V_{in}$ in the previous cycle.

Example 21.8

Construct a discrete-time feedback system that implements Eq. (21.17).

Solution

We can view the right-hand side as follows: the (scaled) input is subtracted from the output and *then* the result is delayed by one clock cycle to generate the new output. We thus arrive at the embodiment shown in Fig. 21.22, where the subtractor output is equal to $V_{out}[n-1] - (C_2/C_1)V_{in}[n-1]$. The next value of V_{out} is equal to this difference. Note that the loop provides *positive* feedback, a necessary attribute for signal accumulation (integration). For simplicity, we assume $C_2/C_1 = 1$ in this chapter.

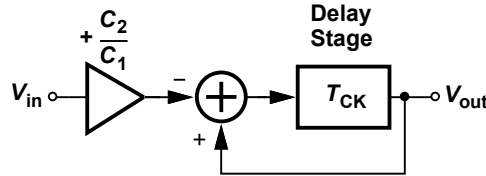


Figure 21.22 Discrete-time integrator model.

The reader may recognize that the integrator of Fig. 21.19(c) suffers from two effects: (1) input-dependent charge injection onto C_2 (Chapter 3), and (2) the nonlinear junction capacitances of S_1 and S_2 at node P . In Chapter 22, we introduce a topology that avoids both issues.

Integrator Transfer Function Recall from Section 21.2.3 that noise shaping relies on an integrator transfer function equal to $1/s$. We naturally wonder whether our DT design exhibits such a behavior as well. Let us consider the simple loop depicted in Fig. 21.23, where $x[n-1] + y[n-1]$ is delayed to produce the present value of $y[n]$.⁶ We know that the transfer function of the delay stage itself is equal to $\exp(-sT_{CK})$.⁷ Thus,

$$\frac{Y}{X}(s) = \frac{\exp(-sT_{CK})}{1 - \exp(-sT_{CK})}. \quad (21.18)$$

⁶We might call this a “noninverting” integrator as it resembles Fig. 21.22 but without the $-C_2/C_1$ scaling factor.

⁷This is obtained by taking the Laplace transform of the stage’s impulse response.

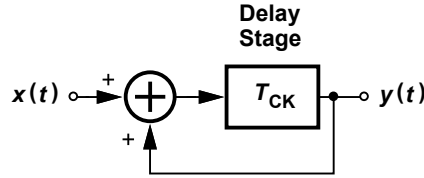


Figure 21.23 Loop containing a delay stage.

If the frequencies of interest are far below the clock rate, we have $|sT_{CK}| \ll 1$ and hence $\exp(-sT_{CK}) \approx 1 - sT_{CK}$. That is,

$$\frac{Y}{X}(s) \approx \frac{\exp(-sT_{CK})}{sT_{CK}}. \quad (21.19)$$

This transfer function corresponds to a CT system and signifies a *cascade* consisting of a delay stage and an integrator. It reveals that the DT topology in Fig. 21.23 approximates a CT structure if the signal frequency range is well below f_{CK} . Whether or not the net delay of T_{CK} (the numerator) is undesirable will be revisited later.

In a general case, the foregoing approximation may not hold, and we must resort to discrete-time analysis and the z -transform. We note that, if a discrete-time (i.e., sampled) signal experiences a delay of one clock period, its z -transform is multiplied by z^{-1} . We then represent the one-cycle delay in Fig. 21.23 by z^{-1} and write

$$\frac{Y}{X}(z) = \frac{z^{-1}}{1 - z^{-1}}. \quad (21.20)$$

Contrasting Eqs. (21.20) and (21.18), the reader may wonder whether $z = \exp(sT_{CK})$. This is not entirely correct. To compute the Fourier transform of a discrete-time signal, we actually have $z = \exp(j2\pi fT_{CK})$, i.e., s is confined to the imaginary axis in the complex plane. In this expression, f denotes the frequency of the continuous-time signal before it is sampled. Thus, f has a continuous range. It is helpful to remember that, for low frequencies, $z \rightarrow 1$ and $Y/X \rightarrow \infty$. Equation (21.20) suggests that a DT integrator can be modeled by the positive-feedback loop shown in Fig. 21.24.

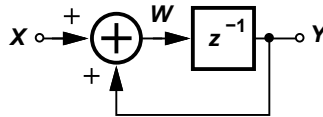


Figure 21.24 Loop containing a one-clock delay.

Example 21.9

We redraw the loop of Fig. 21.24 as depicted in Fig. 21.25. Compute the transfer function $W(z)/X(z)$.

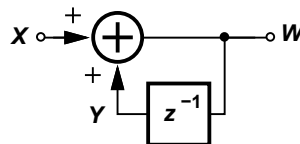


Figure 21.25 Loop containing a one-clock delay in feedback path.

Solution

The forward gain is unity and the feedback factor equal to z^{-1} . It follows that

$$\frac{W}{X}(z) = \frac{1}{1 - z^{-1}}. \quad (21.21)$$

Both (21.20) and (21.21) represent integrators, but the former delays the input by one clock cycle. The corresponding circuits are thus called “delaying” and “nondelaying” integrators.

Example 21.10

Prove that Eq. (21.17) represents a delaying integrator.

Solution

Taking the z -transform of both sides, we have $V_{out}(z) = z^{-1}V_{out}(z) - (C_2/C_1)z^{-1}V_{in}(z)$ and hence

$$\frac{V_{out}}{V_{in}}(z) = \frac{-(C_2/C_1)z^{-1}}{1 - z^{-1}}. \quad (21.22)$$

21.3.2 First-Order $\Delta\Sigma$ Loop

The DT integrator developed in the previous section can now be placed in a $\Delta\Sigma$ loop [Fig. 21.26(a)]. As mentioned in Example 21.7, this integrator *inverts* its input, but we ignore this polarity reversal in our high-level drawings.⁸ In a manner similar to our approach in Fig. 21.15, we model quantization by additive noise and the DAC by a unity gain, arriving at the simplified system shown in Fig. 21.26(b). The noise transfer function is obtained by setting $X(z)$ to zero:

$$\frac{Y}{Q}(z) = \frac{1}{1 + \frac{z^{-1}}{1 - z^{-1}}} \quad (21.23)$$

$$= 1 - z^{-1}. \quad (21.24)$$

Does this result agree with our prediction in Section 21.2.4 that the output Q -noise in the steady state is equal to $q_{j+1} - q_j$? Since $Y(z) = Q(z) - z^{-1}Q(z)$ and since a factor of z^{-1} signifies a unit delay, we can write in the time domain $y[n] = q[n] - q[n-1]$. The two perspectives are therefore consistent. We call Eq. (21.24)

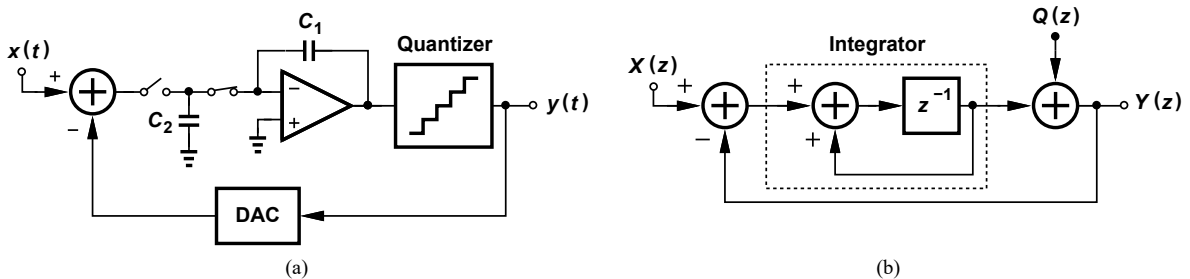


Figure 21.26 (a) First-order $\Delta\Sigma$ loop, and (b) its equivalent block diagram.

⁸This negative sign is readily accommodated in differential implementations.

a first-order noise-shaping function. Similarly, a system producing $q[n] - q[n - 1]$ is considered to perform first-order shaping on $q(t)$. Note that $z \rightarrow 1$ at low frequencies, yielding $Y/Q \rightarrow 0$.

The signal transfer function in Fig. 21.26(b) is computed by setting $Q(z)$ to zero:

$$\frac{Y}{X}(z) = \frac{\frac{z^{-1}}{1 - z^{-1}}}{1 + \frac{z^{-1}}{1 - z^{-1}}} \quad (21.25)$$

$$= z^{-1}. \quad (21.26)$$

That is, the system simply delays the input by one clock cycle. The reader may wonder how the input subtractor in Fig. 21.26(a) is implemented. We show in Chapter 22 that the integrator itself can perform this subtraction.

Interpretations of $1 - z^{-1}$ The noise transfer function expressed by (21.24) plays a central role in our understanding of $\Delta\Sigma$ modulators. We thus examine $1 - z^{-1}$ from two perspectives.

First, recall from Example 21.9 that the transfer function $(1 - z^{-1})^{-1}$ represents an integrator. It follows that $1 - z^{-1}$ must correspond to a differentiator (why?). This can also be seen by the CT approximations made in the previous section, where a one-cycle delay was modeled by $\exp(-sT_{CK}) \approx 1 - sT_{CK}$. We then write $1 - z^{-1} \approx 1 - (1 - sT_{CK}) = sT_{CK}$, which denotes a CT differentiator having a magnitude response of $2\pi fT_{CK}$.

Example 21.11

Using a time-domain view, explain why a response given by $1 - z^{-1}$ attenuates low frequencies.

Solution

As shown in Fig. 21.27, we construct a system that implements $1 - z^{-1}$. Suppose the input, $g(t)$, varies slowly. Thus, its delayed copy, $g_{\Delta}(t)$, is “similar” to $g(t)$, yielding a small difference at the output. The arrangement then rejects low frequencies, as a differentiator should.

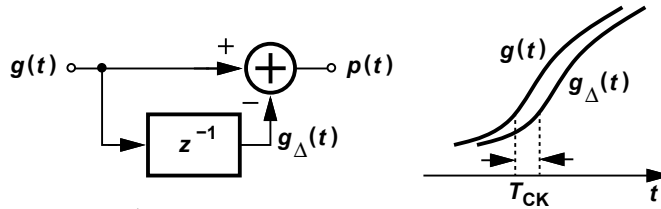


Figure 21.27 Response of a $1 - z^{-1}$ function.

Example 21.12

Explain qualitatively how the attenuation provided by $1 - z^{-1}$ changes if the clock frequency increases.

Solution

From Fig. 21.27, we observe that a shorter clock cycle makes $g(t)$ and $g_{\Delta}(t)$ more “similar,” thereby attenuating low frequencies to a greater extent (Fig. 21.28). In other words, a greater *oversampling ratio* leads to more suppression.

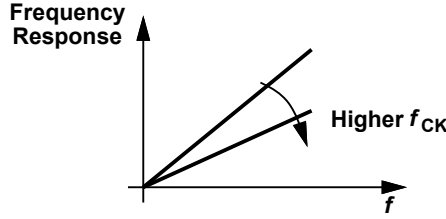


Figure 21.28 Effect of clock frequency on the response of $1 - z^{-1}$.

The second perspective for understanding the behavior of $1 - z^{-1}$ is based on the relation $z = \exp(j2\pi f T_{CK})$. In Fig. 21.27, we compute the magnitude of the corresponding frequency response⁹ as

$$\left| \frac{P}{G}(z = e^{j2\pi f T_{CK}}) \right| = |1 - e^{-j2\pi f T_{CK}}| \quad (21.27)$$

$$= |1 - \cos(2\pi f T_{CK}) + j \sin(2\pi f T_{CK})| \quad (21.28)$$

$$= \sqrt{2 - 2 \cos(2\pi f T_{CK})}. \quad (21.29)$$

(Recall that $|a + jb| = \sqrt{a^2 + b^2}$.) Since $1 - \cos \alpha = 2 \sin^2(\alpha/2)$, we have

$$\left| \frac{P}{G}(z = e^{j2\pi f T_{CK}}) \right| = |2 \sin(\pi f T_{CK})|. \quad (21.30)$$

We plot this noise-shaping function and its square in Fig. 21.29. Both carry a wealth of information.

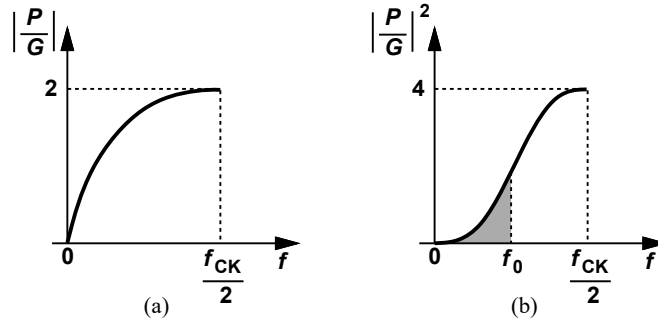


Figure 21.29 (a) Noise-shaping function, and (b) its magnitude squared.

Let us first examine the behavior in Fig. 21.29(a), bearing in mind that f denotes the frequency of the input, $g(t)$. If in Eq. (21.30), $\pi f T_{CK} \ll 1$ rad, i.e., if $f \ll f_{CK}/\pi$, we have $2 \sin(\pi f T_{CK}) \approx 2\pi f T_{CK}$. The response is then similar to that of a continuous-time differentiator, revealing that a very high oversampling ratio makes DT and CT systems nearly the same.

We next consider the plot in Fig. 21.29(b), recognizing that the Q-noise spectrum, $S_Q(f)$, in Fig. 21.10 is multiplied by this function as it travels to the output:

$$S_Y(f) = S_Q(f) |1 - z^{-1}|^2 \quad (21.31)$$

$$= S_Q(f) |2 \sin(\pi f T_{CK})|^2. \quad (21.32)$$

⁹The frequency response in this case represents how the output behaves if the frequency of the continuous-time input is varied.

For $f \ll f_{CK}/\pi$,

$$S_Y(f) \approx S_Q(f)(4\pi^2 f^2 T_{CK}^2), \quad (21.33)$$

indicating a parabolic shape at low frequencies. It is important not to confuse $S_Q(f)$ and $S_Y(f)$: the former is the unshaped noise generated by the quantizer while the latter denotes the shaped spectrum created by the $\Delta\Sigma$ loop and observed at the loop's output. As explained in Section 21.1, $S_Q(f) = \Delta^2/(12f_{CK})$, but we must write $S_Q(f) = 2\Delta^2/(12f_{CK})$ here because the spectra in Fig. 21.29 are one-sided.

Noise Shaping and Oversampling The foregoing studies elucidate the relation between noise shaping and oversampling in $\Delta\Sigma$ modulators. The key point emerging from Fig. 21.29(b) is that the narrower the analog input signal bandwidth, f_0 , the smaller the integrated Q-noise power. Conversely, for a given input bandwidth, we can *increase* f_{CK} , and hence the oversampling ratio, so as to reduce the Q-noise (Fig. 21.30).

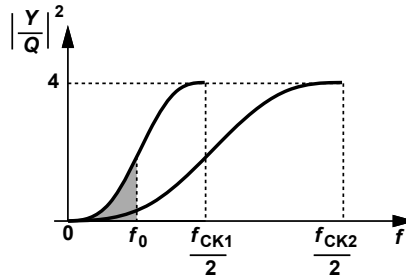


Figure 21.30 Effect of clock frequency on noise shaping.

Example 21.13

A student is skeptical about our results, reasoning that the “gain” of the DT integrator in Fig. 21.19 is simply equal to $-C_2/C_1$ and independent of f_{CK} . So why should f_{CK} have an effect on the $\Delta\Sigma$ modulator's behavior?

Solution

We must first reconcile the CT and DT models shown in Fig. 21.31. Suppose the CT integrator in Fig. 21.31(a) is represented by K/s , where $K > 1$. Then,

$$\frac{Y}{Q}(s) = \frac{1}{1 + \frac{K}{s}} \quad (21.34)$$

$$= \frac{s}{s + K} \quad (21.35)$$

$$\approx \frac{s}{K}, \quad (21.36)$$

if the frequencies of interest are much lower than K . Thus, a greater integrator gain provides higher Q-noise suppression.

Next, we turn to Fig. 21.31(b) and recall from Eq. (21.19) that $z^{-1}/(1 - z^{-1})$ can be viewed as $\exp(-sT_{CK})/(sT_{CK}) = \exp(-sT_{CK})(f_{CK}/s)$. That is, the DT integrator gain is, in fact, proportional to f_{CK} . We thus expect that a higher clock frequency translates to less Q-noise at the $\Delta\Sigma$ modulator output. This is, in fact, to be expected: a higher f_{CK} switches C_2 in Fig. 21.19(c) more frequently between V_{in} and C_1 , accumulating greater charge. Note that we have assumed a constant input signal bandwidth while f_{CK} is allowed to increase. We conclude that calling the factor $-C_2/C_1$ the integrator “gain” is not quite correct—but it is common in the field.

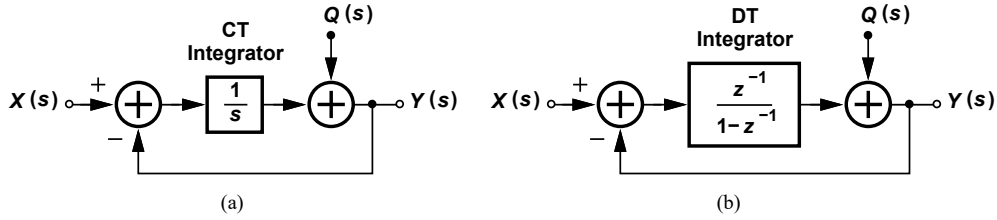


Figure 21.31 (a) Continuous-time, and (b) discrete-time $\Delta\Sigma$ loops.

Example 21.14

The loop shown in Fig. 21.32(a) incorporates a non-delaying integrator (Example 21.9). Determine Y in terms of X and Q .

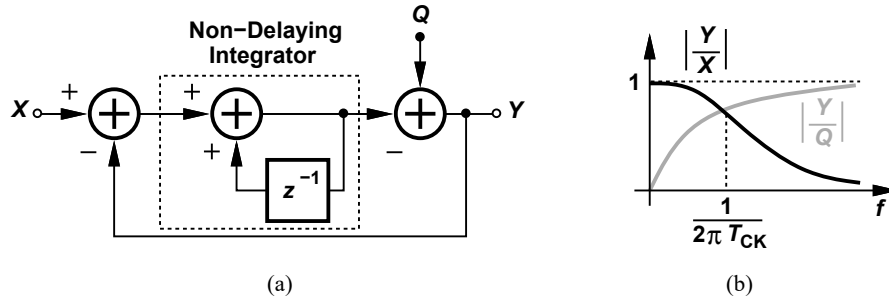


Figure 21.32 (a) $\Delta\Sigma$ loop using a non-delaying integrator, and (b) its frequency responses.

Solution

Using superposition, we have

$$Y = \frac{\frac{1}{1-z^{-1}}}{1 + \frac{1}{1-z^{-1}}} X + \frac{1}{1 + \frac{1}{1-z^{-1}}} Q \quad (21.37)$$

$$= \frac{1}{2-z^{-1}} X + \frac{1-z^{-1}}{2-z^{-1}} Q. \quad (21.38)$$

Unlike the modulator of Fig. 21.26(b), this circuit introduces a denominator equal to $2 - z^{-1}$. How do we interpret this? As an approximation, we write $2 - z^{-1} = 2 - e^{-sT_{CK}} \approx 2 - (1 - sT_{CK}) = 1 + sT_{CK}$, which corresponds to a pole at $f_p = -1/(2\pi T_{CK})$. Sketched in Fig. 21.32(b) are $|Y/X|$ and $|Y/Q|$. For a sufficiently high clock frequency, the effect of this pole proves negligible.

Example 21.15

For the first-order modulator of Fig. 21.33, compute the error, E , and the integrator output, A , in terms of X and Q .

Solution

Since $Y = z^{-1}X + (1 - z^{-1})Q$, we have $E = X - Y = (1 - z^{-1})(X - Q)$. That is, the error is equal to the derivative of $X - Q$. Similarly, $A = [z^{-1}/(1 - z^{-1})]E = z^{-1}(X - Q)$. The integrator output thus simply contains the difference between the input and the Q -noise but with a one-cycle delay.

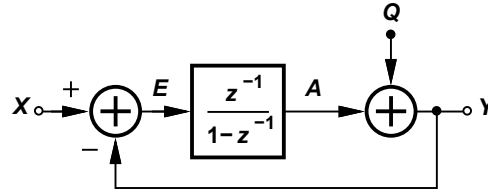


Figure 21.33 First-order discrete-time $\Delta\Sigma$ loop.

Example 21.16

A student finds the previous example's conclusion confusing. Why does the integrator output not contain the integral of X ?

Solution

This is due to the feedback loop action. Consider the simpler situation depicted in Fig. 21.34. The reader can readily prove that $A = (10/11)X + (1/11)Q$. We observe that X is *not* just multiplied by a factor of 10 as it travels to A .

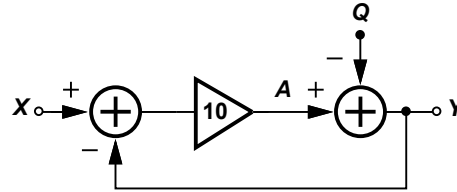


Figure 21.34 Simple loop containing an amplifier instead of an integrator.

21.3.3 Q-Noise Calculation

The first-order DT modulator of Fig. 21.26(b) shapes the quantizer's white noise by the functions plotted in Fig. 21.29. That is, $S_Y(f) = S_Q(f)[2 \sin(\pi f T_{CK})]^2$. To determine the total Q-noise corrupting the digitized output, we integrate $S_Y(f)$ across the input signal bandwidth, i.e., from zero to f_0 . Note that $(f_{CK}/2)/f_0$ is the oversampling ratio (OSR) and, for clarity of mathematical expressions, is denoted by the letter M . It is reasonable to use the approximation shown in Eq. (21.33). To this end, we recall from Section 21.1 that $S_Q(f) = \Delta^2/(12f_{CK})$ for $-f_{CK}/2 < f < +f_{CK}/2$. The total "in-band" Q-noise power is thus expressed

$$P_Q = \int_{-f_0}^{+f_0} \frac{\Delta^2}{12f_{CK}} (4\pi^2 f^2 T_{CK}^2) df \quad (21.39)$$

$$= \frac{2\pi^2 \Delta^2}{9} \left(\frac{f_0}{f_{CK}} \right)^3. \quad (21.40)$$

Since the oversampling ratio is equal to $f_{CK}/(2f_0)$, we have

$$P_Q = \frac{\pi^2 \Delta^2}{36M^3}. \quad (21.41)$$

This result illustrates the dependence of P_Q upon the OSR, motivating us to clock the integrator (and the quantizer) at high frequencies. Of course, our assumption is that the $\Delta\Sigma$ modulator's output is applied to a

low-pass filter having a corner frequency of f_0 so that the “out-of-band” Q-noise is removed (Chapter 24). It is also interesting to note that, according to our simple model, P_Q is independent of the analog input signal amplitude. In the next chapter, we prove that the overall output signal-to-noise ratio is given by

$$\text{SNR} = \frac{9M^3}{2\pi^2}. \quad (21.42)$$

21.4 Continuous-Time First-Order $\Delta\Sigma$ Modulator

The analysis and design presented in the previous section can be repeated for CT $\Delta\Sigma$ loops as well. In this case, the integrator is realized by the CT topology of Fig. 21.19(a), leading to the $\Delta\Sigma$ modulator shown in Fig. 21.35. Note that the sampling occurs *after* integration. In fact, the quantizer is typically formed by a flash stage and hence performs sampling itself. We should point out that, due to sampling, CT modulators still entail discrete-time action, i.e., the adjective “continuous-time” is not quite correct.

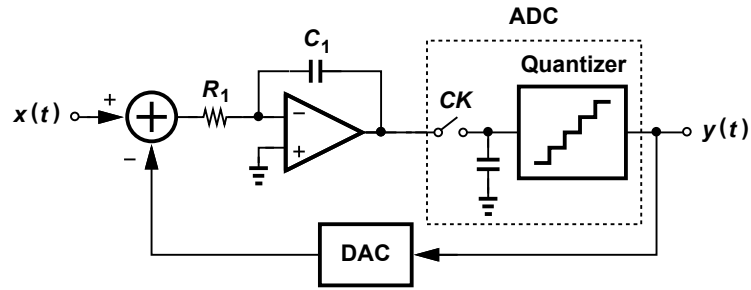


Figure 21.35 First-order continuous-time $\Delta\Sigma$ loop.

Most of the results obtained in the previous section also apply to CT $\Delta\Sigma$ loops. In particular, the system in Fig. 21.35 provides a first-order noise-shaping function given by $1 - z^{-1}$, yielding the in-band Q-noise power expressed by Eq. (21.41) and the SNR shown by Eq. (21.42). We delve into the design of DT and CT modulators in the next chapter. The advantages and disadvantages of DT and CT $\Delta\Sigma$ modulators are described in Section 21.6.2.

Note on Poles and Zeros The simple $\Delta\Sigma$ model shown in Fig. 21.18 implies that an integrator, i.e., a pole at the origin, yields an NTF having a *zero* at the origin. This can be generalized as follows. From Fig. 21.36, we have

$$\frac{Y}{Q}(s) = \frac{1}{1 + H(s)}. \quad (21.43)$$

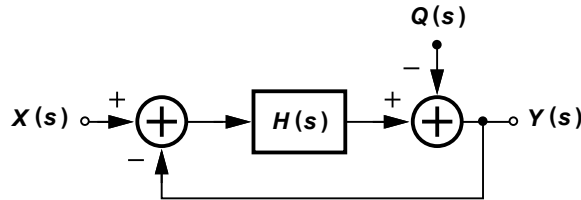


Figure 21.36 Basic loop for study of poles and zeros.

If $H(s) = N(s)/D(s)$, then

$$\frac{Y}{Q}(s) = \frac{D(s)}{D(s) + N(s)}, \quad (21.44)$$

revealing that the poles of $H(s)$ indeed become the zeros of the NTF. In $\Delta\Sigma$ modulator design, we sometimes place the NTF zeros at a number of frequencies so as to obtain a tailored noise-shaping function. We elaborate on this point in Problem 21.10.

21.5 Second-Order $\Delta\Sigma$ Modulators

21.5.1 Basic Idea

As explained in Section 21.3.2, first-order $\Delta\Sigma$ modulators exhibit a noise-shaping function, $1 - z^{-1}$, whose squared magnitude is proportional to f^2 at low frequencies. We have also seen that the integrated Q-noise power falls in proportion to the *third* power of f_{CK} and hence the oversampling ratio. We then ask, is it possible to reduce the Q-noise further?

Considering the first-order modulator studied in the previous sections, we surmise that Q-noise decreases if the quantizer is changed to a “finer” one [Fig. 21.37(a)]. For example, we can attempt to employ an 8-bit flash stage, but such a choice entails not just greater complexity, but other issues as well (Chapter 24). Alternatively, we can replace the quantizer with an ADC that naturally generates less Q-noise [Fig. 21.37(b)], for example, a $\Delta\Sigma$ modulator [Fig. 21.37(c)]. These thoughts lead to the architecture shown in Fig. 21.37(d), which “nests” one $\Delta\Sigma$ loop within another while still employing a simple (1-bit) quantizer. Interestingly, $W_1 = W_2$ (why?), allowing the two DACs to collapse into one [Fig. 21.37(e)].

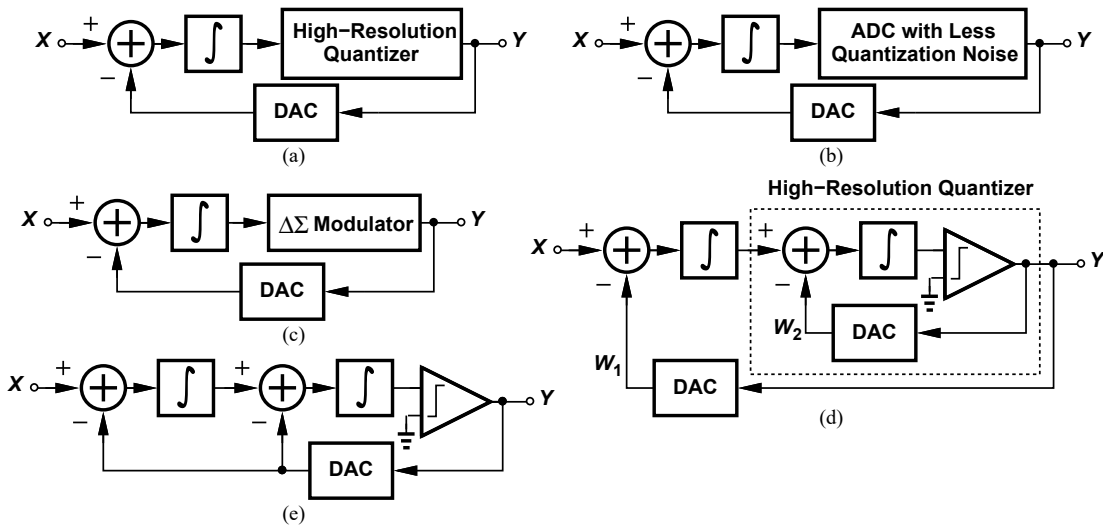


Figure 21.37 (a) $\Delta\Sigma$ using a high-resolution quantizer, (b) alternative view, (c) use of a $\Delta\Sigma$ as a quantizer, (d) realization of (c), and (e) simplified result.

The architecture of Fig. 21.37(e) is called a “second-order $\Delta\Sigma$ modulator” as it includes two integrators. In a manner similar to our analysis of the first-order counterpart, we wish to derive its properties here.

21.5.2 Transfer Functions

Let us first repeat the simple continuous-time analysis of Section 21.2.2 for the second-order loop. Shown in Fig. 21.38(a), the CT approximation yields $(X - Y)(1/s)$ at the output of the first integrator and $[(X - Y)(1/s) - Y](1/s)$ at the output of the second. Since $Y = Q + [(X - Y)(1/s) - Y](1/s)$, we have

$$Y(s) = \frac{1}{s^2 + s + 1} X(s) + \frac{s^2}{s^2 + s + 1} Q(s). \quad (21.45)$$

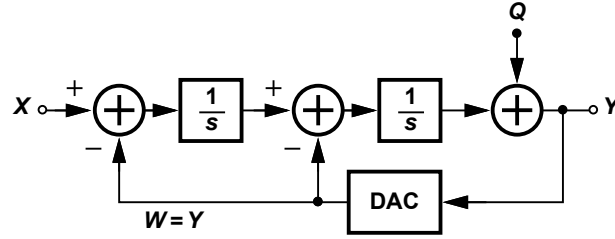


Figure 21.38 Equivalent block diagram of a second-order $\Delta\Sigma$ loop.

Thus, $X(s)$ experiences a low-pass response and $Q(s)$ a high-pass one, both of which can be verified by letting $s = j\omega$ approach zero or infinity. For the output Q-noise spectrum, we write

$$S_Y(f) = \left| \frac{Y}{Q}(s = j2\pi f) \right|^2 S_Q(f) \quad (21.46)$$

$$= \frac{(2\pi f)^4}{[1 - (2\pi f)^2]^2 + (2\pi f)^2} S_Q(f), \quad (21.47)$$

which, at low frequencies, reduces to

$$S_Y(f) \approx (2\pi f)^4 S_Q(f). \quad (21.48)$$

We thus expect second-order noise shaping in this case. Sketched in Fig. 21.39 is S_Y/S_Q for first- and second-order loops, revealing that the latter suppresses Q-noise to a greater extent up to some frequency.

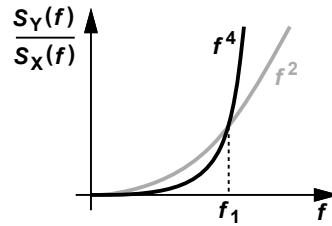


Figure 21.39 First- and second-order noise shaping.

We now construct the discrete-time model of the second-order $\Delta\Sigma$ modulator [Fig. 21.40(a)] and obtain the signal and noise transfer functions. The outputs of the first and second integrators emerge as $(X - Y)[z^{-1}/(1 - z^{-1})]$ and $\{(X - Y)[z^{-1}/(1 - z^{-1})] - Y\}[z^{-1}/(1 - z^{-1})]$, respectively. That is,

$$Y = \left[(X - Y) \frac{z^{-1}}{1 - z^{-1}} - Y \right] \frac{z^{-1}}{1 - z^{-1}} + Q \quad (21.49)$$

and hence

$$Y(z) = \frac{z^{-2}}{1 - z^{-1} + z^{-2}} X(z) + \frac{(1 - z^{-1})^2}{1 - z^{-1} + z^{-2}} Q(z). \quad (21.50)$$

Unfortunately, the denominator proves problematic as it leads to undesirable signal and noise transfer functions. The proof is beyond our scope here, but the remedy is fairly simple: we insert a gain of 2 in the inner feedback loop [Fig. 21.40(b)] [1]. It follows that

$$Y = \left[(X - Y) \frac{z^{-1}}{1 - z^{-1}} - 2Y \right] \frac{z^{-1}}{1 - z^{-1}} + Q \quad (21.51)$$

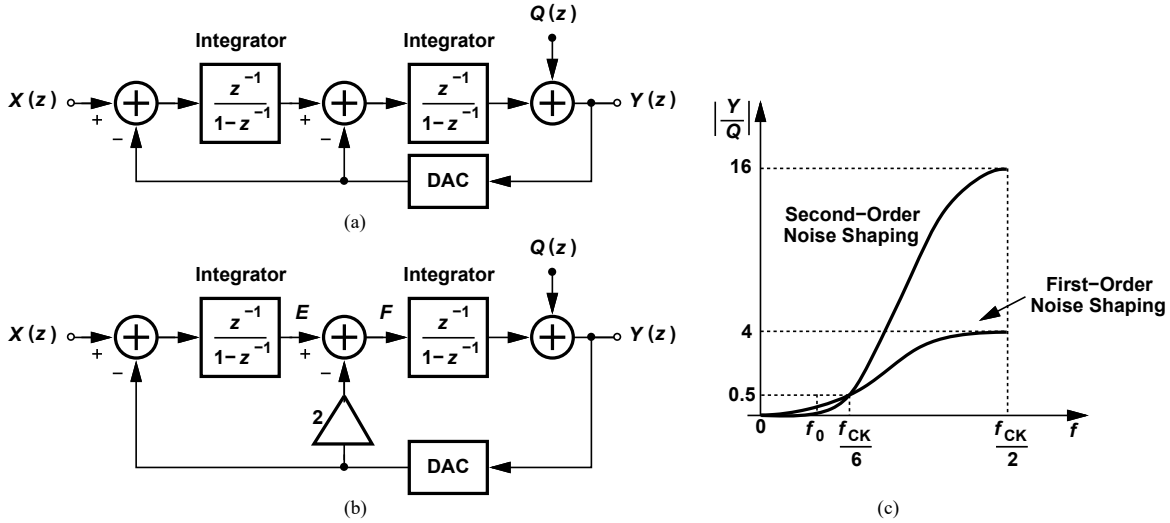


Figure 21.40 (a) $\Delta\Sigma$ using delaying integrators, (b) necessary modification, and (c) comparison of first- and second-order noise shaping.

and thus

$$\left[1 + \left(\frac{z^{-1}}{1 - z^{-1}} \right)^2 + 2 \frac{z^{-1}}{1 - z^{-1}} \right] Y = \left(\frac{z^{-1}}{1 - z^{-1}} \right)^2 X + Q. \quad (21.52)$$

The terms in the square brackets add up to $[z^{-1}/(1 - z^{-1})]^2$, producing

$$Y(z) = z^{-2}X(z) + (1 - z^{-1})^2Q(z). \quad (21.53)$$

The $\Delta\Sigma$ modulator therefore delays the input signal by two clock cycles while providing second-order noise shaping. Recall from Eq. (21.30) that $|1 - z^{-1}| = |2 \sin(\pi f T_{CK})|$. In this case, we have $S_Y(f) = |Y/Q|^2 S_Q(f) = |1 - z^{-1}|^4 S_Q(f) = 16 \sin^4(\pi f T_{CK})$. Figure 21.40(c) plots $|Y/Q|^2$ for first- and second-order loops, revealing that the latter provides greater suppression for Q-noise frequencies up to $f_{CK}/6$.

Example 21.17

Explain what second-order noise shaping means in the time domain.

Solution

We have $(1 - z^{-1})^2 Q(z) = (1 - 2z^{-1} + z^{-2})Q(z)$. Since z^{-1} signifies a one-cycle delay, this result suggests that the quantization error is processed as $q(t) - 2q(t - T_{CK}) + q(t - 2T_{CK})$.

The architecture of Fig. 21.40(b) implies that the DAC output must be *amplified* by a factor of 2 before reaching the second subtractor, a difficult task. But a simple transformation obviates this need. Since $F = E - 2Y = 2(E/2 - Y)$, we can scale E down by a factor of 2, subtract Y from the result, and scale the difference up by a factor of 2 [Fig. 21.41(a)]. We thus arrive at the modulator shown in Fig. 21.41(b), where the factors of 1/2 and 2 can readily be absorbed by the integrators' input capacitors. [Recall that the DT integrator of Fig. 21.19(c) provides a transfer function equal to $(-C_2/C_1)z^{-1}/(1 - z^{-1})$.]

Example 21.18

Determine the output of the first integrator in Fig. 21.41(b) in terms of X and Q .

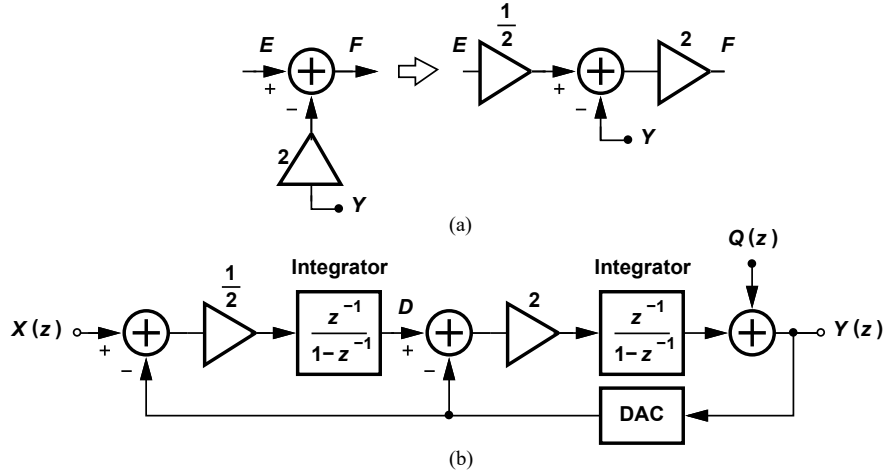


Figure 21.41 (a) Transformation of one topology to another, and (b) $\Delta\Sigma$ loop using only delaying integrators.

Solution

Since $Y(z) = z^{-2}X(z) + (1 - z^{-1})^2Q(z)$, the first subtractor output is given by

$$X(z) - Y(z) = (1 - z^{-2})X(z) - (1 - z^{-1})^2Q(z). \quad (21.54)$$

The first integrator thus generates

$$D(z) = \frac{1}{2}(1 + z^{-1})z^{-1}X(z) - \frac{1}{2}(1 - z^{-1})z^{-1}Q(z). \quad (21.55)$$

We show in Problem 21.14 that $1 + z^{-1}$ represents a low-pass filter. Thus, the first integrator output carries a low-pass-filtered and delayed copy of $X(z)$ and a first-order-shaped and delayed copy of $Q(z)$.

21.5.3 Total Quantization Noise

The in-band noise introduced by the quantizer in a second-order loop can be computed by integrating the product of $|Y/Q|^2$ in Fig. 21.40(c) and $S_Q(f)$ from $f = -f_0$ to $f = +f_0$ (the signal bandwidth). Approximating this function by $16(\pi f T_{CK})^4$, we have

$$P_Q = \int_{-f_0}^{+f_0} \frac{\Delta^2}{12f_{CK}} (16\pi^4 f^4 T_{CK}^4) df \quad (21.56)$$

$$= \frac{8\pi^4 \Delta^2}{15} \left(\frac{f_0}{f_{CK}} \right)^5. \quad (21.57)$$

The oversampling ratio, M , is given by $f_{CK}/(2f_0)$, yielding

$$P_Q = \frac{\pi^4 \Delta^2}{60M^5}. \quad (21.58)$$

The total Q-noise drops in proportion to the *fifth* power of the OSR. For this and other reasons (Chapter 23), second-order modulators are much more popular than first-order loops. We prove in the next Chapter that

$$SNR = \frac{15M^5}{2\pi^4}. \quad (21.59)$$

21.6 Advantages of Oversampling

21.6.1 General Advantages

We have seen in previous sections that oversampling and noise shaping can dramatically reduce the in-band Q-noise accompanying the digital output of $\Delta\Sigma$ modulators. It is therefore possible to achieve high signal-to-noise ratios by clocking the loop at great rates—provided that the converter’s building blocks operate properly. For example, oversampling affords resolutions as high as 20 bits for high-quality audio applications, an extremely difficult challenge for Nyquist-rate converters.

Another advantage often attributed to oversampling relates to the anti-aliasing filter that must precede any ADC. In Nyquist-rate architectures, to ensure that unwanted signals and out-of-band noise are not aliased, such a filter limits the input bandwidth to slightly greater than that of the desired signal [Fig. 21.42(a)]. The filter must therefore provide a sharp roll-off beyond f_0 so as to suppress the undesirable components. With oversampling, on the other hand, the filter design is greatly relaxed [Fig. 21.42(b)] because the sampling rate, $f_S = 2Mf_0$, is much higher than f_0 , and only components above Mf_0 can alias. Of course, although not aliased, the unwanted components may not be suppressed by the filter. The suppression of these components is thus deferred to the digital filter that appears after the ADC and limits the bandwidth to approximately f_0 .

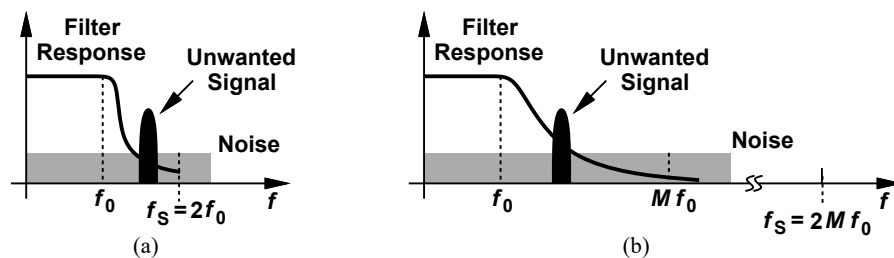


Figure 21.42 Antialiasing filter for (a) Nyquist-rate, and (b) oversampling ADCs.

Example 21.19

Consider the scenario depicted in Fig. 21.42(b), where the unwanted signal (the “interferer”) is far stronger than the desired component. How should the input voltage range (the full scale) of the oversampling ADC be chosen in this case?

Solution

The full scale must be large enough to accommodate the interferer. This is because the ADC must not saturate or become nonlinear while digitizing both signals. It can be shown that nonlinearity arising from the interferer may corrupt the desired signal. The key point here is that the relaxed anti-aliasing filter, in fact, makes the oversampling ADC design *more* difficult and hence is not a significant advantage. In practice, this issue may require the filter to be as selective as in Fig. 21.42(a).

The third advantage of oversampling by $\Delta\Sigma$ modulation is that the performance is less sensitive to analog circuit imperfections. As analyzed in Chapter 23, issues such as comparator offset, op amp gain, settling time, and device mismatches prove less stringent here than in Nyquist-rate ADCs.

21.6.2 CT Modulator Advantages

We have briefly considered a CT modulator architecture in Section 21.4 and in Fig. 21.35, but we can still predict its advantages over its DT counterparts. First, this modulator requires no front-end sampling. In fact, the clocking action within the comparator can perform sampling at the integrator output. Thus, bootstrapped samplers are generally not necessary.

Second, the absence of switching makes the input of the CT circuit “quieter.” That is, the circuit draws smaller transient currents from its input, thereby relaxing the design of the buffer that would precede the ADC. Third, the first integrator removes some of the out-of-band input frequency components and hence acts as a partial anti-aliasing filter. This eases the design of the filter that must precede the ADC.

21.7 Disadvantages of Oversampling

The principal disadvantage of oversampling stems from the need for clocking the circuit at much higher speeds than the Nyquist rate. With realistic integrator and quantizer designs, the maximum clock frequency is limited, and so is the input signal bandwidth. For this reason, oversampling proves a good candidate for low-speed high-resolution applications, typically for signal bandwidths up to a few tens of megahertz.

Another disadvantage of oversampling relates to the digital filter that must follow the ADC so as to remove the out-of-band Q-noise (Chapter 24). The filter poses significant additional work for the designer and may introduce a long latency in the data path (Chapter 24).

The CT modulator described in Section 21.4 presents its own disadvantages, which we study in Chapter 22.

21.8 Performance Metrics

Our studies thus far reveal that the digital output of a $\Delta\Sigma$ modulator tracks the analog input closely but it also contains high-frequency Q-noise. In other words, only the “running average” of the output follows the input. The use of an integrator in the loop creates memory for this averaging action, and the memory effect means that there is no one-to-one correspondence between the instantaneous input and output values.¹⁰ For this reason, static ADC metrics such as DNL and INL are not meaningful (or useful) here. We therefore rely on other metrics.

Let us ask how the output signal-to-noise ratio of a $\Delta\Sigma$ loop varies as the analog input swing increases. Given that P_Q in Eq. (21.58) is independent of the input, we expect that a small input amplitude produces a low SNR. We also predict that an excessively large input amplitude leads to nonlinearity in the integrator, dropping its open-loop gain and hence the SNR. Another cause of SNR degradation is the reduction of the quantizer gain as the integrator swings increase, as explained in the next section. From these points emerges the general SNR plot shown in Fig. 21.43. For a sufficiently low input swing, V_{in0} , the SNR is equal to unity. Due to nonlinearity, the maximum (or peak) SNR, SNR_{max} , is less than the ideal value, SNR_i . We call V_{in1} the “maximum stable amplitude.”

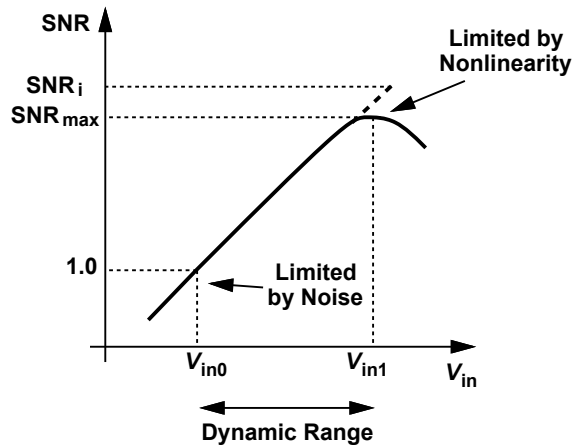


Figure 21.43 Dynamic range of a $\Delta\Sigma$ ADC.

¹⁰This is similar to the behavior of a simple RC circuit: the output depends on both the present and the past values.

We should remark that Fig. 21.43 plots only the signal-to-noise ratio, excluding the distortion. The fall of SNR beyond V_{in1} is due to the degraded shaping of Q-noise, which occurs because the integrator experiences large voltage swings and does not follow the $1/s$ or $1/(1 - z^{-1})$ behavior.

Another metric is the “dynamic range” (DR), defined as the *input* range across which the SNR rises from unity to its maximum, i.e., V_{in1}/V_{in0} in Fig. 21.43.

Example 21.20

Explain why the dynamic range of oversampling ADCs is typically greater than their maximum SNR.

Solution

As illustrated in Fig. 21.43, owing to nonlinearity, large input swings do not translate to proportionally higher SNRs, i.e., the SNR does not reach its ideal value at $V_{in} = V_{in1}$. For example, a $\Delta\Sigma$ modulator may exhibit a DR of 80 dB but a peak SNR of only 75 dB.

The third metric quantifying the performance of $\Delta\Sigma$ loops is the signal-to-(noise + distortion) ratio (SNDR), defined in the same manner as explained in Chapter 12 for Nyquist-rate ADCs. That is, at any input level between V_{in0} and V_{in1} in Fig. 21.43, we can compute $SNDR = P_{sig}/(P_{sig} + P_{dist})$. We predict that $SNDR < SNR_{max} < DR$. One can also plot the SNDR versus V_{in} and observe a similar behavior to that in Fig. 21.43.

21.9 Properties of 1-Bit Quantizers

$\Delta\Sigma$ modulators often employ a 1-bit (binary) quantizer, i.e., a single comparator. Illustrated in Fig. 21.36(e), such an architecture avoids certain DAC imperfections (Chapter 23), leading to a simple design. This quantizer, however, exhibits a number of interesting attributes that differ from those studied in Section 21.1 for multibit topologies.

An ideal 1-bit quantizer generates a logical ZERO for a negative input and a logical ONE for a positive input. It is more convenient to represent these logical values by -1 and $+1$, respectively (Fig. 21.44). The output levels of -1 and $+1$ can be visualized by assuming that the quantizer drives a 1-bit DAC, and the DAC delivers these values in response to the digital output of the quantizer. We must address two important questions here: (1) how much is the quantizer’s output noise? and (2) how much is the quantizer’s gain?

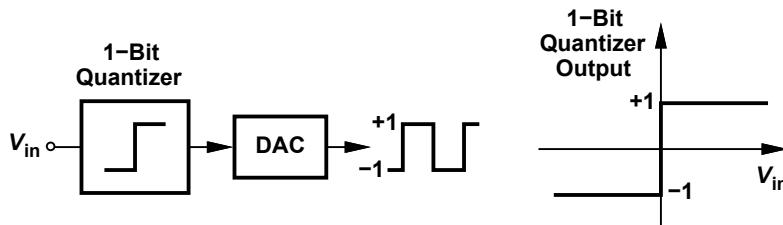


Figure 21.44 Ideal 1-bit quantizer characteristic.

The first question cannot be answered from Section 21.1 because binary quantization does not entail an analog LSB, Δ ! Thus, the Q-noise expression $\Delta^2/12$ appears meaningless here. Nevertheless, it is shown in [2] that the Q-noise is given by $A^2/12$, where A denotes the total *height* of the quantizer’s characteristic. For example, the height of 2 in Fig. 21.44 translates to a total noise of $2^2/12 = 1/3$ and a spectral density of $(1/3)/(f_{CK})$. This is not always accurate, as illustrated by the following example.

Example 21.21

A sinusoid, $x(t) = A \sin \omega t$, is applied to a quantizer with output levels equal to $\pm A$. Determine the quantization error power and compare the result to the theoretical value of $(2A)^2/12 = A^2/3$.

Solution

As shown in Fig. 21.45(a), the quantizer generates a square wave, $y(t)$, in response to this input. The error is equal to $y(t) - x(t)$ and plotted in Fig. 12.8(b). The associated power is given by

$$P_e = \frac{2}{T} \int_0^{T/2} (A - A \sin \omega t)^2 dt. \quad (21.60)$$

It follows that

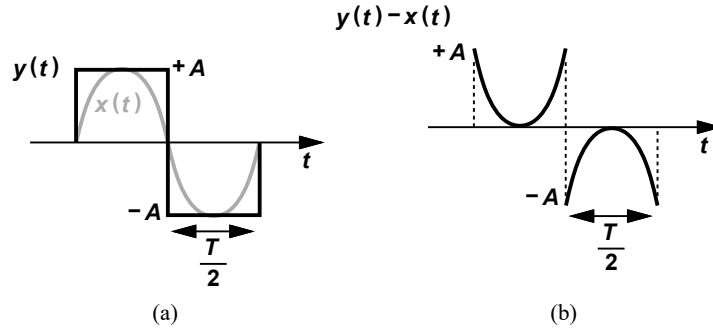


Figure 21.45 (a) Quantized sine wave, and (b) the resulting error.

$$P_e = \left(\frac{3}{2} - \frac{4}{\pi} \right) A^2 \quad (21.61)$$

$$\approx \frac{A^2}{4.4}. \quad (21.62)$$

Thus, the theoretical value overestimates the “noise” power in this case.

The second question relates to the gain of the quantizer, G_Q , predicted to be infinity by the abrupt characteristic of Fig. 21.44 at $x = 0$ and zero elsewhere. But the actual situation is more complex. Let us begin with the scenario depicted in Fig. 21.46(a), where $x(t) = B \sin \omega t$ is applied to a quantizer having output levels equal to $\pm A$. Suppose $B < A$ for now. How do we define the gain here? One interpretation is to seek this sinusoid in the output, determine its peak amplitude, and divide it by B . Since the output square wave, $y(t)$, contains a fundamental with an amplitude equal to $4A/\pi$, we can conclude that the quantizer gain is given by $G_Q = (4A/\pi)/B = (4/\pi)(A/B)$. This perspective reveals that the gain *falls* as the input amplitude rises [Fig. 21.46(b)]. The 1-bit quantizer thus presents interesting modeling challenges.

We now assume the quantizer input is a random signal [Fig. 21.46(c)], noting that the output toggles between $-A$ and $+A$ while the zero crossings of $y(t)$ coincide with those of $x(t)$. In analogy with the behavior depicted in Fig. 21.46(b), we say the gain of the quantizer fluctuates with the input level—even though we do not have an exact definition of gain for this case yet. For example, at $t = t_1$, $x(t)$ exhibits a low swing while $y(t)$ reaches a value of $+A$, whereas at $t = t_2$, $x(t)$ is greater and $y(t)$ is still equal to $+A$. Sketched in Fig. 21.46(c), $y(t)/x(t)$ varies from high values to low values.

The key observation here is that the quantizer’s gain depends on the input amplitude statistics: if $x(t)$ exhibits small fluctuations most of the time, then the gain is relatively “high” and vice versa. Thus, the behavior of a 1-bit quantizer in a $\Delta\Sigma$ loop is given by the integrator’s output swing, which itself contains the integral of the shaped Q-noise (reproduced by the DAC). As a starting point, we may select a unity gain for the quantizer, but we must eventually examine the performance of the loop with various types of input signals. The reader is referred to the literature for details.

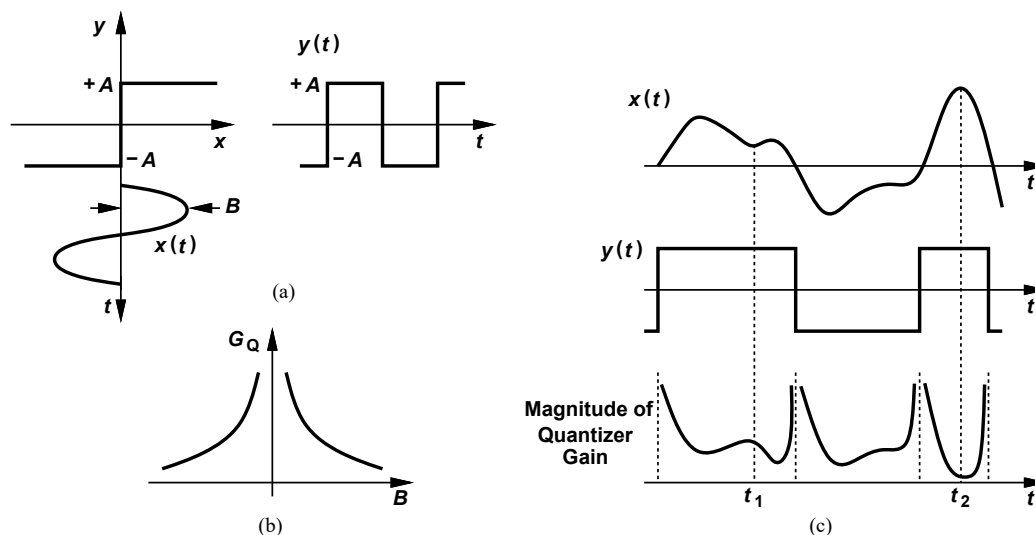


Figure 21.46 (a) Input and output waveforms of a 1-bit quantizer, (b) illustration of the circuit's gain, and (c) quantizer waveforms for a random input.

References

- [1]. B. Boser and B. A. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 1298-1308, Dec. 1988.
- [2]. S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Transactions on Circuits and Systems*, vol. CAS-34, no. 6, pp. 593-603, June 1987.

Problems

- 21.1. Explain what happens to the spectrum in Fig. 21.2(b) if the input sinusoid is shifted up by a small amount (i.e., if the input has a finite dc value).
- 21.2. Determine the NTF and STF in Fig. 21.13 if
 - (a) $H(s)$ is a “lossy” integrator and equal to $1/(s + \omega_1)$.
 - (b) $H(s)$ consists of two integrators and equal to $1/s^2$.
- 21.3. Repeat Example 21.5 if the loop employs a double integrator with a transfer function given by $1/s^2$.
- 21.4. Assuming that $V_{in} = V_0 \cos \omega_1 t$ in Fig. 21.19(c), plot the output waveform if the clock frequency is five times the input frequency. Verify that the output is a sine.
- 21.5. If $C_1 \neq C_2$ in Fig. 21.22, how do we modify the corresponding model in Fig. 21.24?
- 21.6. Derive the NTF and the STF of Fig. 21.26(b) if $C_1 \neq C_2$ in Fig. 21.26(a).
- 21.7. Expressing the noise-shaping function of Fig. 21.29(b) as $[2 \sin(\pi f T_{CK})]^2$, determine the increase in Q-noise if f_0 increases from $f_{CK}/16$ to $f_{CK}/8$. If we approximate this function by $(2\pi f T_{CK})^2$, do we obtain fairly accurate results for this calculation?
- 21.8. Determine the STF and the NTF in Fig. 21.32(a) if the integrator is preceded by an amplifier having a gain of A_0 . (This is equivalent to allowing a gain of A_0 for the integrator.)
- 21.9. Equation (21.42) suggests that a high SNR can be achieved by increasing the oversampling ratio, M . Suppose we wish to digitize a 100-MHz sinusoid with an SNR of 74 dB (corresponding to a resolution of 12 bits). Compute the necessary clock frequency. Is that a reasonable value?
- 21.10. Suppose we have $H(s) = 1/[s(s + a)]$ in Fig. 21.36. We expect the NTF to exhibit zeros at $s = 0$ and $s = -a$. Determine Y/Q and sketch its magnitude.
- 21.11. Determine Y/X and Y/Q in Fig. 21.38 if the transfer functions of the first and second stages are equal to a/s and b/s , respectively. Can we select a and b such that $Y/X = 1$?
- 21.12. Suppose the transfer functions of the first second integrators in Fig. 21.40(a) are changed to $\alpha/(1 - z^{-1})$ and $\beta z^{-1}/(1 - z^{-1})$, respectively. (The first one is now a non-delaying integrator.) Can we select α and β such that $Y/X = z^{-1}$?
- 21.13. Repeat the previous problem if the first and second integrators are expressed as $\alpha z^{-1}/(1 - z^{-1})$ and $\beta/(1 - z^{-1})$, respectively.
- 21.14. We wish to show intuitively that $1 + z^{-1}$ represents a low-pass filter. Suppose the input, $x(t) = A \cos \omega_0 t$, and the delay corresponding to z^{-1} is given by ΔT . Expand $A \cos \omega_0 t + A \cos[\omega_0(t + \Delta T)]$ and find the amplitude of the overall result. (Hint: the amplitude of $a \cos \theta + b \sin \theta$ is equal to $\sqrt{a^2 + b^2}$.) Prove that the amplitude is maximum for $\omega_0 = 0$ and falls as ω_0 increases, signifying a low-pass response.
- 21.15. A student mistakenly omits the gain of 2 preceding the second integrator in Fig. 21.41(b). Determine Y/X and Y/Q .
- 21.16. Determine the output of the second subtractor in Fig. 21.41(b) in terms of X and Q .
- 21.17. In Fig. 21.1(a), we select $x(t) = 2A \sin \omega t$. Determine the quantization error power.
- 21.18. In Fig. 21.1(a), we select $x(t)$ to be a triangular waveform having a peak-to-peak swing of $2A$. Determine the quantization error power and compare it to that obtained in the previous problem.
- 21.19. In Fig. 21.46(a), we continue to increase B . Find an equation for the gain plotted in Fig. 21.46(b).
- 21.20. Sketch a waveform for $x(t)$ in Fig. 21.46(c) such that the quantizer gain falls close to zero at some points in time.

Basic Design of $\Delta\Sigma$ Modulators

With the fundamental understanding developed in Chapter 21, we can now study the circuit-level design of fully-differential $\Delta\Sigma$ modulators. We wish to construct loops that *function* correctly but we are not concerned with their *performance* yet. We optimize $\Delta\Sigma$ modulators in Chapter 24, after we analyze their imperfections in Chapter 23.

We begin our endeavor with discrete-time first-order loops employing 1-bit quantizers. Next, we extend this work to second-order modulators. We then repeat our design effort for continuous-time converters. The reader is encouraged to review Chapters 3 and 5 before continuing here.

22.1 First-Order DT Loop Design

22.1.1 Improved DT Integrator

The DT integrator introduced in Chapter 21 suffers from several drawbacks, including input-dependent charge injection and nonlinearity due to switch junction capacitances. A topology avoiding these issues is shown in Fig. 22.1(a). In the sampling mode, S_1 and S_3 are on and C_2 senses V_{in} [Fig. 22.1(b)]. In the integration mode [Fig. 22.1(c)], S_2 and S_4 are on, allowing C_2 to deliver its charge to C_1 through the virtual ground, X . An alternate perspective is to recognize the circuit in Fig. 22.1(c) as an *amplifier* (Chapter 17) and note that a voltage change of $\Delta V = 0 - V_{in}$ on the left plate of C_2 is multiplied by $-(C_2/C_1)$ as it travels to the output, yielding $V_{out} = (C_2/C_1)V_{in}$. This output voltage change is added to the value stored on C_1 in the previous cycle.

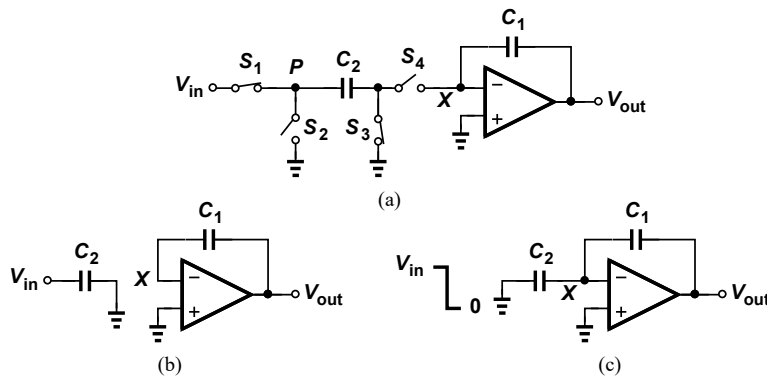


Figure 22.1 (a) Improved discrete-time integrator, (b) its equivalent in sampling mode, and (c) its equivalent in integration mode.

The problem of input-dependent charge injection is solved by turning S_3 in Fig. 22.1(a) off *before* S_1 at the end of the sampling mode. Such “bottom-plate” sampling (Chapter 16) ensures that the input-dependent

charge injected by S_1 does not corrupt the signal. The switch junction capacitance nonlinearity is suppressed, as shown in the following example.

Example 22.1

In the integrator of Fig. 22.1(a), explain why the junction capacitances of S_1 and S_2 play no role and those of S_3 and S_4 have little effect.

Solution

The left plate of C_2 is driven by either V_{in} or ground. As illustrated in Fig. 22.2(a), the switch junction capacitance, C_j , at P does not affect the charge on C_2 . As for the capacitance of S_3 and S_4 , $C_{j3,4}$, we first note from Fig. 22.2(b) that, in the sampling mode, R_{on3} is low enough to maintain V_X near zero, overwhelming $C_{j3,4}$. Depicted in Fig. 22.2(c), the integration mode also forces V_X to return to near zero so as to maintain a virtual ground, minimizing the charge on $C_{j3,4}$. This capacitance therefore exchanges negligible charge with the rest of the circuit.

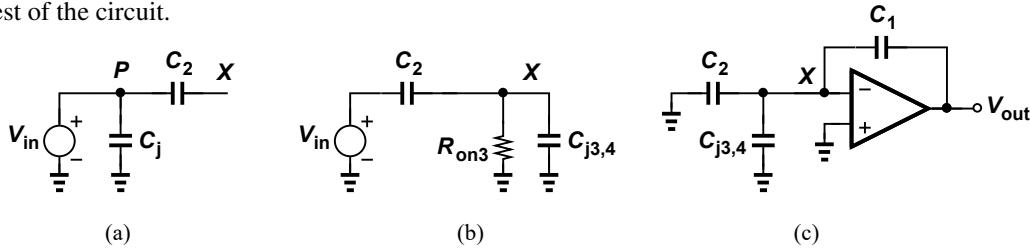


Figure 22.2 Effect of switch junction capacitance at (a) node P , (b) at node X in sampling mode, and (c) at node X in integration mode.

The Need for Nonoverlapping Clocks The timing of the clocks that drive the switches in Fig. 22.1(a) merits attention. We expect that S_1 and S_3 operate on the same clock edge, and so do S_2 and S_4 . But the question is, what happens if the clocks exhibit finite rise and fall times? During the clock transitions, it is possible that all four switches are on for a brief period of time, yielding the situation shown in Fig. 22.3(a). The voltage on C_2 is thus not a faithful replica of V_{in} due to voltage division between R_{on1} and R_{on2} . Also, V_{in} sees a low resistance to ground during this overlap time.¹ For these reasons, DT integrators incorporate *nonoverlapping* clocks [Fig. 22.3(b)]. To avoid input-dependent charge injection, the edge that turns off S_1 must occur *after* the one turning off S_3 [Fig. 22.3(c)]. The generation of these phases and the choice of the nonoverlap time, ΔT , are described in Section 22.5.

Differential Integrator We can readily extend the integrator topology of Fig. 22.1(a) to a differential structure (Fig. 22.4). One important point here is that S_{3a} and S_{3b} must connect to a common-mode voltage, V_{CM} , that is appropriate for the op amp, even though the latter does not see V_{CM} directly. We elaborate on this point in the following example.

Example 22.2

Explain how the op amp in Fig. 22.4 behaves if $V_{CM} = 0$.

Solution

In the sampling mode, C_2 stores a voltage equal to $V_{in}^+ - V_{CM} = V_{in}^+$. In the integration mode, C_2 must deliver *all* of its charge to C_1 , requiring $V_X \approx 0$, a CM level ill-suited to the op amp input [Fig. 22.5(a)]. The op amp's common-mode feedback loop now attempts to rectify the situation, possibly degrading the settling

¹The path from V_{in} to C_1 may corrupt the integrated voltage as well.

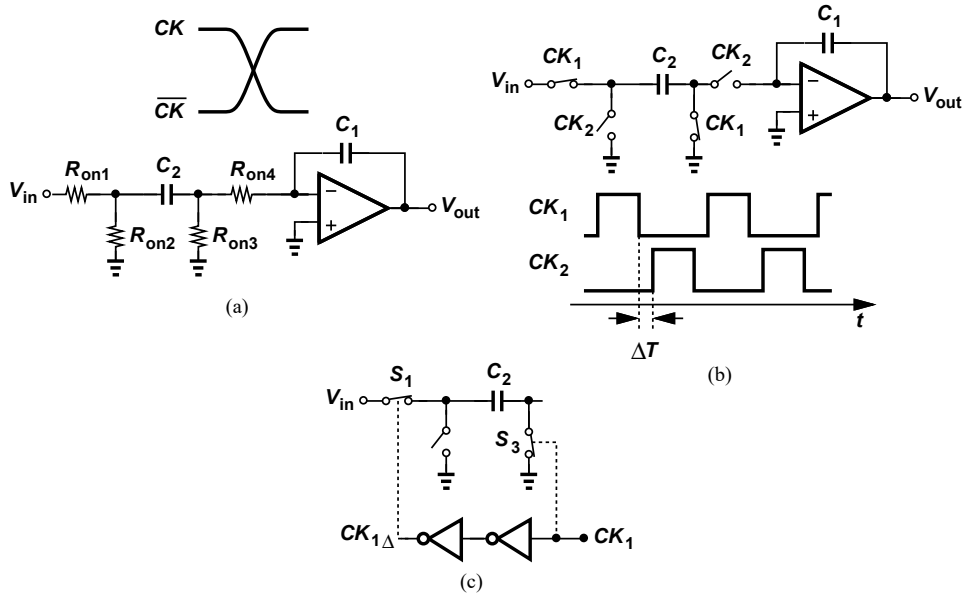


Figure 22.3 (a) Equivalent integrator circuit during overlap time of clocks, (b) use of nonoverlapping clocks, and (c) use of a delayed clock for S_1 .

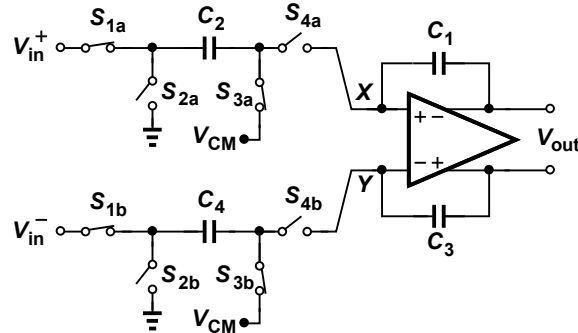


Figure 22.4 Differential discrete-time integrator.

speed. If, on the other hand, V_{CM} is chosen properly, C_2 samples a voltage equal to $V_{in}^+ - V_{CM}$ and ends with $V_X = V_{CM}$ [Fig. 22.5(b)] (why?), minimally disturbing the op amp's input CM level.

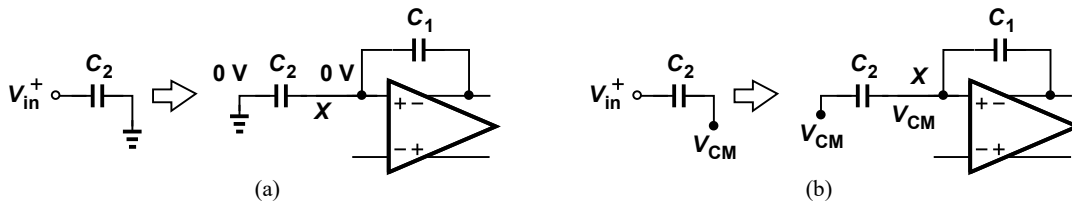


Figure 22.5 Use of (a) a zero voltage, or (b) a finite CM voltage for the right plate of C_2 .

Inclusion of Subtraction Recall from Chapter 21 that a $\Delta\Sigma$ loop includes an input subtractor. This function can be readily added to the integrator topology of Fig. 22.1(a). As shown in Fig. 22.6, we simply tie the bottom terminal of S_2 to the voltage that must be subtracted from the input. Consequently, during the transition from

sampling to integration, the left plate of C_2 swings from V_{in} to V_F . Viewing the circuit as an amplifier, we write the change in V_{out} as $(V_F - V_{in}) \times (-C_2/C_1) = (V_{in} - V_F)(C_2/C_1)$. That is, V_{out} rises by an amount equal to $(C_2/C_1)(V_{in} - V_F)$.

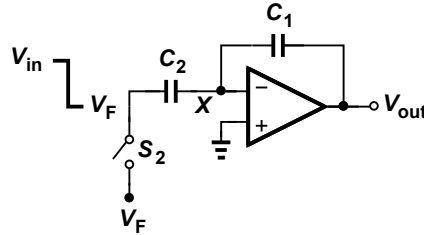


Figure 22.6 Inclusion of subtraction in integrator.

22.1.2 First-Order Loop with 1-Bit Quantizer

Our foregoing studies lead to the differential $\Delta\Sigma$ loop design depicted in Fig. 22.7(a), where $C_2 = C_4$ and $C_1 = C_3$. The integrator samples the input, computes the difference between $V_{in}^+ - V_{in}^-$ and $V_{DAC}^+ - V_{DAC}^-$, and accumulates proportional charge on C_1 and C_3 . After the integrator output settles, the comparator, Comp_1 , is clocked, commanding the DAC to produce a proper binary output for subtraction from the input in the next cycle. As explained in Chapter 5, Comp_1 employs an RS latch so as to provide a valid logical output when it is in the precharge mode. As with any ADC, the circuit must rely on a full-scale reference voltage to

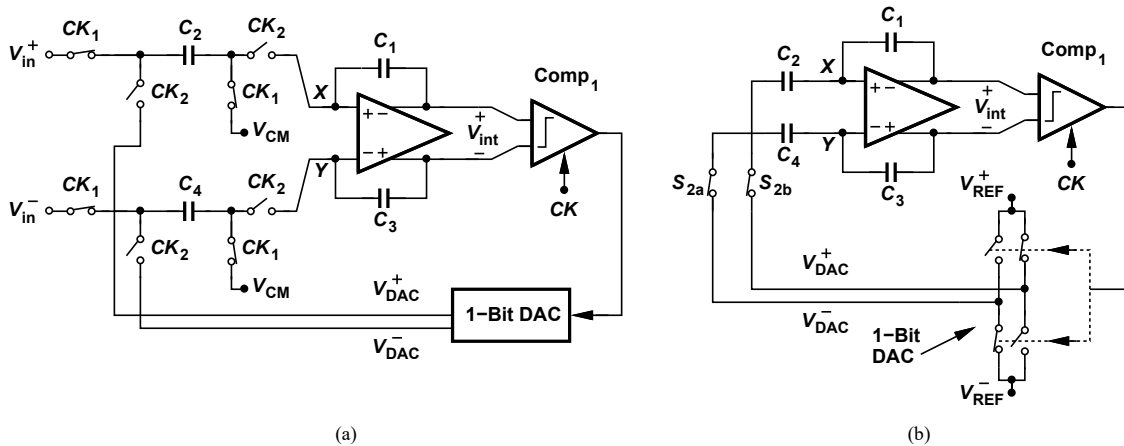


Figure 22.7 (a) Basic first-order loop, and (b) loop in the subtraction mode.

define its input range. In this case, the 1-bit DAC simply generates voltages equal to V_{REF}^+ and V_{REF}^- and sets the range of $V_{in}^+ - V_{in}^-$. For example, $V_{REF}^+ = 0.8$ V and $V_{REF}^- = 0.2$ V.

We must answer two questions at this point. (1) How is the 1-bit DAC implemented? (2) How is the comparator clock phase chosen with respect to those of the integrator? The DAC circuit is shown in Fig. 22.7(b). Depending on the comparator's decision, the DAC provides a voltage equal to V_{REF}^+ for C_2 and V_{REF}^- for C_4 or vice versa. In every cycle, therefore, C_2 and C_4 sense a net input equal to $(V_{in}^+ - V_{in}^-) - (V_{REF}^+ - V_{REF}^-)$ or $(V_{in}^+ - V_{in}^-) - (V_{REF}^- - V_{REF}^+)$. The reader is encouraged to ponder whether S_{2a} and S_{2b} can be removed here, i.e., whether they can be merged with the DAC switches.

Example 22.3

Determine the change in V_{int} in Fig. 22.7(a) after the left plates of C_2 and C_4 swing to V_{DAC}^+ and V_{DAC}^- , respectively.

Solution

From the previous section, we recall that the integrator output rises by an amount equal to $(C_2/C_1)(V_{in}^+ - V_{DAC}^+)$. With differential operation, V_{int} increases by $(C_2/C_1)[V_{in}^+ - V_{in}^- - (V_{DAC}^+ - V_{DAC}^-)]$. If Comp_1 delivers a high output, then the DAC produces $V_{DAC}^+ > V_{DAC}^-$, thus *lowering* V_{int} . This is because $V_{in}^+ - V_{in}^- < V_{REF}^+ - V_{REF}^-$.

It is important to note that the feedback around the $\Delta\Sigma$ loop in Fig. 22.7(b) must be negative: if Comp_1 determines that $V_{int} > 0$, then the loop must reduce V_{int} , i.e., a DAC value of $V_{REF}^+ - V_{REF}^-$ must be subtracted from $V_{in}^+ - V_{in}^-$. Conversely, if $V_{int} < 0$, then the DAC output is set to $V_{REF}^- - V_{REF}^+$. We are tempted to say that the $\Delta\Sigma$ modulator's differential input voltage range is equal to $2(V_{REF}^+ - V_{REF}^-)$, but the actual situation is more complex (Chapter 23).

For the comparator clock phase, we observe that the integrator output in Fig. 22.7(b) must be quantized only after it has settled. Consider the phases and waveforms shown in Fig. 22.8, where CK_1 controls sampling and CK_2 , subtraction and integration. The falling edge of CK_2 can clock Comp_1 . After $t = t_3$, the

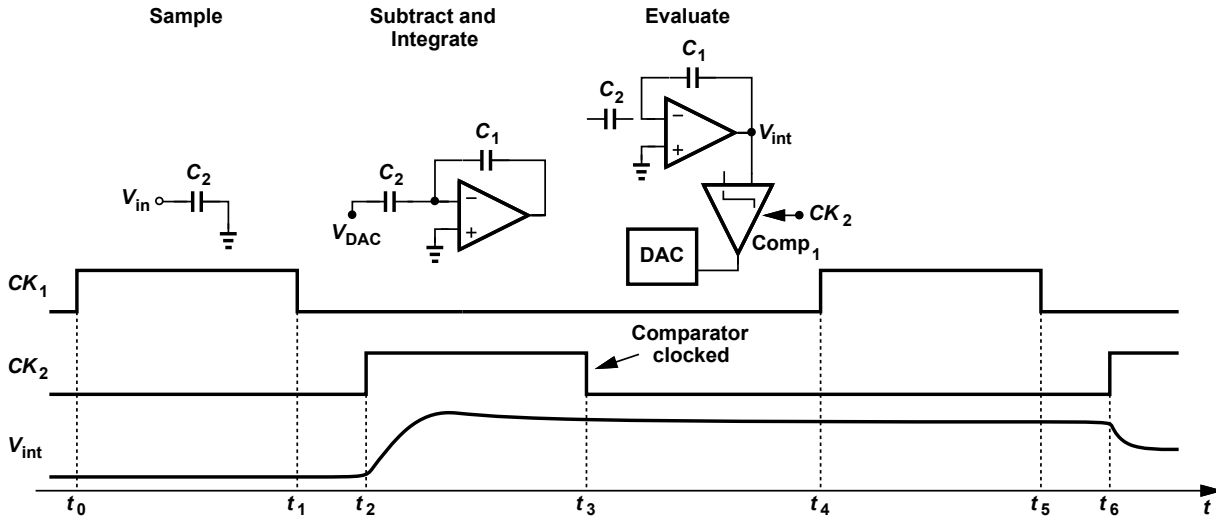


Figure 22.8 First-order loop waveforms.

comparator updates the DAC outputs, but since C_2 is no longer attached to the op amp input, V_{int} remains unchanged, i.e., the integrator is idle. The new DAC value is then subtracted from the *next* input sample, as prescribed by the principle of noise shaping. Note that Comp_1 enters the precharge mode when CK_2 goes high,² finding sufficient time to reset its nodes.

We recognize that after $t = t_3$, C_2 can be connected to the input. That is, we need not allow much time between t_3 and t_4 . The reader is encouraged to show that Comp_1 can be clocked on the rising edge of CK_1 .

Figure 22.9 summarizes our findings. The input capacitor samples V_{in} from $t = t_0$ to $t = t_1$ while Comp_1 decides and updates the DAC outputs; from $t = t_2$ to $t = t_3$, the integrator subtracts the DAC value from the sampled input and completes its settling while Comp_1 is precharged. Thus, V_{int} has less than half of a clock cycle to settle before Comp_1 is activated.

Loop Waveforms It is instructive to sketch the $\Delta\Sigma$ modulator's waveforms for different types of inputs. For the sake of simplicity, we reduce the circuit of Fig. 22.7(b) to the single-ended form shown in Fig. 22.10(a), where $C_2 = C_1$ and V_{in} can range from $-V_{REF}$ to $+V_{REF}$. How do we ensure that the feedback around this

²We assume Comp_1 is realized by a StrongArm latch.

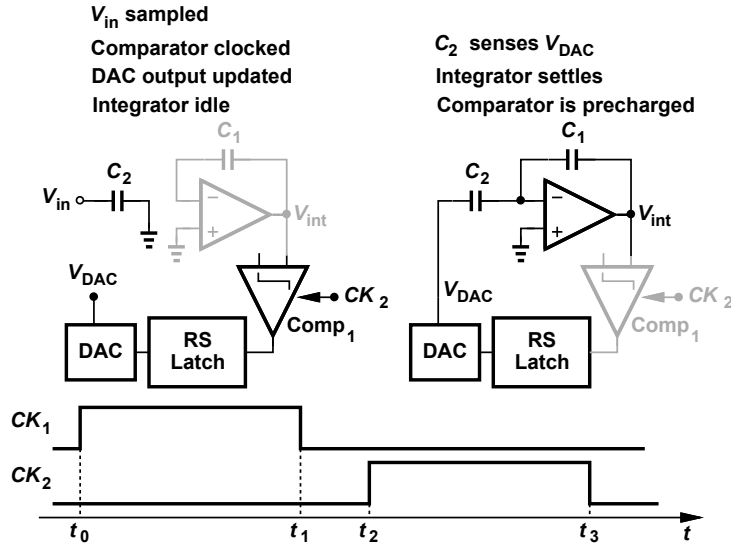


Figure 22.9 Summary of $\Delta\Sigma$ loop timing.

loop is negative? If $V_{int} > 0$, the next cycle must reduce this value. Given that the integrator *adds* $V_{in} - V_{DAC}$ to its previous output, we predict that the next value of this difference should be negative, i.e., V_{DAC} should jump to $+V_{REF}$. Thus, a logical ONE produced by the comparator must translate to $+V_{REF}$ at the DAC output.

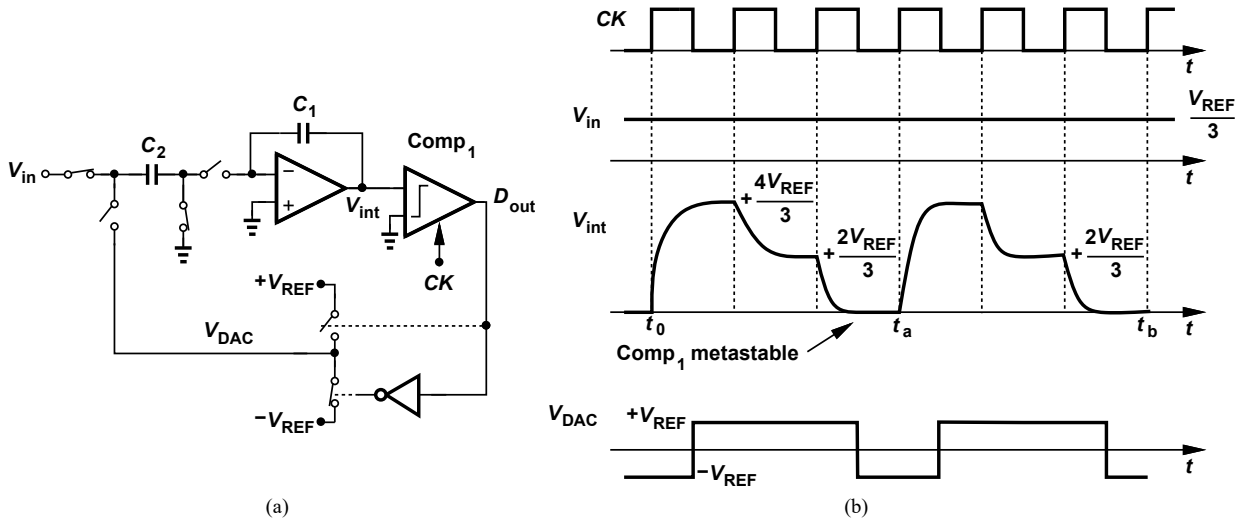


Figure 22.10 (a) First-order loop sensing a constant input, and (b) its waveforms.

We articulate the above point by writing:

$$\text{If } V_{int}[k] > 0 \text{ then } V_{int}[k+1] = V_{int}[k] + V_{in}[k] - V_{REF}, \quad (22.1)$$

$$\text{If } V_{int}[k] < 0 \text{ then } V_{int}[k+1] = V_{int}[k] + V_{in}[k] + V_{REF}, \quad (22.2)$$

where $V_{int}[k]$ and $V_{int}[k+1]$ denote the integrator output in the present and the next clock cycles, respectively.

We begin our analysis of Fig. 22.10(a) with a constant input, e.g., $V_{in} = +V_{REF}/3$. We assume the loop is driven by a single clock, CK , such that the integration mode begins on its rising edge and $Comp_1$ is activated

on its falling edge. (Recall from Fig. 22.9 that the comparator decision reaches the integrator one-half of a clock cycle later.) Equations (22.1) and (22.2) prescribe the integrator output trajectory as

$$\text{If } V_{int}[k] > 0 \text{ then } V_{int}[k+1] = V_{int}[k] - \frac{2}{3}V_{REF}, \quad (22.3)$$

$$\text{If } V_{int}[k] < 0 \text{ then } V_{int}[k+1] = V_{int}[k] + \frac{4}{3}V_{REF}. \quad (22.4)$$

We observe that the $\Delta\Sigma$ loop “pumps” the integrator *down* by $2V_{REF}/3$ if Comp_1 generates a ONE, and *up* by $4V_{REF}/3$ if it delivers a ZERO. This asymmetry is necessary so that V_{DAC} provides an average equal to $V_{in} = +V_{REF}/3$, as seen below.

Assuming C_1 carries no charge and Comp_1 holds a ZERO in its RS latch, we note that the loop first pumps up and V_{int} travels from zero to $+4V_{REF}/3$ [Fig. 22.10(b)]. In response to this “overshoot,” the loop pumps V_{int} down by $-2V_{REF}/3$ twice, yielding $V_{int} = 0$ again. Interestingly, the comparator now becomes metastable, generating a high or low output with equal probabilities. Suppose the former occurs and forces V_{int} to drop again by $2V_{REF}/3$. According to (22.4), V_{int} must then rise by $4V_{REF}/3$, taking another two clock cycles to return to zero. Comp_1 is now metastable and assumed to generate a low level at $t = t_b$. Figure 22.10(b) also plots V_{DAC} , revealing that, from $t = t_0$ to $t = t_b$, it is equal to $+V_{REF}$ for four clock cycles and equal to $-V_{REF}$ for two. Its average is thus equal to $V_{REF}/3$. The reader is encouraged to repeat the analysis from t_a if the metastable comparator generates a low output.

The waveforms in Fig. 22.10(b) offer several useful insights. First, the comparator inevitably becomes metastable in every third clock cycle. Second, there are only two repetitive patterns, each lasting three clock cycles. We recognize them as one between $t = 0$ and $t = t_a$ and another between $t = t_a$ and $t = t_b$. Since comparator metastability produces ONES and ZEROS with equal probabilities, the two patterns also repeat at equal rates. Third, the average value of V_{DAC} is equal to $V_{in} = V_{REF}/3$, as expected of the negative-feedback loop. Fourth, the quantizer output waveform is similar to that of V_{DAC} and hence exhibits an average proportional to V_{in} . Fifth, depending on the input waveform, the integrator output may exceed the nominal ADC input range of $\pm V_{REF}$. For example, the op amp must accommodate an output swing of $+4V_{REF}/3$ after $t = 0$, a serious design challenge.

Example 22.4

Repeat the foregoing analysis for $V_{in} = +V_{REF}$ and determine the largest output swing required of the op amp.

Solution

From (22.1) and (22.2), we have

$$\text{If } V_{int}[k] > 0 \text{ then } V_{int}[k+1] = V_{int}[k], \quad (22.5)$$

$$\text{If } V_{int}[k] < 0 \text{ then } V_{int}[k+1] = V_{int}[k] + 2V_{REF}. \quad (22.6)$$

The $\Delta\Sigma$ loop thus pumps the integrator output up by $2V_{REF}$ if $V_{int} < 0$ and keeps it *constant* if $V_{int} > 0$. As in the previous example, we begin with $V_{int} = 0$ and a low level at the comparator output. According to (22.6), V_{int} rises to $+2V_{REF}$ and then, according to (22.5), remains constant indefinitely! Depicted in Fig. 22.11, this is necessary so that V_{DAC} assumes an average of $V_{in} = +V_{REF}$. The op amp must accommodate an output range as wide as $\pm 2V_{REF}$.

In the next stage of our studies, we examine the first-order loop’s response to $V_{in} = V_{REF} \sin \omega_{int} t$, again assuming that the integrator in Fig. 22.10(a) begins with a zero output and Comp_1 holds a low level. Recall that the loop adds $V_{in} + V_{REF}$ or $V_{in} - V_{REF}$ to V_{int} in each cycle. As shown in Fig. 22.12, since $V_{in} = 0$ at $t = 0$, V_{int} climbs to $V_{in} + V_{REF} = V_{REF}$ in the first cycle. Next, the loop adds $V_a - V_{REF}$ to V_{int} , yielding $V_{int} = V_a$. Since V_{int} is still positive, the $\Delta\Sigma$ modulator next adds $V_{in} - V_{REF} = V_b - V_{REF}$ to it. Suppose

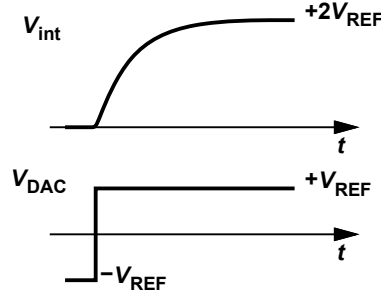


Figure 22.11 Integrator and DAC waveforms for $V_{in} = V_{REF}$.

the net result, $V_{int} = V_a + V_b - V_{REF}$, is negative, causing the loop to add $V_{in} + V_{REF} = V_c + V_{REF}$ to V_{int} in the next cycle. The integrator now produces $V_a + V_b + V_c$. Following these steps, we observe that V_{int} changes to $V_a + V_b + V_c + V_d - V_{REF}$ and, if this value is positive, then to $V_a + V_b + V_c + V_d + V_e - 2V_{REF}$. In other words, the integrator continues to accumulate the input samples, but also receives $+V_{REF}$ and $-V_{REF}$ so as to avoid indefinite growth in one direction.

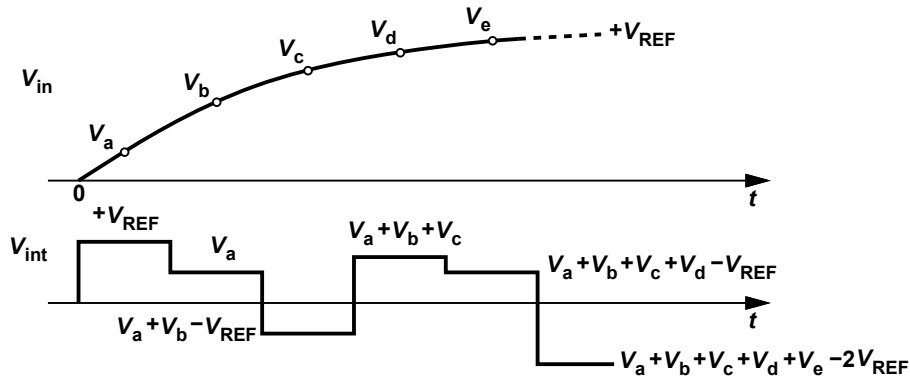


Figure 22.12 Integrator output waveform for an input sinusoid.

Reduction of Integrator Output Swing The waveform examples depicted in Figs. 22.10(b) and 22.11 reveal that the integrator output can reach values well beyond the nominal input range of $\pm V_{REF}$. The design of the op amp thus faces severe difficulties.

It is possible to reduce the swings by scaling the integrator's capacitors. Returning to Fig. 22.10(a) and noting that the loop adds $(C_2/C_1)(V_{in} - V_{REF})$ or $(C_2/C_1)(V_{in} + V_{REF})$, we recognize that the accumulation becomes less steep if $C_2/C_1 < 1$. This “capacitor scaling” method is therefore commonly used in $\Delta\Sigma$ modulator design. The procedure for scaling relies on simulations and proceeds as follows. First, we characterize the integrator output swing statistics with $C_1 = C_2$ and an ideal op amp (a voltage-controlled voltage source having a high gain). In this analysis, we apply a relatively slow sinusoid to the $\Delta\Sigma$ loop and construct the distribution of the integrator output voltage [Fig. 22.13(a)]. Such an endeavor predicts the maximum swing, V_{max} , required of the op amp. Now, to reduce V_{max} to a desired value, αV_{max} , we scale C_2/C_1 by a factor of α , i.e., $C_2/C_1 = \alpha$, and verify the resulting distribution [Fig. 22.13(b)]. The choice of the capacitor values themselves is governed by kT/C noise and described in Chapter 23.

In practice, a $\Delta\Sigma$ modulator receives a random signal having a certain bandwidth f_0 . To emulate this situation, we can create a random waveform from thermal noise, e.g., by a resistor, limit its bandwidth by a low-pass filter, apply it to the $\Delta\Sigma$ loop (Fig. 22.14), and construct the voltage distribution again.³ This test offers a more realistic view of the integrator output swings.

³Such a study requires “transient noise” simulations, e.g., in Cadence.

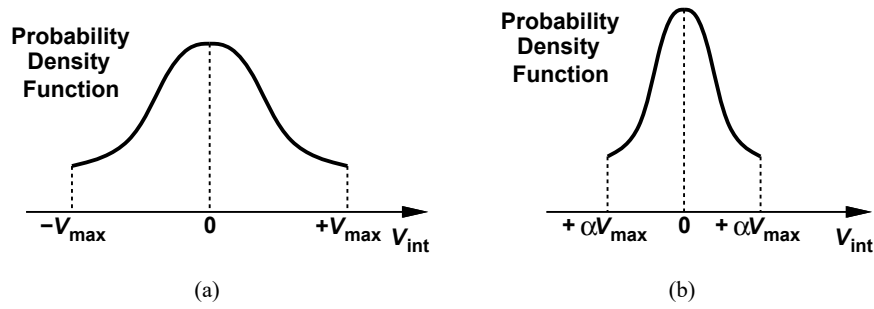


Figure 22.13 Integrator output voltage distribution (a) without scaling, and (b) with scaling.

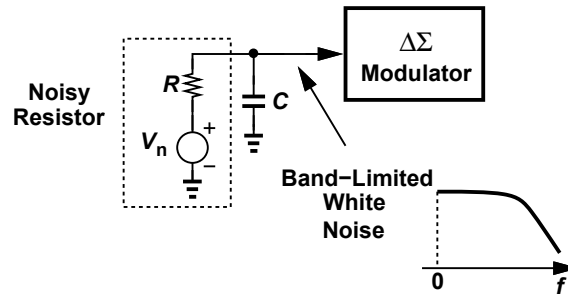


Figure 22.14 Approximation of a random signal by band-limited white noise.

SNR Calculation Our analysis of the first-order loop in Chapter 21 provided an in-band Q-noise power of

$$P_Q = \frac{\pi^2 \Delta^2}{36M^3}, \quad (22.7)$$

where Δ denotes the quantizer's analog LSB and M the oversampling ratio. As explained in Section 21.9, we select Δ equal to the total output swing of the 1-bit quantizer. Since the design shown in Fig. 22.10(a) employs a DAC whose output toggles between $-V_{REF}$ and $+V_{REF}$, the Q-noise returning to the input subtractor exhibits the same swing, i.e., $\Delta = 2V_{REF}$. Equation (22.7) thus yields

$$P_Q = \frac{\pi^2 V_{REF}^2}{9M^3}. \quad (22.8)$$

If the $\Delta\Sigma$ modulator senses an input sinusoid, $V_{REF} \sin \omega_{int} t$, the output SNR emerges as

$$\text{SNR} = \frac{V_{REF}^2/2}{P_Q} \quad (22.9)$$

$$= \frac{9M^3}{2\pi^2}. \quad (22.10)$$

Note the underlying assumptions here: the input is allowed to swing from $-V_{REF}$ to $+V_{REF}$, and the integrator output remains within an acceptable range by proper capacitor scaling.

Example 22.5

A video signal has a bandwidth of 5 MHz. Determine the necessary sampling frequency if the signal must be digitized with a resolution of 14 bits. Assume a first-order modulator.

Solution

From Chapter 12, we have $\text{SNR} = 6.02N + 1.74 \text{ dB} = 86 \text{ dB} \equiv 3.98 \times 10^8$ if $N = 14$. It follows from (22.10) that $M = 956$ and hence $f_{CK} = (2 \times 5 \text{ MHz}) \times M = 9.56 \text{ GHz}$. It is difficult to design such a loop as it necessitates very fast integrator settling. This issue motivates us to consider a second-order loop instead.

22.2 Second-Order DT Loop with 1-Bit Quantizer

The developments in the previous sections can be extended to second-order modulators using a 1-bit quantizer. Invoking the architecture described in Chapter 21, we construct the fully-differential circuit shown in Fig. 22.15(a). Two discrete-time integrators, a comparator, and a 1-bit DAC form the system. We study various aspects of this modulator as we did for the first-order loop in Section 22.1.

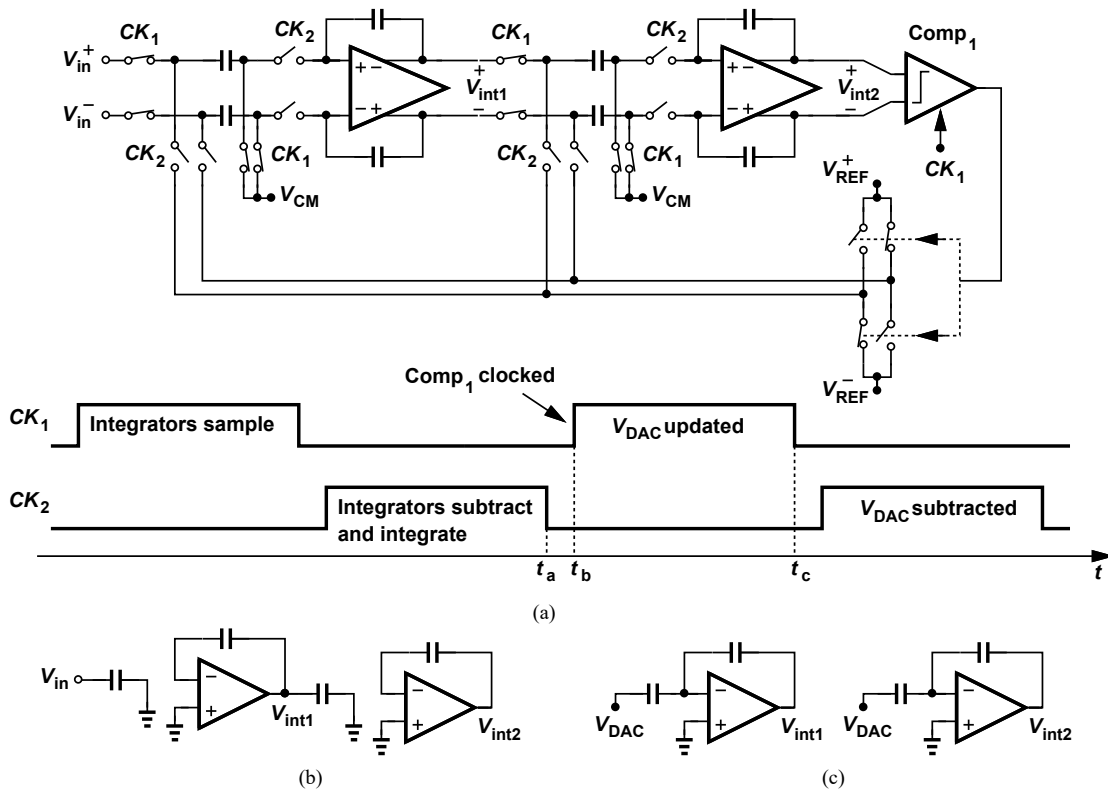


Figure 22.15 (a) Second-order modulator, (b) its equivalent when the integrators sample, and (c) its equivalent when the integrators integrate.

Let us first deal with clocking, bearing in mind that the integrators prefer nonoverlapping phases. Also, Comp₁ includes an RS latch that holds its decision. When CK_1 is high, the integrators sample $V_{in} = V_{in}^+ - V_{in}^-$ and V_{int1} , respectively, as conceptually illustrated in single-ended form in Fig. 22.15(b). Next, CK_1 falls and CK_2 rises, allowing Int₁ and Int₂ to perform subtraction and integration [Fig. 22.15(c)]. That is, the high level of CK_2 must be long enough for V_{int1} and V_{int2} to settle.

We must now decide when the comparator should be clocked. This can occur after V_{int2} has settled, namely, at t_a or t_b in Fig. 22.15(a). Since the switching actions at t_a may briefly perturb V_{int2} , we prefer to clock Comp₁ on the rising edge of CK_1 . Thus, from t_b to t_c , Comp₁ makes a decision and the DAC updates its outputs.

Example 22.6

How do we ensure negative feedback in Fig. 22.15(a)?

Solution

As a test, we set $V_{in}^+ - V_{in}^-$ to zero and V_{int1} and V_{int2} to a positive value, e.g., V_{REF}^+ . We expect the modulator to pump these outputs *down*. That is, a logical ONE produced by Comp_1 must translate to $V_{DAC} = V_{REF}^+ - V_{REF}^-$, as was the case for the first-order loop.

Capacitor Scaling As explained in Chapter 21 and depicted in Fig. 21.41(b), the second-order loop requires gain factors equal to 2 and $1/2$. Our task is to determine how these factors should be implemented. For the factor of $1/2$, we can select the sampling capacitors of the first integrator in Fig. 22.15(a) be half of its feedback devices. The factor of 2 can be created in a similar manner in the second integrator. Recall from Section 22.1.2, however, that the outputs of the integrators experience voltage accumulation, potentially reaching large values. This issue is resolved by proper scaling of the capacitors. We apply a full-scale sinusoidal or random input, run transient simulations, and plot the distributions (histograms) for the integrator outputs. The capacitor ratios are then so chosen as to ensure the output swings are agreeable to the op amps.

Figure 22.16(a) depicts a possible scaling scenario [1]: the second integrator's gain is set equal to 0.5 rather than 2. Figure 22.16(b) presents the voltage swing histograms before and after this scaling [1], revealing two points. First, the first integrator output range shrinks from $\pm V_{REF}$ to about $\pm 0.7V_{REF}$. Second, more importantly, the second integrator output range is reduced from $\pm 2V_{REF}$ to about $\pm 0.6V_{REF}$.

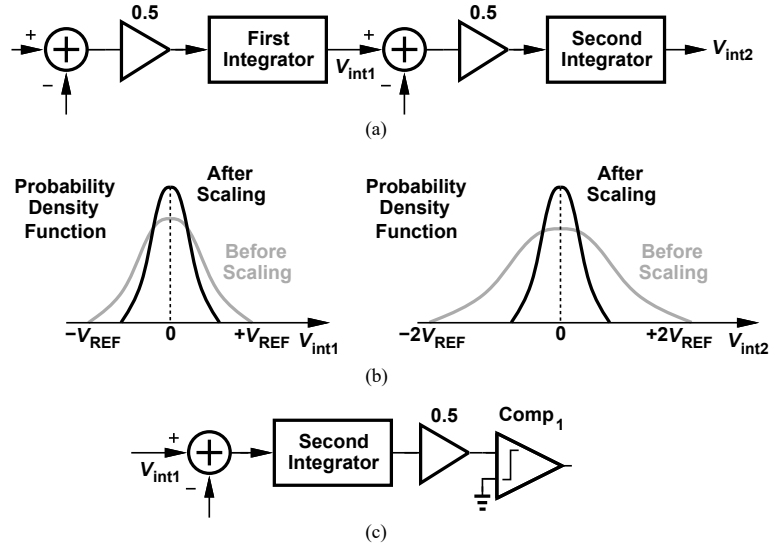


Figure 22.16 (a) Second-order modulator with swing scaling, (b) effect of scaling on output voltage distributions, and (c) integration and scaling swapped.

The reader may notice an inconsistency here: do signal and noise transfer functions not change when the second integrator scaling factor drops from 2 to 0.5? Indeed, a linear model implies that a denominator equal to $0.75z^{-2} - 1.5z^{-1} + 1$ appears in both transfer functions, as the reader can prove. However, the 1-bit quantizer defies linear modeling in this case! To understand this point, let us swap the gain block and the integrator [Fig. 22.16(c)]. We can now *merge* the gain block with the comparator because an ideal 1-bit quantizer provides an infinite gain and is therefore fairly insensitive to signal amplification or attenuation preceding it.

Example 22.7

Following our foregoing observation, a student reasons that the gain of the second integrator has no effect on the transfer functions. Is this true?

Solution

No, not quite. Recall from Chapter 21 that the equivalent gain of a 1-bit quantizer drops for larger input swings. We surmise that a higher integrator gain is partially compensated by a lower quantizer gain. Of course, a lower gain also exacerbates the noise of the comparator. A detailed treatment of this point is beyond our scope. In practice, both integrator gains can be chosen equal to 0.5 [1].

22.2.1 SNR Calculation

As seen in Chapter 21, the in-band Q-noise power of a second-order system is given by

$$P_Q = \frac{\pi^4 \Delta^2}{60M^5}. \quad (22.11)$$

Repeating the analysis of the previous section, we write $\Delta = 2V_{REF}$ and

$$P_Q = \frac{\pi^4 V_{REF}^2}{15M^5}. \quad (22.12)$$

For an input of the form $V_{REF} \sin \omega_{in} t$, we obtain the SNR as

$$\text{SNR} = \frac{15M^5}{2\pi^4}. \quad (22.13)$$

Figure 22.17 plots the SNR of first- and second-order modulators, revealing the significant advantage of the latter.

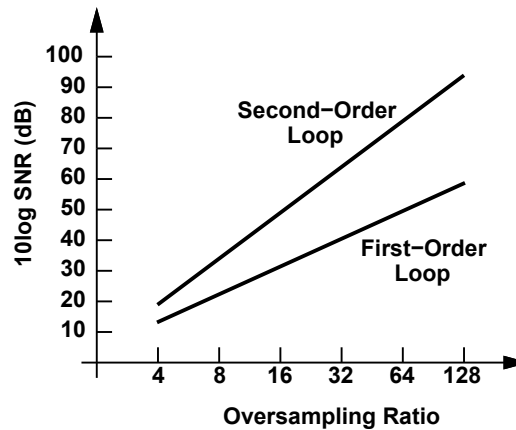


Figure 22.17 SNR vs. oversampling ratio for first- and second-order loops.

Example 22.8

Repeat Example 22.5 for a second-order loop.

Solution

We seek $\text{SNR} = 86 \text{ dB} = 3.98 \times 10^8$. Equation (22.13) gives $M \approx 88$ and hence

$$f_{CK} = (2 \times 5 \text{ MHz}) \times 88 = 880 \text{ MHz}. \quad (22.14)$$

The sampling rate is much more practical now.

22.2.2 Additional Insights

Our initial study of second-order modulators in Chapter 21 began with the simple continuous-time realization depicted in Fig. 22.18(a). We obtained

$$Y(s) = \frac{1}{s^2 + s + 1} X(s) + \frac{s^2}{s^2 + s + 1} Q(s). \quad (22.15)$$

This result suggests that the loop is stable because $s^2 + s + 1$ does not contain any roots on the $j\omega$ axis or

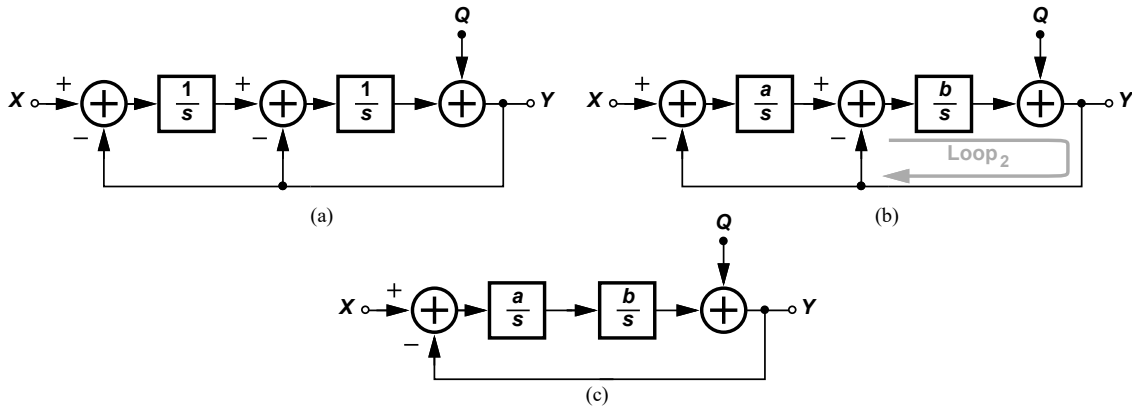


Figure 22.18 (a) Loop with simple integrator models, (b) inclusion of integrator gains, and (c) omission of feedback to the second integrator.

in the right half plane (why?). Nevertheless, we have seen that the integrators may employ certain scaling factors, e.g., $1/2$ or 2 . Does the CT model remain stable if the first and second integrators are expressed as a/s and b/s , respectively [Fig. 22.18(b)]? We now have

$$\left[(X - Y) \frac{a}{s} - Y \right] \frac{b}{s} + Q = Y \quad (22.16)$$

and hence

$$Y(s) = \frac{ab}{s^2 + bs + ab} X(s) + \frac{s^2}{s^2 + bs + ab} Q(s). \quad (22.17)$$

The poles are equal to $(-b \pm \sqrt{b^2 - 4ab})/2$, and remain in the left half plane if $b > 0$, i.e., if Loop₂ contains only one net inversion.

An interesting question that arises here is, can we omit the second subtractor in Fig. 22.18(a) or (b). Shown in Fig. 22.18(c), such an arrangement yields

$$Y(s) = \frac{ab}{s^2 + ab} X(s) + \frac{s^2}{s^2 + ab} Q(s). \quad (22.18)$$

The circuit still provides second-order noise shaping. But, if $ab > 0$ (so that the feedback is negative), then $s^2 + ab$ has two imaginary poles, and the loop is unstable. This is because the integrators act “sluggishly,” creating excessive phase shift around the loop and hence instability. In Fig. 22.18(b), on the other hand, Loop₂ partially bypasses the first integrator, shortening the delay and stabilizing the system.

22.3 First-Order Continuous-Time Loop Design

In Chapter 21, we briefly introduced a first-order CT modulator at a high level. The loop is shown in Fig. 22.19(a). We must first decide how to implement the DAC and the subtractor. A natural choice here is to employ a current-steering DAC that directly attaches to the integrator’s virtual ground. Illustrated in Fig. 22.19(b) for a 1-bit quantizer, such a topology allows the comparator to perform the necessary sampling operation. (Comp₁ contains an RS latch that holds its state.) We wish to examine this circuit’s operation and waveforms, as we have for the DT counterpart in Section 22.1.2.

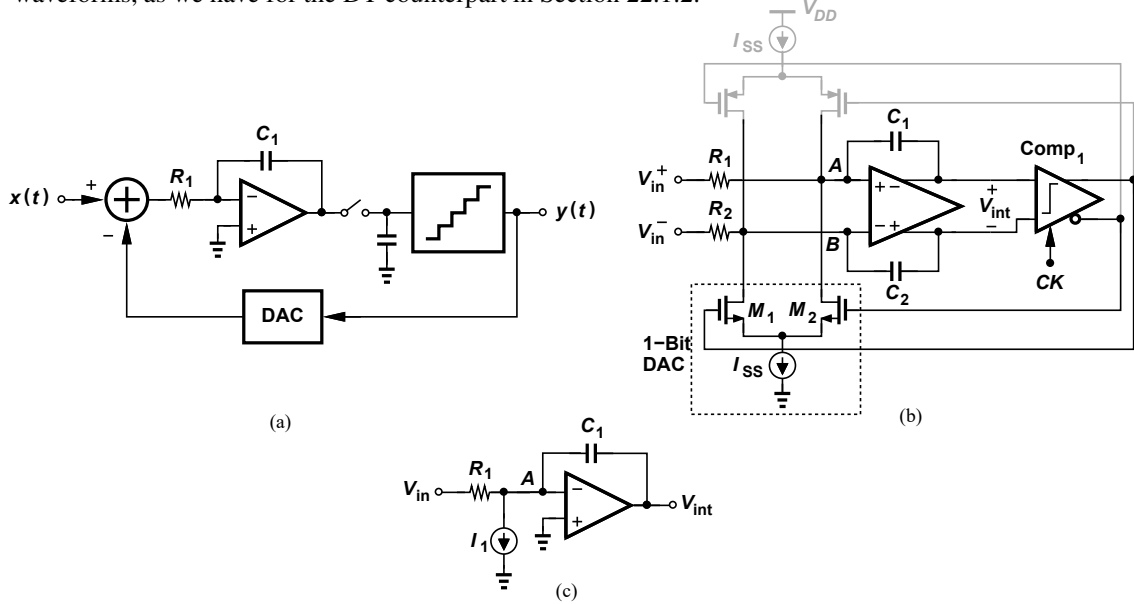


Figure 22.19 (a) First-order continuous-time loop, (b) its differential implementation, and (c) its equivalent circuit.

Since, on average, half of I_{SS} flows through R_1 and R_2 , the common-mode level at A and B is lower than that of the input by an amount equal to $R_1 I_{SS}/2$. If this level shift is undesirable, one can also attach a PMOS differential pair to A and B (shown in gray). The reader can prove that the CM level at A and B is now equal to that of the input.

22.3.1 Basic Operation

How do we ensure negative feedback around the loop in Fig. 22.19(b)? Suppose $V_{in}^+ - V_{in}^- = 0$ but V_{int} begins with a positive value. For the modulator to lower this difference, I_{SS} must be steered to node B (why?). Thus, a positive V_{int} and hence a high level at the comparator output must turn M_1 on and turn M_2 off.

How is this ADC’s input full scale defined? Let us study the single-ended representation in Fig. 22.19(c) and write the current entering node A as $V_{in}/R_1 - I_1$, obtaining

$$V_{int} = \frac{-1}{C_1} \int \left(\frac{V_{in}}{R_1} - I_1 \right) dt \quad (22.19)$$

$$= -\frac{1}{R_1 C_1} \int (V_{in} - I_1 R_1) dt. \quad (22.20)$$

In analogy with the DT modulator of Fig. 22.10(a), we reason that a DAC value of $+V_{REF}$ or $-V_{REF}$ should be subtracted from the input; i.e., $V_{in} - I_1 R_1$ should be interpreted as $V_{in} - V_{REF}$ or $V_{in} + V_{REF}$. Thus, we consider $\pm I_1 R_1 = \pm I_{SS} R_1$ as the input full scale. Each time Comp_1 detects a positive value for V_{int} , the loop adds an equivalent voltage of $+I_{SS} R_1$ to $V_{in}^+ - V_{in}^-$ so that the (inverting) integrator lowers V_{int} . This requires that, in the single-ended circuit of Fig. 22.19(c), I_1 toggle between $+V_{REF}/R_1$ and $-V_{REF}/R_1$.⁴

It is interesting that the architecture of Fig. 22.19(b) does not require nonoverlapping clocks. In comparison to the eight clocked switches in Fig. 22.15(a), this modulator applies the clock only to the comparator.

22.3.2 Loop Waveforms

We now study the response of the first-order CT loop to different types of inputs.

Constant Input Consider the single-ended model shown in Fig. 22.20(a), assuming $I_1 = \pm V_{REF}/R_1$ and

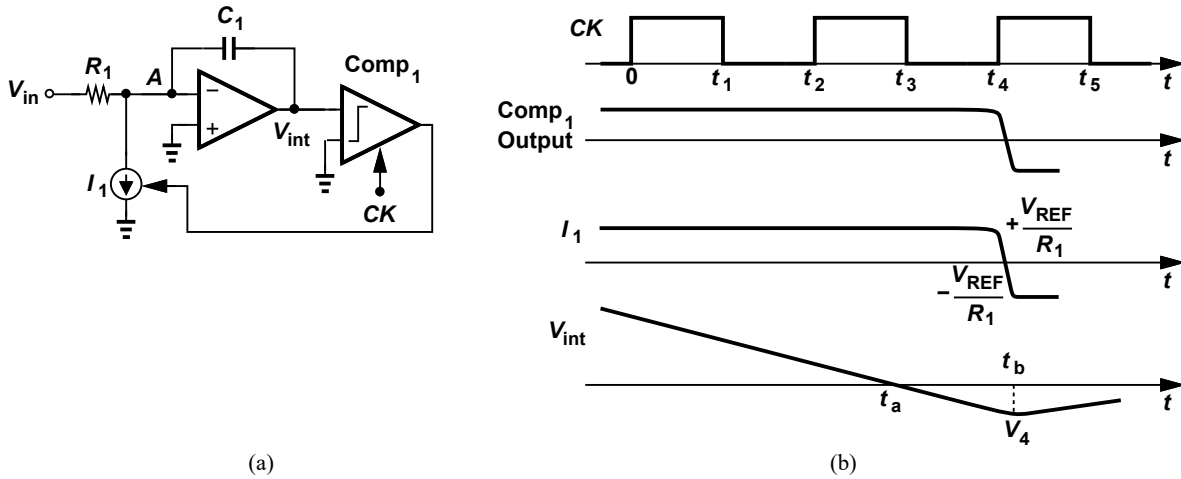


Figure 22.20 (a) Conceptual model of first-order loop, and (b) loop's waveforms.

V_{in} is constant and equal to $+V_{REF}/2$. We observe that V_{int} is a ramp that rises if $V_{in} - I_1 R_1$ in Eq. (22.20) is negative and falls if it is positive. Suppose the loop begins with $V_{int} = +V_{REF}$ while Comp_1 holds a logical ONE [Fig. 22.20(b)]. As explained above, the DAC must pump V_{int} down in this case, i.e., $I_1 = -V_{REF}/R_1$. Thus,

$$V_{int} = +V_{REF} - \frac{1}{R_1 C_1} (V_{in} + V_{REF}) t \quad (22.21)$$

$$= +V_{REF} - \frac{3V_{REF}}{2R_1 C_1} t. \quad (22.22)$$

At $t = t_1$, V_{int} may or may not cross zero, depending on the value of the integrator's time constant, $R_1 C_1$. If it does not, then Comp_1 , which is clocked at $t = t_2$, still delivers a ONE, allowing V_{int} to fall further. This continues until a time such as t_a , at which V_{int} changes sign, informing Comp_1 to produce a ZERO. Comp_1 is clocked at t_4 , after which I_1 changes its direction and V_{int} begins to rise. We then have

$$V_{int} = V_4 - \frac{1}{R_1 C_1} (V_{in} - V_{REF}) (t - t_b) \quad (22.23)$$

$$= V_4 + \frac{V_{REF}}{2R_1 C_1} (t - t_b). \quad (22.24)$$

⁴Since we allow V_{in} to be positive or negative in this circuit, I_1 cannot swing between only zero and I_{SS} because its running average then fails to track the input.

Note that the absolute value of the ramp's slope has decreased by a factor of 3. Then, V_{int} climbs until it becomes positive, causing Comp_1 to generate a ONE in another cycle, etc.

Example 22.9

How much is the average value of I_1 in Fig. 22.20(a) when $V_{in} = V_{REF}/2$?

Solution

We recognize that negative feedback around the loop prohibits the integral in Eq. (22.20) from growing indefinitely. This is possible only if the average value of $V_{in} - I_1 R_1$ (which flows through C_1) is zero and hence that of I_1 is equal to $V_{in}/R_1 = V_{REF}/(2R_1)$. We expect this relation to hold even if V_{in} varies (slowly): the “short-term” (“running”) average of the DAC output tracks V_{in}/R_1 . This property was also noted in DT modulators in Section 22.1.2.

The waveforms shown in Fig. 22.20(b) also help us address another important question: what is the largest voltage swing that the op amp must accommodate? Consider a case where $V_{in} = 0$, requiring that I_1 toggle between $+V_{REF}/R_1$ and $-V_{REF}/R_1$ with a 50% duty cycle (Fig. 22.21). We construct the integrator's

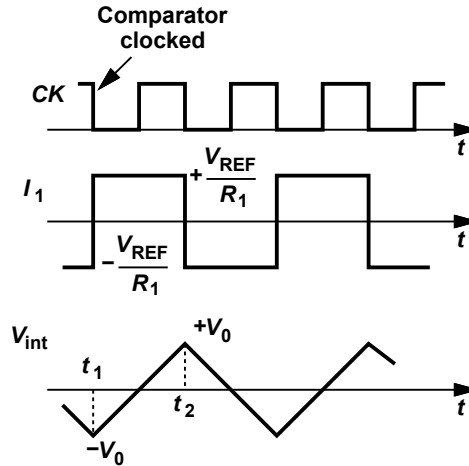


Figure 22.21 Loop waveforms for computing largest integrator swing.

steady-state output waveform while neglecting the comparator delay. Thus, V_{int} changes its direction every time the comparator is clocked and hence travels from $-V_0$ and $+V_0$ or vice versa once every clock cycle. It follows that $I_1(T_{CK})/C_1 = 2V_0$ and hence

$$V_0 = \frac{I_1 T_{CK}}{2C_1} \quad (22.25)$$

$$= \frac{T_{CK} V_{REF}}{2R_1 C_1}. \quad (22.26)$$

The op amp must handle such a swing.

The reader may wonder whether it is possible for V_{int} to take *more* than one clock cycle to swing between $-V_0$ and $+V_0$. Suppose V_{int} takes three cycles (Fig. 22.22). We show that this is not possible. The comparator is clocked at t_1 , causing V_{int} to rise. It is clocked again at t_2 , maintaining its decision. But, at t_4 , Comp_1 requires that V_{int} begin to fall. Thus, the trip from $-V_0$ to $+V_0$ cannot last three clock cycles.

We also face another question: are we certain that the foregoing scenario yields the largest swing at the integrator output? Let us consider another extreme case where V_{in} is slightly less than $+V_{REF}$, causing I_1 to remain at $+V_{REF}/R_1$ most of the time (Fig. 22.23). Writing $V_{in} = V_{REF} - \Delta V$, we recognize that

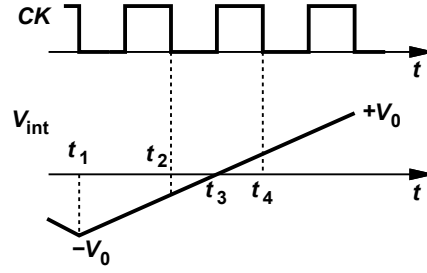


Figure 22.22 Waveforms in a loop taking three clock cycles to drive V_{int} from $-V_0$ to $+V_0$.

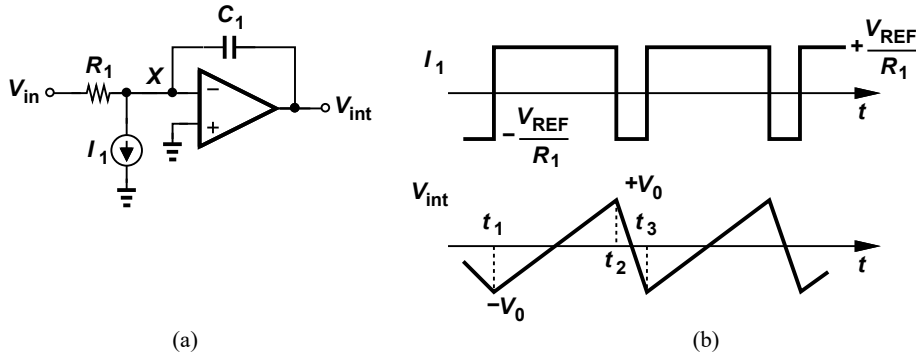


Figure 22.23 Loop waveforms for the case of $V_{in} = V_{REF} - \Delta V$.

the current flowing through C_1 between $t = t_1$ and $t = t_2$ is equal to $(V_{REF} - \Delta V)/R_1 - V_{REF}/R_1$, which amounts to $-\Delta V/R_1$, a small value. From $t = t_2$ to $t = t_3$, on the other hand, the current is given by $(V_{REF} - \Delta V)/R_1 + V_{REF}/R_1 \approx 2V_{REF}/R_1$. For the average DAC current to be equal to $(V_{REF} - \Delta V)/R_1$, we have

$$\frac{V_{REF} - \Delta V}{R_1} T_{CK} = \frac{V_{REF}}{R_1} (t_2 - t_1) - \frac{V_{REF}}{R_1} (t_3 - t_2), \quad (22.27)$$

where the right-hand side represents the total charge drawn by I_1 in one clock cycle. That is,

$$(V_{REF} - \Delta V) T_{CK} = V_{REF} (t_2 - t_1) - V_{REF} (t_3 - t_2). \quad (22.28)$$

But this calculation is not correct! The integrator waveform shown in Fig. 22.23 presents an interesting question: if the comparator is clocked *periodically*, how can $t_2 - t_1$ and $t_3 - t_2$ be unequal? We surmise that these time intervals need not last only one clock cycle. Suppose $t_2 - t_1 = mT_{CK}$ and $t_3 - t_2 = nT_{CK}$, where m and n are integers. Over a total of $m + n$ cycles, the average DAC current, $(m - n)V_{REF}/R_1$, must be equal to V_{in}/R_1 :

$$(m + n)(V_{REF} - \Delta V) = (m - n)V_{REF}. \quad (22.29)$$

That is,

$$\frac{m - n}{m + n} = 1 - \frac{\Delta V}{V_{REF}}. \quad (22.30)$$

The loop therefore finds the smallest values of m and n that satisfy this equation. For example, if $\Delta V = 0.1V_{REF}$, then $(m - n)/(m + n) = 9/10$ and hence $m = 19n$. The smallest m and n are thus equal to

19 and 1, respectively. We observe that the integrator receives a net current of $(V_{REF} - \Delta V)/R_1 - V_{REF}/R_1$ for 19 cycles, generating a peak output equal to $1.9T_{CK}V_{REF}/(2R_1C_1)$ (Fig. 22.24). This swing is greater than that expressed by Eq. (22.26) for $V_{in} = 0$.

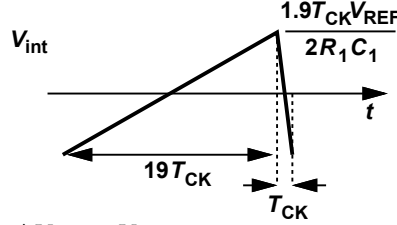


Figure 22.24 Integrator waveform for $\Delta V = 0.1V_{REF}$.

Example 22.10

Generalize the foregoing results for a constant input equal to $V_{REF} - \alpha V_{REF}$.

Solution

We have

$$\frac{m-n}{m+n} = 1 - \alpha \quad (22.31)$$

and hence

$$\frac{m}{n} = \frac{2}{\alpha} - 1. \quad (22.32)$$

For example, $n = 1$ and $m = 2/\alpha - 1$. The peak integrator swing then occurs after $(2/\alpha - 1)T_{CK}$ seconds and is given by $-0.5(2/\alpha - 1)T_{CK}[(V_{REF} - \alpha V_{REF})/R_1 - V_{REF}/R_1]/C_1$:

$$V_{int,p} = (2 - \alpha) \frac{T_{CK}V_{REF}}{2R_1C_1}. \quad (22.33)$$

Sinusoidal Input We have learned from our general analysis of first-order $\Delta\Sigma$ modulators that their DAC output tries to track the input. Thus, with a sinusoidal input and a 1-bit quantizer, the DAC output current toggles between its high and low levels such that its running average is approximately equal to V_{in} divided by the input resistor.

Consider the loop shown in Fig. 22.25(a), noting that C_1 carries two currents, namely, V_{in}/R_1 and $-I_1$. We have

$$V_{int} = -\frac{1}{R_1C_1} \int V_{in} dt + \frac{1}{C_1} \int I_1 dt. \quad (22.34)$$

For $V_{in} = V_{REF} \sin \omega_{in} t$ and $I_1 = \pm V_{REF}/R_1$,

$$V_{int} = \frac{V_{REF}}{R_1C_1\omega_{in}} \cos \omega_{in} t \pm \frac{V_{REF}}{R_1C_1} t, \quad (22.35)$$

where the second term bears a positive sign if the comparator output is low (why?) and vice versa. In other words, the integrator output contains a succession of rising or falling ramps that oppose the integrated input and force V_{int} toward zero. Figure 22.25(b) depicts the simulated integrator waveforms.

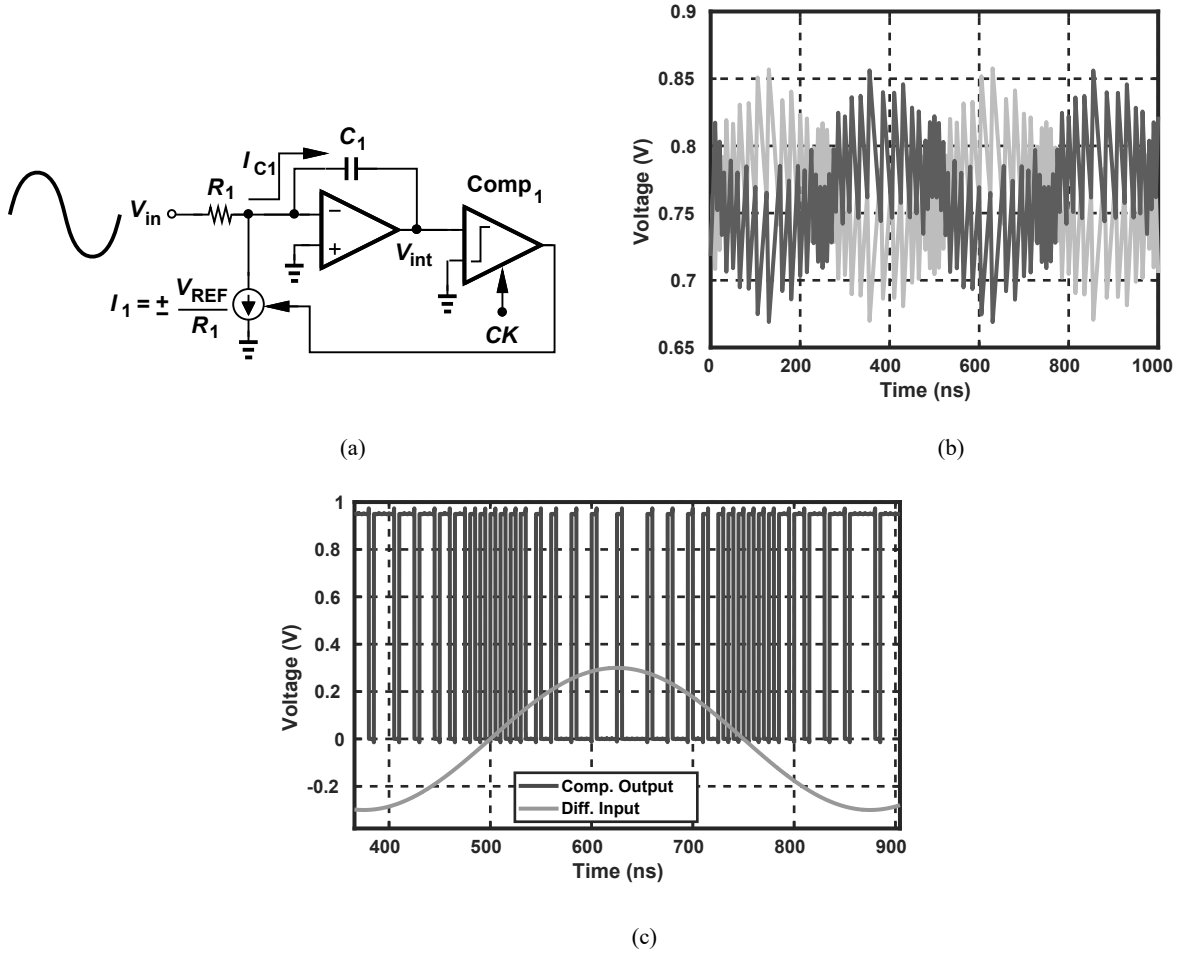


Figure 22.25 (a) First-order loop with sinusoidal input (shown in single-ended form for simplicity), (b) differential integrator outputs, and (c) analog input and comparator output.

The comparator output and the DAC current toggle so as to create a running average equal to V_{in}/R_1 . Figure 22.25(c) illustrates this behavior, revealing nearly equal high and low time intervals when $V_{in} \approx 0$, mostly high levels when $V_{in} \approx +V_{REF}$, and mostly low levels when $V_{in} \approx -V_{REF}$.

A key result offered by Eq. (22.35) is that the integrator output swings are generally greater for smaller values of R_1C_1 . To allow a realistic op amp design, therefore, R_1C_1 must be *large* enough to avoid “overload” and high nonlinearity. As explained in the next section, this point translates to a trade-off between the op-amp voltage swings and the modulator clock frequency.

22.3.3 SNR Calculation

Continuous-time $\Delta\Sigma$ modulators follow the original intuition that we employed in Chapter 21 to arrive at the concept of noise shaping. As such, they lend themselves to a simple CT noise model (Fig. 22.26). We have

$$Y(s) = \frac{1}{R_1C_1s + 1}X(s) + \frac{R_1C_1s}{R_1C_1s + 1}Q(s). \quad (22.36)$$

For low-frequency Q-noise components, the noise transfer function reduces to R_1C_1s , yielding

$$S_Y(f) \approx (2\pi R_1C_1f)^2 S_Q(f). \quad (22.37)$$

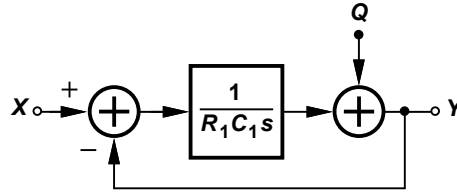


Figure 22.26 Simple CT loop model.

This result is similar to that of the first-order DT counterpart expressed by Eq. (21.33), with two exceptions. First, it implies that Q-noise can be reduced by lowering R_1C_1 , but we know from the previous section that such a remedy leads to greater voltage swings at the output of the integrator(s). Second, this equation reveals that $S_Y(f)$ does not depend on the clock frequency! But why? The following example explains.

Example 22.11

Find a CT approximation of the improved DT integrator described in Section 22.1.1 and explain why its gain is proportional to f_{CK} .

Solution

As shown in Fig. 22.27(a), once every clock cycle, the input capacitor is charged to V_{in} and discharged to the (virtual) ground. We then visualize this action as illustrated in Fig. 22.27(b), recognizing that the network can be approximated by a resistor equal to $R_{eq} = 1/(f_{CK}C_2)$ (Chapter 21). It follows that the CT approximation of the integrator [Fig. 22.27(c)] exhibits a transfer function given by

$$\frac{V_{out}}{V_{in}}(s) = -\frac{1}{R_{eq}C_1s} \quad (22.38)$$

$$= -f_{CK} \frac{C_2}{C_1s}. \quad (22.39)$$

For a DT integrator, the “gain,” $f_{CK}(C_2/C_1)$, is thus proportional to f_{CK} , a point of contrast to that of the CT topology used in Fig. 22.25(a).

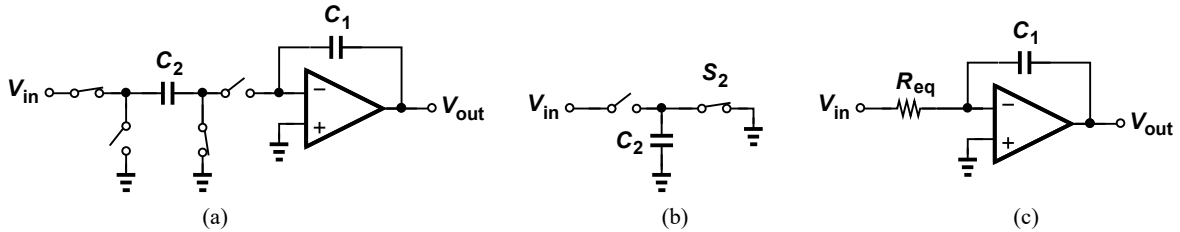


Figure 22.27 (a) Discrete-time integrator, (b) input sampler, and (c) CT equivalent.

The foregoing thoughts imply that noise shaping in a CT modulator does *not* improve as f_{CK} increases—unless R_1C_1 in Eq. (22.37) is reduced proportionally. Indeed, if the modulator must operate with different clock frequencies, C_1 is programmed accordingly.

To compute the SNR, we first integrate (22.37) across the signal bandwidth, f_0 , assuming that $S_Q(f) = \Delta^2/(12f_{CK})$:

$$P_Q = \int_{-f_0}^{+f_0} \frac{\Delta^2}{12f_{CK}} (2\pi R_1C_1f)^2 df \quad (22.40)$$

$$= \frac{2}{9} \pi^2 R_1^2 C_1^2 \frac{f_0^3}{f_{CK}} \Delta^2. \quad (22.41)$$

For a 1-bit quantizer, Δ is given by the DAC's peak-to-peak output swing. We have a swing of $2V_{REF}/R_1$ for I_1 in Fig. 22.25(a). This quantity can be referred to the input as a voltage swing of $2V_{REF}$, which, along with $V_{in} = V_{REF} \cos(2\pi f_0 t)$, yields

$$\text{SNR}_{\text{CT}} = \frac{9f_{CK}}{16\pi^2 R_1^2 C_1^2 f_0^3}. \quad (22.42)$$

Note that $\text{SNR} \propto f_{CK}$ because $S_Q(f) \propto 1/f_{CK}$. By comparison, the SNR of the DT modulator is written from (22.10) as

$$\text{SNR}_{\text{DT}} = \frac{9M^3}{2\pi^2} \quad (22.43)$$

$$= \frac{9}{16\pi^2} \frac{f_{CK}^3}{f_0^3}. \quad (22.44)$$

One can select $R_1 C_1 = 1/f_{CK}$ to obtain the same SNR.

Example 22.12

Repeat Example 22.5 for a first-order CT modulator incorporating a 1-bit quantizer.

Solution

For $\text{SNR} = 86 \text{ dB} \equiv 3.98 \times 10^8$ with $f_0 = 5 \text{ MHz}$, Eq. (22.42) requires $f_{CK}/(R_1^2 C_1^2) = 8.73 \times 10^{29} \text{ Hz}^3$. For example, if $R_1 = 1 \text{ k}\Omega$ and $C_1 = 100 \text{ fF}$, we must have $f_{CK} = 8.73 \text{ GHz}$, a challenging constraint. These values motivate us to consider second-order noise shaping.

This example and Eq. (22.42) suggest that $R_1 C_1$ must be chosen *small* enough to accommodate a reasonable clock frequency. However, as revealed by Eq. (22.35), a small $R_1 C_1$ yields high voltage swings at the integrator output, making the op amp nonlinear.

22.4 Second-Order Continuous-Time Loop Design

We begin with the simple architecture shown in Fig. 22.28(a). We assume an input range of $-V_{REF}$ to $+V_{REF}$ and $|I_1|R_1 = |I_2|R_2 = V_{REF}$. As explained in Section 22.2.2, a single loop containing two ideal integrators is unstable, but the inner loop in this topology partially bypasses the first integrator, stabilizing the circuit.

The DAC currents in Fig. 22.28(a) can be referred to the integrator inputs as voltages [Fig. 22.28(b)]. We can view the comparator output, Y , as a waveform toggling between -1 and $+1$ and write $I_1 R_1 = Y|I_1|R_1 = YV_{REF}$. Alternatively, we can simply assume that Y swings between $-V_{REF}$ and $+V_{REF}$ and construct the equivalent circuit shown in Fig. 22.28(c). This leads to the equivalent modulator illustrated in Fig. 22.28(d), where V_{in} is denoted by X . Beginning from the input, we can now write

$$\left(-\frac{X - Y}{R_1 C_1 s} - Y \right) \left(\frac{-1}{R_2 C_2 s} \right) + Q = Y \quad (22.45)$$

and hence

$$Y(s) = \frac{1}{R_1 C_1 R_2 C_2 s^2 - R_1 C_1 s + 1} X(s) + \frac{R_1 C_1 R_2 C_2 s^2}{R_1 C_1 R_2 C_2 s^2 - R_1 C_1 s + 1} Q(s). \quad (22.46)$$

Unfortunately, the poles of the NTF lie in the right half plane (why?), indicating instability. This occurs because the integrators in Fig. 22.28(a) invert. In differential implementations, we can readily change both to

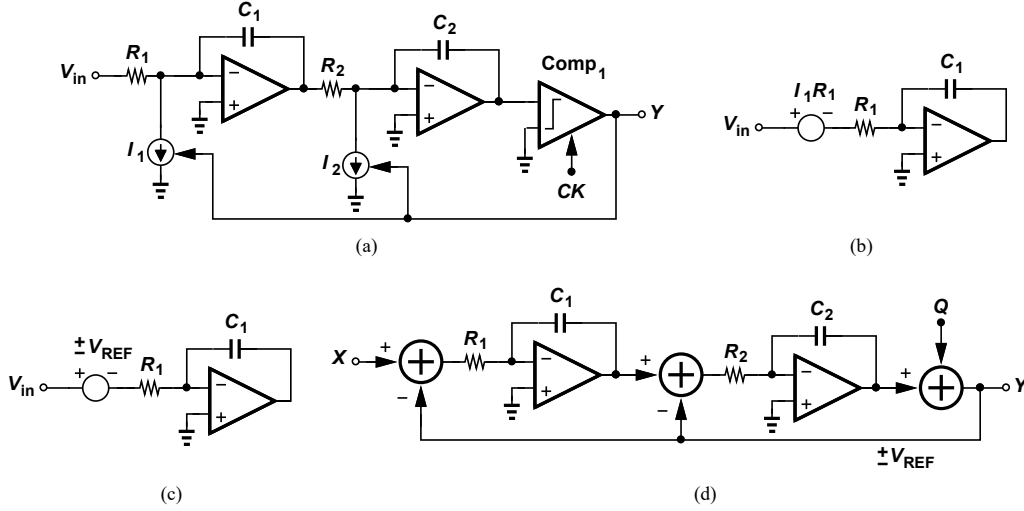


Figure 22.28 (a) Second-order CT loop, (b) first integrator/DAC model, (c) alternate model, and (d) overall loop model.

a noninverting topology, obtaining

$$Y(s) = -\frac{1}{R_1 C_1 R_2 C_2 s^2 - R_1 C_1 s + 1} X(s) + \frac{R_1 C_1 R_2 C_2 s^2}{R_1 C_1 R_2 C_2 s^2 + R_1 C_1 s + 1} Q(s). \quad (22.47)$$

The poles are now real and negative. At low Q-noise frequencies, the second-order NTF emerges as

$$\frac{Y}{Q}(s) \approx R_1 C_1 R_2 C_2 s^2, \quad (22.48)$$

and the output shaped noise as

$$S_Y(f) \approx (4\pi^2 R_1 R_2 C_1 C_2 f^2)^2 S_Q(f). \quad (22.49)$$

22.4.1 SNR Calculation

As with the first-order CT modulator studied in Section 22.3.3, we assume an input signal bandwidth of f_0 and integrate the Q-noise power from Eq. (22.49):

$$P_Q = \int_{-f_0}^{+f_0} \frac{\Delta^2}{12 f_{CK}} (4\pi^2 R_1 C_1 R_2 C_2 f^2)^2 df \quad (22.50)$$

$$= \frac{8\pi^4 \Delta^2}{15} R_1^2 C_1^2 R_2^2 C_2^2 \frac{f_0^5}{f_{CK}}. \quad (22.51)$$

Selecting $R_1 C_1 R_2 C_2 = 1/f_{CK}^2$ yields the same P_Q as that of the DT counterpart [Eq. (22.11)]. We also write $\Delta = 2V_{REF}$ and $V_{in} = V_{REF} \cos(2\pi f_0 t)$ and hence

$$\text{SNR} = \frac{15 f_{CK}}{64\pi^4 R_1^2 C_1^2 R_2^2 C_2^2 f_0^5}. \quad (22.52)$$

If $R_1 C_1 R_2 C_2 = 1/f_{CK}^2$, this expression reduces to that derived for the DT loop [Eq. (22.13)].

Example 22.13

Repeat Example 22.5 for a second-order CT loop.

Solution

For $\text{SNR} = 3.98 \times 10^8$ and $f_0 = 5 \text{ MHz}$, we have $f_{CK}/(R_1^2 C_1^2 R_2^2 C_2^2) = 5.17 \times 10^{44} \text{ Hz}^5$. For example, if $R_1 C_1 = R_2 C_2 = 1/f_{CK}$, then $R_1 C_1 = 1.14 \text{ ns}$ and $f_{CK} = 876 \text{ MHz}$, both much more feasible values than those encountered in Example 22.12 for the first-order loop.

22.4.2 Integrator Output Swings

Recall from Section 22.2 that the output swings of the integrators in a second-order discrete-time $\Delta\Sigma$ modulator can become rather large, requiring proper gains and hence capacitor ratios. The same issue arises in CT counterparts, except that their gain is controlled by their time constant. This effect is particularly acute at the second integrator output, where signals appear after two integrations and hence tend to generate large swings. We must then so select $R_1 C_1$ and $R_2 C_2$ in Fig. 22.28(a) as to obtain acceptable voltage swings at the two integrators' outputs. This often necessitates careful simulations. We must bear in mind that these outputs contain the integrals of both the input signal and the DAC jumps, as exemplified by Eq. (22.35) for first-order loops.

22.5 Generation of Nonoverlapping Clocks

The nonoverlapping clock phases necessary for $\Delta\Sigma$ modulators (and for pipelined ADCs) can be generated by an RS latch. Let us first consider the simple arrangement shown in Fig. 22.29(a). Before $t = t_1$, CK is

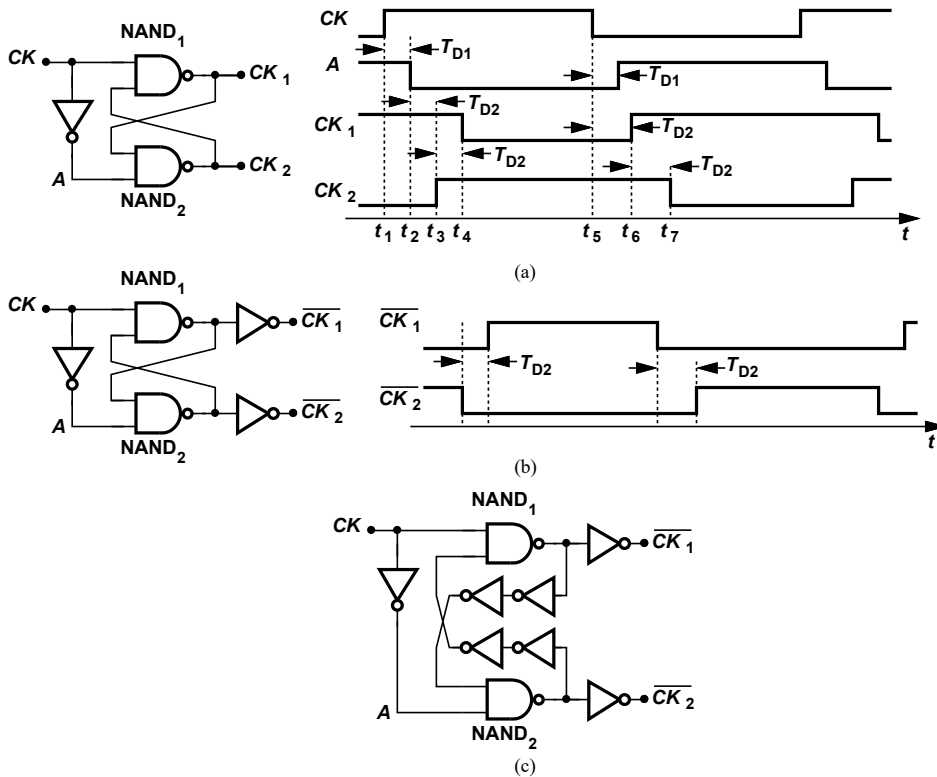


Figure 22.29 (a) Nonoverlapping clock generator, (b) addition of inverters to the outputs, and (c) use of additional feedback delays to increase the nonoverlap time.

low, CK_1 is high and so is A . Thus, CK_2 is low. At $t = t_1$, CK rises and one inverter delay, T_{D1} , later, A falls. Now, NAND₂ senses a low level, generating a high output after T_{D2} seconds (at $t = t_3$). This transition

travels through NAND_1 to produce a low level at $t = t_4$. At $t = t_5$, CK falls, causing A to rise T_{D1} seconds later and also CK_1 to rise T_{D2} seconds later. Upon sensing two high input levels, NAND_1 delivers a low output at $t = t_7$. We observe that the low level of CK_1 is always “contained” within the high level of CK_2 and vice versa. Inverting these outputs as shown in Fig. 22.29(b), we obtain nonoverlapping clocks $\overline{CK_1}$ and $\overline{CK_2}$. The nonoverlapping time is equal to one NAND delay, T_{D2} .

This nonoverlap time proves excessively short as it is comparable to the rise and fall times of the clocks. For this reason, we deliberately increase the NAND delay, by inserting inverters in the feedback paths [Fig. 22.29(c)]. The reader can show that the nonoverlap time is now equal to $T_{D2} + 2T_{D1}$.

References

- [1]. B. Boser and B. A. Wooley, “The design of sigma-delta modulation analog-to-digital converters,” *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 1298-1308, Dec. 1988.

Problems

- 22.1. Assuming $V_{in} = V_0 \cos \omega_1 t$ in Fig. 22.1(a), sketch V_p and V_{out} versus time if $2T/\omega_1 = 5T_{CK}$.
- 22.2. In Fig. 22.7(b), the DAC switches exhibit an on-resistance of R_{on} . If the op amp and switches S_{2a} and S_{2b} are ideal, determine the settling time constant of the integrator.
- 22.3. Can the comparator in Fig. 22.7(b) be clocked while S_{2a} and S_{2b} are on? Explain.
- 22.4. Repeat the analysis depicted in Fig. 22.10 if $V_{in} = V_{REF}/5$.
- 22.5. Repeat the analysis depicted in Fig. 22.10 if $C_2 = 2C_1$.
- 22.6. Repeat the analysis depicted in Fig. 22.12 if $C_2 = 0.5C_1$ in Fig. 22.10(a).
- 22.7. Suppose the op amps in Fig. 22.15(a) can be modeled by open-loop response of the form $A_0/(1 + s/\omega_0)$. Neglect their output resistance and assume that all of the capacitors in Fig. 22.15(a) are equal. Determine whether the first integrator suffers from longer time constant in Fig. 22.15(b) or in Fig. 22.15(c).
- 22.8. A student decides to omit the feedback path to the first integrator in Fig. 22.18(a). Determine Y/X and Y/Q .
- 22.9. Suppose $V_{in}^+ = V_{in}^-$ in Fig. 22.19(b). If V_{int} begins with a value equal to $+I_{SS}R_1$, sketch V_{int} and the comparator output versus time. You may need to consider different scenarios depending on how fast the DAC discharges C_1 and C_2 and how long the clock period is.
- 22.10. If $V_{in} = V_0 \cos(2\pi f_{in} t)$ in Fig. 22.20(a), sketch V_{int} and the comparator output versus time. Assume $V_{int}(t = 0) > 0$ and the comparator is clocked for the first time at $t = 0$. Also, assume $f_{in} = f_{CK}/8$.
- 22.11. Suppose that the CT modulator of Fig. 22.20(a) must provide an SNR of 74 dB (corresponding to 12 bits) for $f_0 = 100$ MHz. Selecting $R_1 C_1 = 1/(f_{CK})$, determine the required clock frequency. If $C_1 = 2$ pF, what is the value of R_1 ?
- 22.12. Repeat the previous problem for the second-order loop of Fig. 22.28(a), assuming that $R_1 = R_2$, $C_1 = C_2$, and $I_1 = I_2$.
- 22.13. If $V_{in}^+ - V_{in}^-$ in Fig. 22.19(b) is constant and equal to ΔV , sketch V_{int} and the comparator output versus time. You may need to consider different scenarios depending on how fast the DAC discharges C_1 and C_2 and how long the clock period is.
- 22.14. If the common-mode level of V_{in}^+ and V_{in}^- in Fig. 22.19(b) is equal to 0.5 V, how much is the CM level at A and B ? Assume the comparator produces logical ONES and ZEROS with equal probabilities.
- 22.15. In Fig. 22.19(b), must the CM level of the integrator output be equal to that of A and B ? Explain.
- 22.16. Suppose $V_{in} = \alpha t$ in Fig. 22.10(a). Sketch V_{int} and the comparator output versus time.
- 22.17. A student mistakenly implements *positive* feedback in Fig. 22.10(a). That is, when the comparator senses $V_{int} > 0$, the DAC produces $V_{DAC} = -V_{REF}$ rather than $+V_{REF}$. Repeat the analysis in Fig. 22.10(b) for this case.
- 22.18. Repeat the analysis depicted in Fig. 22.20 if $V_{in} = V_{REF}/3$.
- 22.19. Is it possible to realize the nonoverlapping clock generator of Fig. 22.29(a) by means of NOR gates rather than NAND gates? Explain.
- 22.20. A student uses $\overline{CK_1}$ and $\overline{CK_2}$ in Fig. 22.29(b) as feedback signals to the inputs of NAND₂ and NAND₁, respectively. Does the circuit still act as a nonoverlapping clock generator?

Nonidealities in $\Delta\Sigma$ Modulators

Having developed a basic knowledge of $\Delta\Sigma$ modulators in the last two chapters, we naturally wonder how we can push their performance. To this end, we must first quantify their nonidealities.

We begin with discrete-time modulators and analyze the effect of capacitor mismatch, kT/C noise, op-amp limitations, comparator imperfections, reference settling, and clock jitter. Next, we deal with two phenomena, namely, unwanted “tones” and noise folding. Then, we repeat our study for continuous-time loops and delve into some of their specific issues, such as op-amp slewing, comparator delay, and clock jitter. Since both $\Delta\Sigma$ modulators and pipelined ADCs rely on op amps, we will often draw comparisons between the two architectures.

23.1 DT Modulator Nonidealities

23.1.1 Capacitor Mismatch

In contrast to SAR and pipelined ADCs, $\Delta\Sigma$ modulators can tolerate a great deal of capacitor mismatch. Consider the first-order loop shown in Fig. 23.1(a) and recall from Chapter 22 that the integrator transfer

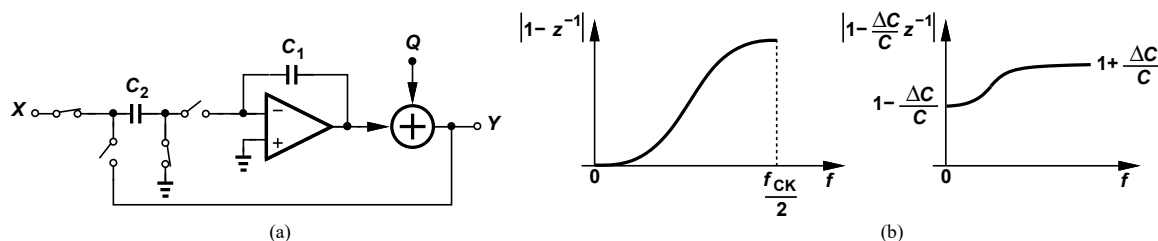


Figure 23.1 (a) DT modulator, and (b) its ideal and mismatch-induced noise-shaping functions.

function is expressed as $(C_2/C_1)z^{-1}/(1 - z^{-1})$. Also, the closed-loop noise and signal transfer functions are respectively given by

$$\frac{Y}{Q}(z) = \frac{1 - z^{-1}}{1 - z^{-1} + \frac{C_2}{C_1}z^{-1}}, \quad (23.1)$$

$$\frac{Y}{X}(z) = \frac{C_2}{C_1} \frac{z^{-1}}{1 - z^{-1} + \frac{C_2}{C_1}z^{-1}}. \quad (23.2)$$

We choose a nominal value of unity for C_2/C_1 so as to obtain $Y/Q = 1 - z^{-1}$. Recall the approximations $1/(1 + \epsilon) \approx 1 - \epsilon$ and $\sqrt{1 - \epsilon} \approx 1 - \epsilon/2$ for $\epsilon \ll 1$. In the presence of capacitor mismatch, we have $C_1 = C$

and $C_2 = C + \Delta C$, and hence

$$\frac{Y}{Q}(z) = \frac{1 - z^{-1}}{1 + \frac{\Delta C}{C} z^{-1}} \quad (23.3)$$

$$\approx (1 - z^{-1}) \left(1 - \frac{\Delta C}{C} z^{-1} \right). \quad (23.4)$$

Replacing z^{-1} with $e^{-j2\pi f T_{CK}}$, we compute the magnitude of the second factor as

$$\left| 1 - \frac{\Delta C}{C} z^{-1} \right| \approx \sqrt{1 - \frac{2\Delta C}{C} \cos(2\pi f T_{CK})} \quad (23.5)$$

$$\approx 1 - \frac{\Delta C}{C} \cos(2\pi f T_{CK}). \quad (23.6)$$

Illustrated in Fig. 23.1(b), this phenomenon simply alters the noise-shaping function by a small amount.

Example 23.1

Explain how the STF is affected by capacitor mismatch.

Solution

From (23.2) and (23.6), we have $|Y/X| \approx |z^{-1}| [1 - (\Delta C/C) \cos(2\pi f T_{CK})]$. That is, the passband response is multiplied by (23.6), experiencing some “ripple.” For example, if $\Delta C/C = 1\%$, then Fig. 23.1(b) suggests that this factor varies from 0.99 at $f = 0$ to 1.0 at $f = f_{CK}/4$.

This analysis reveals that the basic DT modulators studied thus far readily tolerate a few percent of capacitor mismatch, a point of contrast to that necessary in pipelined or SAR ADCs. As explained in Chapter 24, however, modulators using multibit DACs impose more stringent capacitor matching.

23.1.2 kT/C Noise

The capacitors in a DT integrator carry noise from the switches connected to them. In the sampling mode, the circuit can be viewed as shown in Fig. 23.2(a), indicating that C_2 inherits a total noise power of kT/C_2 .

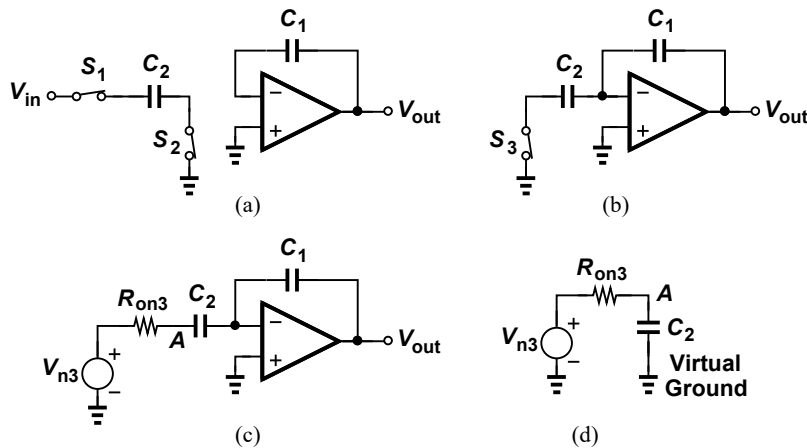


Figure 23.2 (a) DT integrator in sampling mode, (b) the circuit in integration mode, (c) inclusion of noise due to switch resistance, and (d) simplified circuit.

This noise directly corrupts V_{in} .¹ In the integration mode [Fig. 23.2(b)], kT/C_2 is transferred to C_1 , generating an output noise voltage squared equal to $(kT/C_2)(C_2/C_1)^2$ (why?). Moreover, S_3 contributes noise.² To quantify this effect, we draw the equivalent circuit shown in Fig. 23.2(c) and recognize that $|V_{out}/V_A| = C_2/C_1$. Thus, the noise at A due to V_{n3} is simply amplified by this factor. Since the right plate of C_2 is at a virtual ground, we note from Fig. 23.2(d) that the total integrated noise appearing at A is equal to kT/C_2 , yielding a value of $(kT/C_2)(C_2/C_1)^2$ at the integrator output.³ In summary, the integrator inherits two kT/C noise components, each equal to $(kT/C_2)(C_2/C_1)^2$. Dividing this result by the square of the voltage gain yields an input-referred noise power equal to $2kT/C_2$.

In a $\Delta\Sigma$ modulator, however, the effect of kT/C noise is reduced by a factor equal to the oversampling ratio, M . To understand this point, recall that the modulator's oversampled digital output must be eventually converted to the Nyquist rate. We can visualize this operation as shown in Fig. 23.3: every M samples must

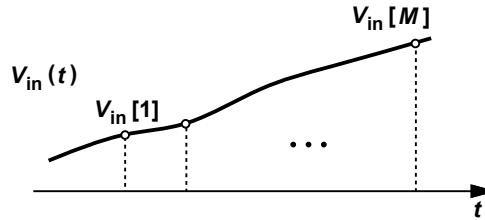


Figure 23.3 Waveform oversampled by a factor of M .

be replaced with one. The simplest method is to compute the average of the M samples

$$D_{out} \equiv \frac{V_{in}[1] + \dots + V_{in}[M]}{M}. \quad (23.7)$$

Since the kT/C noise components corrupting different samples are uncorrelated, their *powers* add. For M samples, the total power is equal to $M(2kT/C_2)$, yielding an rms value of $\sqrt{M(2kT/C_2)}$. This noise is added to the numerator of (23.7). Also, since V_{in} does not change much from one sample to the next, we make a rough approximation and write $V_{in}[1] \approx V_{in}[2] \approx \dots \approx V_{in}[M]$. It follows that

$$D_{out} \equiv V_{in}[1] + \frac{1}{M} \sqrt{\frac{M(2kT)}{C_2}} \quad (23.8)$$

$$\equiv V_{in}[1] + \sqrt{\frac{2kT}{MC_2}}. \quad (23.9)$$

Thus, C_2 is equivalently enlarged by a factor of M . We say the kT/C noise power is reduced by a factor of M .

This reduction can also be proved in the frequency domain. The modulator is followed by a digital filter that removes the quantization noise outside the signal bandwidth, f_0 . If kT/C noise exhibits a flat spectrum up to $f_{CK}/2$, then the filter reduces its power by a factor of $f_{CK}/2/f_0 = M$.

The reduction of kT/C by the oversampling factor proves crucial in high-resolution applications. For example, high-fidelity audio systems require 18-bit converters, in which the LSB is no more than a few tens of microvolts. For $\sqrt{kT/C} \approx 0.2$ LSB, the sampling capacitor value in a Nyquist-rate architecture would reach several hundred picofarads. With oversampling, on the other hand, a few picofarads suffice.

¹In differential implementations, the two input sampling capacitors contribute twice this noise power.

²There is another switch connecting the right plate of C_2 to the virtual ground. The resistance of this switch can be simply merged with R_{on3} .

³We neglect the integrator bandwidth limitation.

Example 23.2

Consider the second-order modulator shown in Fig. 23.4(a). Explain how the kT/C noise associated with the second integrator is referred to the main input.

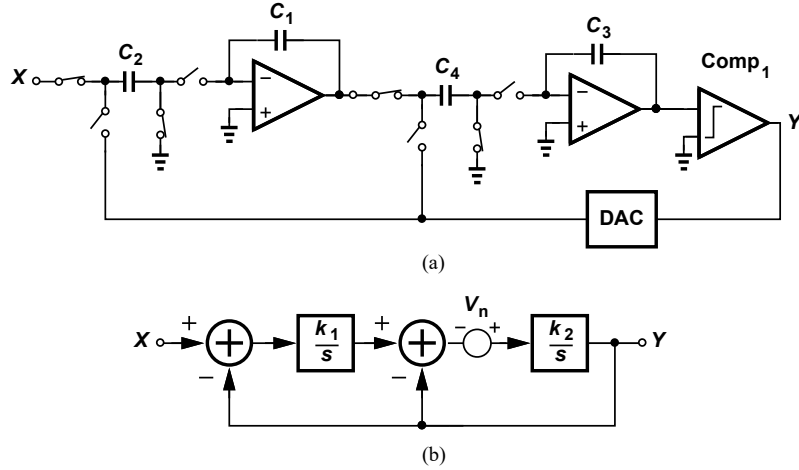


Figure 23.4 (a) Second-order DT modulator, and (b) its equivalent including second integrator's noise.

Solution

Recall from Chapter 21 that we can construct a continuous-time approximation of the system for an intuitive understanding. Depicted in Fig. 23.4(b) is such a model, with V_n denoting the kT/C noise of the second integrator. To refer V_n to the main input, we find the effect of V_n at the output and divide the result by Y/X . Since the first integrator output is equal to $(X - Y)(K_1/s)$, we have

$$\left\{ \left[(X - Y) \frac{K_1}{s} - Y \right] + V_n \right\} \frac{K_2}{s} = Y. \quad (23.10)$$

It follows that

$$Y = \frac{K_1 K_2}{s^2 + K_2 s + K_1 K_2} X + \frac{K_2 s}{s^2 + K_2 s + K_1 K_2} V_n. \quad (23.11)$$

To refer V_n to the input, we must divide its output contribution, i.e., the second term in (23.11), by the transfer function Y/X . Thus,

$$V_{n,in} = V_n \frac{s}{K_1}. \quad (23.12)$$

We say that the input noise of the second integrator is divided by the “gain” of the first. We also say V_n is first-order shaped because it is multiplied by s . For this reason, the kT/C noise of the second integrator is greatly relaxed.

23.1.3 Op-Amp Noise

Shown in Fig. 23.5(a) is the DT integrator including the op-amp noise, V_n . The op amp generates noise in both the sampling mode and the integration mode. But it is only in the latter mode that the comparator senses the output. From Fig. 23.5(b), we recognize that the topology resembles an *amplifier* sensing an input equal

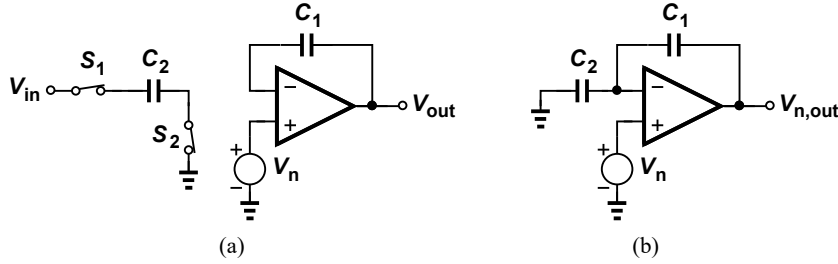


Figure 23.5 Integrator including op-amp noise in (a) sampling mode, and (b) integration mode.

to V_n . In fact, this arrangement is similar to a pipelined (MDAC) stage and its noise behavior was quantified in Chapter 18. Assuming no load capacitance, we proved that the input-referred rms noise is given by

$$V_{n,in,rms}^2 = \overline{V_n^2} \frac{G_m}{C_2}, \quad (23.13)$$

where $\overline{V_n^2}$ denotes the spectrum of the op amp's input-referred noise and G_m its transconductance.

An important property of $\Delta\Sigma$ modulators is that they also reduce the effect of the op-amp noise by the oversampling ratio, M . The proof is similar to that for kT/C noise: the op-amp noise samples corrupting the signal are “averaged out” as the oversampled information is converted to Nyquist-rate data.

Example 23.3

Does the above property allow lower op-amp power consumption?

Solution

Not necessarily. While the op-amp noise is lessened by a factor of M , the circuit must operate *faster* by the same factor. Compared to Nyquist-rate conversion, the op amp now has proportionally less time to settle and must therefore be designed for a wider bandwidth and a greater slew rate. As a result, it suffers from a greater *integrated* noise.

Nevertheless, as explained in Section 23.1.6, the settling precision required of op amps in oversampling ADCs is somewhat more relaxed than that in Nyquist-rate converters, possibly allowing some power savings.

In practice, we select kT/C and op amp noise according to the desired ADC dynamic range. For example, a 14-bit ADC requires that the sum of quantization, kT/C , and op-amp noise be about 86 dB below the full scale. After deciding on the modulator architecture and order, we estimate how fast the circuit can operate and hence assume a value for M . This yields the shaped quantization noise power. We then select the first integrator input capacitor and the op amp design such that their noise contributions (after division by M) are less than or comparable to the Q-noise.

23.1.4 Op-Amp Gain

The finite gain of the op amp in an integrator leads to two effects. First, the nominal gain, e.g., C_2/C_1 in Fig. 23.6 incurs an error. The similarity of the integrator to a pipelined MDAC stage allows us to write the

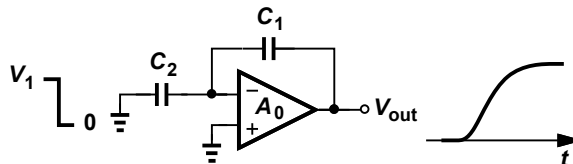


Figure 23.6 Integrator with a finite op-amp gain.

closed-loop gain from Chapter 18 as:

$$\frac{V_{out}}{V_1} = \frac{C_2}{C_1} \frac{1}{1 + \left(1 + \frac{C_2}{C_1}\right) \frac{1}{A_0}} \quad (23.14)$$

$$\approx \left[1 - \left(1 + \frac{C_2}{C_1}\right) \frac{1}{A_0}\right] \frac{C_2}{C_1}. \quad (23.15)$$

The normalized gain error is thus equal to $-(1 + C_2/C_1)/A_0$. Nonetheless, as explained below, $\Delta\Sigma$ modulators can tolerate a fair amount of closed-loop gain error in their integrators. That is, this phenomenon does not typically dictate the open-loop gain of the op amp. Rather, the gain must be high enough to ensure acceptable closed-loop linearity.

Second, since $A_0 < \infty$, the integrator departs from the ideal behavior expressed by $1/s$ or $1/(1 - z^{-1})$. We analyze this effect with the aid of the model shown in Fig. 23.7(a), where, for simplicity, we consider the

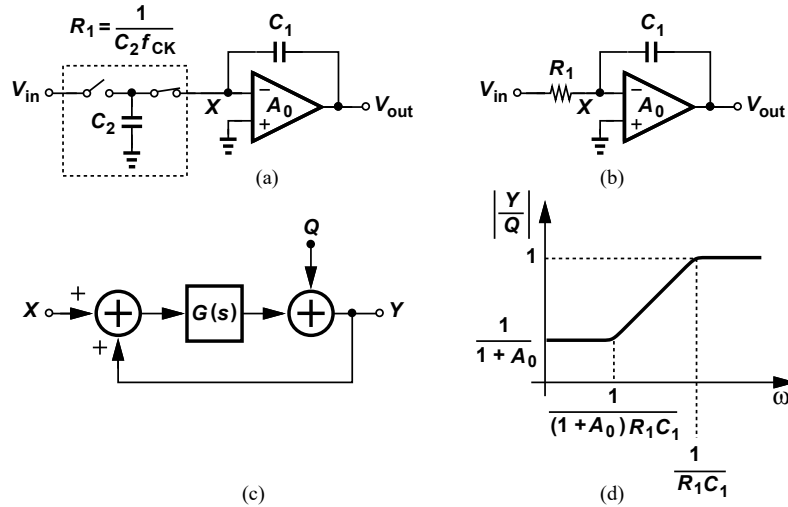


Figure 23.7 (a) DT integrator, (b) its CT equivalent, (c) $\Delta\Sigma$ modulator loop, and (d) corresponding noise transfer function.

original topology described in Chapter 21 and approximate the role of C_2 by a continuous-time resistance equal to $R_1 = 1/(C_2 f_{CK})$. Illustrated in Fig. 23.7(b), the simplified circuit yields $V_X = -V_{out}/A_0$ and hence the current through R_1 as $(V_{in} - V_X)/R_1 = (V_{in} + V_{out}/A_0)/R_1$. This current flows through C_1 , generating a voltage across it equal to $(V_{in} + V_{out}/A_0)/(R_1 C_1 s)$. Subtracting this voltage from V_X , we obtain V_{out} :

$$-\frac{V_{out}}{A_0} - \left(V_{in} + \frac{V_{out}}{A_0}\right) \frac{1}{R_1 C_1 s} = V_{out}. \quad (23.16)$$

It follows that

$$\frac{V_{out}}{V_{in}} = G(s) = \frac{-A_0}{(1 + A_0)R_1 C_1 s + 1}. \quad (23.17)$$

Equation (23.17) reveals two points. (1) The integrator gain at very low input frequencies is less than infinity and equal to $-A_0$. This is to be expected as C_1 approaches an open circuit. (2) The circuit exhibits a pole at

$$\omega_p = \frac{-1}{(1 + A_0)R_1 C_1} \quad (23.18)$$

rather than at the origin.

In the next step, we must determine how the nonideal integrator affects noise shaping in a $\Delta\Sigma$ loop. We know from Chapter 21 that the *infinite* loop gain offered by an ideal integrator suppresses the Q-noise at very low frequencies. The finite gain of $-A_0$ therefore affords less suppression. Turning to Fig. 23.7(c),⁴ we write

$$\frac{Y}{Q}(s) = \frac{1}{1 - G(s)} \quad (23.19)$$

$$= \frac{1}{1 + A_0} \frac{(1 + A_0)R_1C_1s + 1}{R_1C_1s + 1}. \quad (23.20)$$

The noise transfer function contains a zero at $-1/[(1 + A_0)R_1C_1]$ and a pole at $-1/(R_1C_1)$, exhibiting the magnitude response plotted in Fig. 23.6(d). Noise frequencies below the zero experience only a suppression equal to $1/(1 + A_0)$. This effect is called “integrator leakage.”

How large should A_0 be? To answer this question, we must compute the total in-band Q-noise and compare it to that of the ideal case. Let us approximate Y/Q at low frequencies by neglecting the effect of the pole:

$$\left| \frac{Y}{Q}(j2\pi f) \right|^2 \approx R_1^2 C_1^2 (2\pi f)^2 + \frac{1}{(1 + A_0)^2}. \quad (23.21)$$

Multiplying this result by $S_Q(f) = \Delta^2/(12f_{CK})$ yields

$$S_Y(f) = \frac{\Delta^2}{12f_{CK}} \left[4\pi^2 R_1^2 C_1^2 f^2 + \frac{1}{(1 + A_0)^2} \right]. \quad (23.22)$$

Integrating this spectrum across the signal bandwidth, $[-f_0 + f_0]$, and replacing R_1 with $1/(C_2 f_{CK})$, we arrive at the total Q-noise power:

$$P_Q = \frac{\Delta^2}{3} \left[\frac{2\pi^2 C_1^2}{3 C_2^2} \left(\frac{f_0}{f_{CK}} \right)^3 + \frac{1}{2(1 + A_0)^2} \frac{f_0}{f_{CK}} \right]. \quad (23.23)$$

We recognize that the second term within the square brackets arises from integrator leakage and must remain well below the first. The ratio of these two terms can be considered a “penalty.” For example, for a 10% penalty, we have

$$A_0 > \frac{\sqrt{30} C_2}{\pi C_1} M - 1, \quad (23.24)$$

where M is the oversampling ratio. If $C_2/C_1 = 1$, then A_0 must be about $1.7M$.

The foregoing results portray a relaxed design for the op amp. With $M = 64$, for example, one only need select $A_0 \approx 110$, a much more accommodating value than those necessary for high-resolution pipelined ADCs. Nevertheless, as mentioned before, the op-amp *nonlinearity* may require a higher open-loop gain so as to ensure sufficient closed-loop linearity.

23.1.5 Op-Amp Offset

Recall from Chapter 18 that the offset of op amps in a pipelined ADC translates to DNL and INL. Fortunately, the offset proves fairly benign in $\Delta\Sigma$ modulators.

In order to study the effect of the op-amp offset, we proceed in three steps. First, consider the integrator shown in Fig. 23.8(a), where the input, $V_{in} = V_1$, is constant. As the circuit enters the integration mode and the left plate of C_2 swings from V_1 to zero, the topology resembles an amplifier, producing $(-V_1)(-C_2/C_1) = (C_2/C_1)V_1$. In the next clock cycle, C_2 again delivers the same amount of charge, forcing V_{out} to increment by $(C_2/C_1)V_1$. As expected, the circuit integrates V_1 .

⁴Note that Y is *added* to X because $G(s)$ itself inverts.

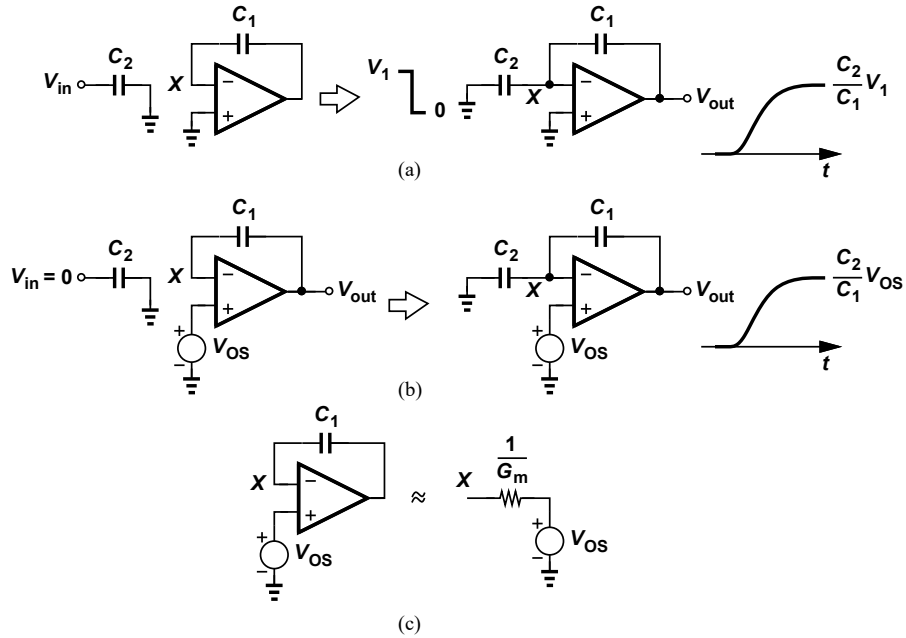


Figure 23.8 (a) Operation of DT integrator, (b) effect of op-amp offset, and (c) simplified model.

Next, we include the op-amp offset, V_{OS} , as depicted in Fig. 23.8(b), where the input is set to zero. Let us assume C_1 has a zero initial condition, recognizing that $V_X = V_{out} \approx V_{OS}$ in the sampling mode. After C_2 is switched to node X , the op amp still retains $V_X \approx V_{OS}$ and C_2 receives charge in the amount of $C_2 V_X \approx C_2 V_{OS}$. This charge must be provided by the op amp and through C_1 . That is, C_1 charges to a voltage equal to $C_2 V_{OS} / C_1$ in the integration mode. In the next clock cycle, a similar sequence occurs, raising V_{out} by $(C_2 / C_1) V_{OS}$ again. The circuit therefore integrates the op amp offset.

From another viewpoint, the op amp, C_1 , and V_{OS} tend to keep V_X approximately equal to V_{OS} (Chapter 18), thus operating as a voltage source [Fig. 23.8(c)].⁵ The periodic switching of C_2 between zero and this voltage creates an average current that charges C_1 .

In the third step, we surmise that the situations illustrated in Figs. 23.8(a) and 23.8(b) are similar, i.e., the output increments due to V_{OS} can be viewed as if they were created by $V_1 = V_{OS}$. Thus, the op-amp offset simply adds to the main input of the integrator (Fig. 23.9). As long as V_{OS} is much less than the modulator's overall dynamic range, it does not degrade the performance.⁶

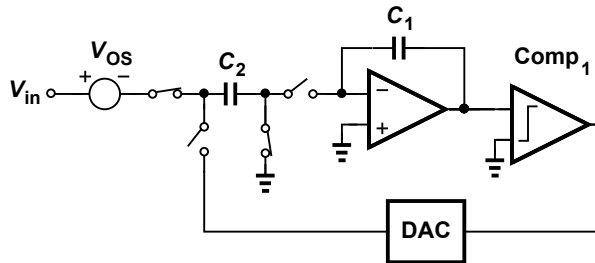


Figure 23.9 Op-amp offset referred to input of a $\Delta\Sigma$ loop.

⁵The impedance seen at X is equal to the inverse of the G_m of the op amp.

⁶Unless the ADC must sense and digitize a dc quantity.

Example 23.4

How does the offset of the second integrator in a second-order $\Delta\Sigma$ modulator manifest itself?

Solution

Recall from Example 23.2 that the noise of the second integrator is divided by the high gain of the first. By the same token, the offset has very little effect.

23.1.6 Op-Amp Speed

We have observed that the first MDAC in a pipelined ADC must settle to full precision while driving the second (Chapter 18). In discrete-time $\Delta\Sigma$ modulators, on the other hand, the op-amp settling speed is more relaxed. To understand this point, we begin with some fundamental concepts.

Linear Settling Consider the RC circuit shown in Fig. 23.10. What happens if we sample the output before

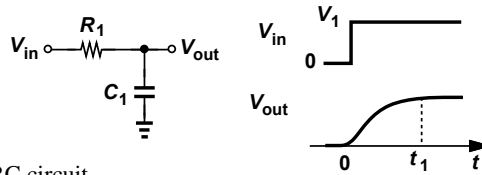


Figure 23.10 Linear settling in an RC circuit.

it settles, e.g., at $t = t_1$? We have

$$V_{out}(t_1) = \left(1 - \exp \frac{-t_1}{\tau}\right) V_1, \quad (23.25)$$

where $\tau = R_1 C_1$. Interestingly, the output incurs only a *gain error* because $V_{out}(t_1) \propto V_1$. We rephrase this result as follows: in a linear system, incomplete settling is equivalent to gain error.

Example 23.5

Determine the gain error in the circuit of Fig. 23.11 due to incomplete settling.

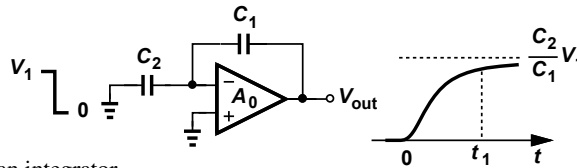


Figure 23.11 Linear settling in an integrator.

Solution

From the one-pole behavior assumed in Section 18.6.1, we write

$$V_{out}(t_1) = \left(1 - \exp \frac{-t_1}{\tau}\right) \frac{C_2}{C_1} V_1, \quad (23.26)$$

where $\tau = C_2/G_m$. The gain error is thus equal to $-\exp(-t_1/\tau)$.

The foregoing example predicts a considerable speed advantage if the gain error due to incomplete settling can be tolerated, a point of contrast to the necessary response of pipelined ADCs. For example, as explained in Chapter 18, a 14-bit MDAC settling to a 0.5-LSB error takes about 10 time constants. Equation (23.26), on the other hand, suggests that the settling time in an integrator can be reduced to 5τ if a gain error of 0.7% is acceptable.

Now, recall from Section 23.1.4 that the integrator(s) in a $\Delta\Sigma$ modulator can indeed tolerate a gain error of several percent without altering the noise-shaping function significantly. We thus conclude that incomplete integrator settling can be accommodated, allowing a higher clock frequency. Of course, the advantage is modest because the circuit is clocked M times faster than in a pipelined converter.

Nonlinear Settling The foregoing thoughts must be revisited if an integrator experiences nonlinear settling. Let us delve into this phenomenon.

We know from basic analog design that an op amp can “slew” during large-signal operation. For example, consider the simple integrator and its implementation shown in Fig. 23.12(a). The circuit begins with a zero initial condition on C_1 and senses a large input step, $-V_1 u(t)$. Neglecting other capacitances, we observe that V_X jumps down by the same amount, turning M_2 off. The circuit then reduces to that depicted in Fig. 23.12(b), where I_{SS} is copied by M_3 onto M_4 . Consequently, V_{out} rises at a constant slope (the “slew rate”), $I_{SS}/[C_1 C_2/(C_1 + C_2)]$, and the op amp slews until M_2 turns on.

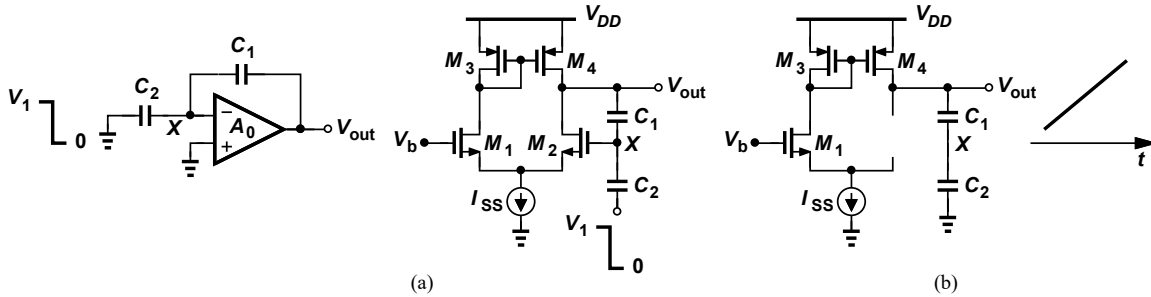


Figure 23.12 (a) Integrator and its implementation, and (b) behavior of op amp during slewing.

Slewing creates nonlinear settling. This point is illustrated in Fig. 23.13, where we apply an input step with different heights and examine the output behavior. In the RC circuit of Fig. 23.13(a), doubling the input swing exactly doubles the voltage at every point on the output waveform—as expected of a linear system. In the integrator of Fig. 23.13(b), on the other hand, the portion of V_{out} from $t = 0$ to $t = t_1$ climbs according to the slew rate and does *not* scale with the input amplitude.

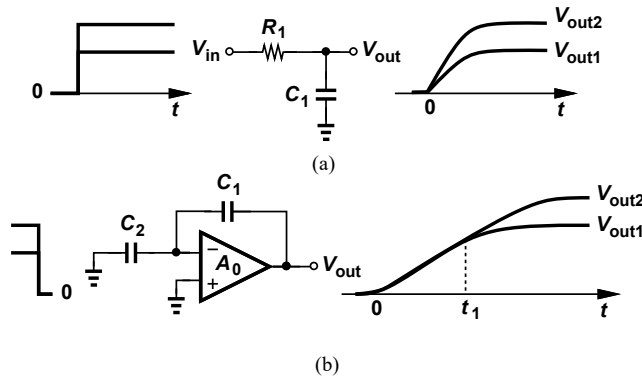


Figure 23.13 Response of (a) an RC circuit, and (b) an integrator to increasingly larger input steps.

The key point here is that incomplete integrator settling is tolerable if it is linear, i.e., if V_{out1} is sensed after $t = t_1$ in Fig. 23.13(b). This effect creates only a gain error. Even after M_2 in Fig. 23.12(a) turns on, we expect some nonlinear settling because most of I_{SS} still flows through M_1 and M_3 . We then ask, how exactly do we determine that the settling is sufficiently linear? The following example answers this question.

Example 23.6

Devise a test that quantifies the nonlinear settling of a circuit. Apply the test to the slewing waveforms of Fig. 23.13(b).

Solution

As shown in Fig. 23.14(a), we drive the circuit with step inputs of heights V_1 and $2V_1$, and measure the corresponding output waveforms. We typically select $2V_1$ such that V_{out2} reaches the maximum desirable or allowable swing. We then plot $2V_{out1} - V_{out2}$ as a function of time [Fig. 23.14(b)]. For a perfectly linear circuit, this difference is always equal to zero. In an actual circuit, however, it represents the nonlinear settling component. (If V_{out1} slews, we must choose a smaller value for V_1 and increase the scaling factor from 2 so that V_{out2} still reaches the full scale.) The residual nonlinear settling, $2V_{out1} - V_{out2}$, must decay to a

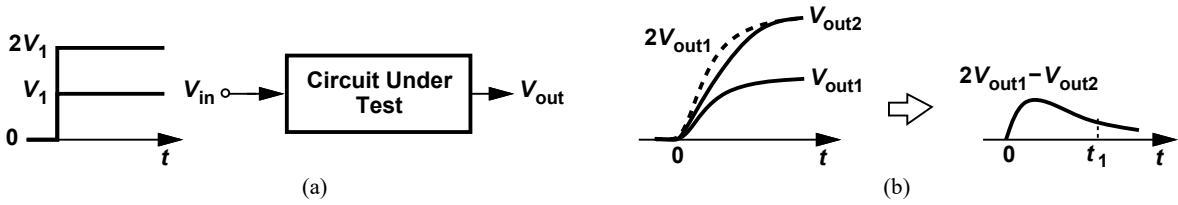


Figure 23.14 (a) Linear settling test for a given circuit, and (b) output and error waveforms.

sufficiently small value before we consider the circuit has entered linear settling. In our integrator example, we target $|2V_{out1} - V_{out2}| < (C_2/C_1) \times 1 \text{ LSB}$ at t_1 , where 1 LSB denotes the input-referred LSB of the converter.

Trade-Offs Given the small-signal time constant, $\tau = C_2/G_m$, in Eq. (23.26), we note that the settling speed trades with power consumption through the G_m of the op amp. Moreover, we also recognize a trade-off with kT/C noise through C_2 . In practice, the design proceeds as follows. Based on the CMOS process at hand, we assume a certain clock frequency and hence a certain oversampling ratio, M . Next, we select C_2 such that $kT/(MC_2)$ is sufficiently small. We then design the op amp so that it reaches linear settling in one half of the clock cycle while driving C_1 and C_2 in Fig. 23.12(a). If the op amp's speed falls short of the target, we must reduce f_{CK} and M and repeat the foregoing steps. We must also confirm that the op amp noise is low enough and may need to scale the op amp power consumption accordingly. Alternatively, one can seek a different op amp topology offering a higher slew rate.

Example 23.7

We select a value for C_2 and design an op amp, obtaining a certain closed-loop settling time and op amp noise. Explain what happens if we double the widths and the drain currents of all of the op amp's transistors, e.g., in Fig. 23.12(a). Note that this "linear" scaling maintains the transistor overdrives and the open-loop gain constant.

Solution

The doubling action multiplies the G_m of the op amp by the same factor, halving the settling time (if the op amp's internal poles do not manifest themselves). Thus, in principle, the clock frequency can be doubled. If we indeed manage to double f_{CK} , the effect of both kT/C noise and op amp noise is halved. These are the benefits afforded by doubling the power consumption and the op amp area.

23.1.7 Comparator Offset and Noise

Consider the first-order loop shown in Fig. 23.15(a), where V_{OS} denotes the offset of the comparator. To quantify the effect of V_{OS} , we observe that it can be viewed as a very low frequency Q-noise component, or it

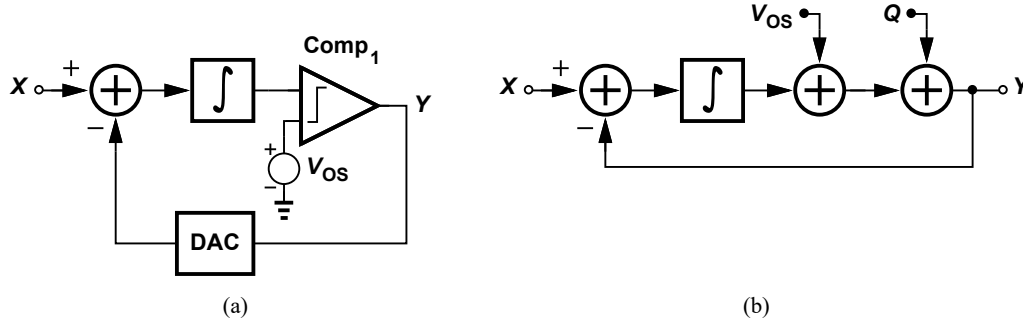


Figure 23.15 (a) First-order loop with comparator offset, and (b) equivalent block diagram.

can be simply injected into the loop as shown in Fig. 23.15(b). The transfer function from V_{OS} to the output is the same as Y/Q , assuming a high-pass shape given by $s/(1+s)$. Thus, an ideal integrator removes V_{OS} completely. With an actual integrator, this suppression factor is equal to $1 + A_0$, where A_0 denotes the gain of the op amp, revealing that the net offset at Y and also referred to X is equal to $V_{OS}/(1 + A_0)$. We then select this value such that it negligibly limits the ADC dynamic range.

The comparator's input-referred noise spectrum, $\overline{V_n^2}$, adds to the integrator output, thereby experiencing the same shaping function as does Q-noise. That is, $S_Y(f) \approx \overline{V_n^2} (2\pi f)^2$ for a first-order loop. We integrate this noise across the signal band and ensure that it is negligible.

23.1.8 Reference Settling

We have seen that Nyquist-rate converters must deal with the additional settling time arising from the finite output impedance of the reference generator. We expect that reference settling manifests itself in oversampling ADCs as well.

Recall from Chapter 21 that the 1-bit feedback DAC in a $\Delta\Sigma$ loop can be simply realized by two switches tied to V_{REF}^+ and V_{REF}^- [Fig. 23.16(a)]. After V_{in} is sampled, the left plate of C_2 swings to V_{REF}^+ or V_{REF}^- ,

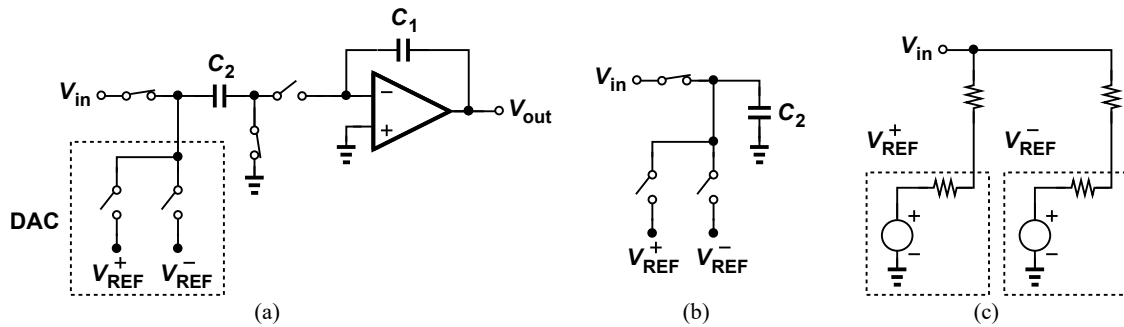


Figure 23.16 (a) Integrator sensing DAC levels, (b) simplified circuits, and (c) CT equivalent.

thereby depositing charge on C_1 in the amount of $(V_{in} - V_{REF}^+)C_2$ or $(V_{in} - V_{REF}^-)C_2$. Since the right plate of C_2 resides at a virtual ground, we can view the input network as shown in Fig. 23.16(b), recognizing that, in every clock period, C_2 takes charge from V_{in} and gives it to V_{REF}^+ or V_{REF}^- . As explained in Chapter 21, this process can be approximated by a continuous-time resistance [Fig. 23.16(c)], a reasonable model if f_{CK} is much greater than the maximum input frequency. We thus observe a coupling path between V_{in} and the reference generator(s) and conclude that V_{REF}^+ and V_{REF}^- suffer from *input-dependent* perturbations if their output impedance is not zero. Of course, if the references are allowed to settle completely, the perturbations

subside and this issue vanishes. However, high-frequency clocks necessary for a large oversampling ratio typically do not accord the circuit enough time to settle.

Input-dependent reference perturbations lead to distortion in the integrated signal. Thus, the reference generator must provide a sufficiently low output impedance. This circuit's design issues are similar to those described in previous chapters for Nyquist-rate ADCs.

It is possible to avoid this effect through the use of separate input capacitors. Illustrated in Fig. 23.17(a), the idea is to swing the left plate of C_2 from V_{in} to zero and implement reference subtraction by switching

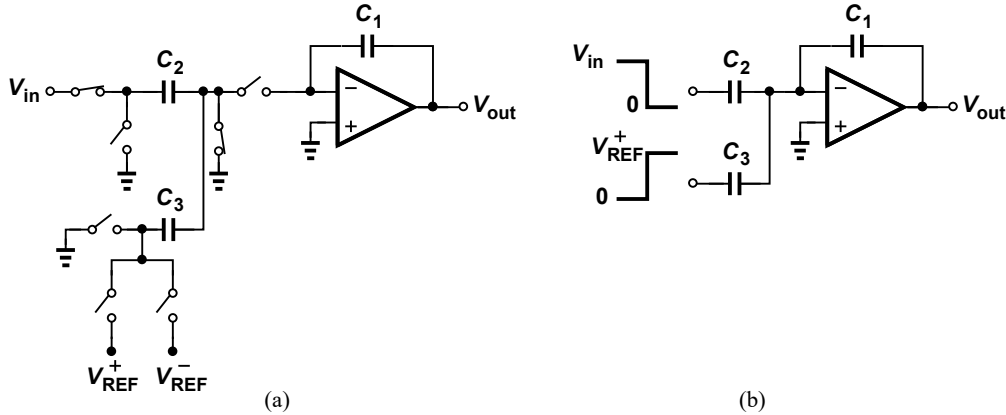


Figure 23.17 (a) Use of C_3 to avoid input-dependent DAC settling, and (b) equivalent circuit.

the left plate of another capacitor, C_3 , from zero to V_{REF}^+ or V_{REF}^- [Fig. 23.17(a)]. From Fig. 23.17(b), we have, for example,

$$V_{out} = \frac{C_2}{C_1} V_{in} - \frac{C_3}{C_1} V_{REF}^+. \quad (23.27)$$

One can select $C_3 = C_2$. Since C_3 does not switch between V_{in} and the references, the latter do not experience input-dependent bounce. Unfortunately, however, the circuit suffers from two drawbacks. First, the presence of $C_3 = C_2$ lowers the integrator loop gain (why?).⁷ Consequently, the closed-loop distortion due to op amp nonlinearity rises. Second, C_3 and its associated switches introduce additional kT/C noise.

Example 23.8

Does the lower loop gain in Fig. 23.17(a) also mean a longer settling time for the integrator?

Solution

Yes, but we can improve the situation. The lower loop gain yields a *greater* phase margin for the op amp (why?). We can then redesign the op amp's compensation network to reduce the PM and speed up the response.

23.1.9 Clock Jitter

Our study of sampling in Chapter 3 has led to simple SNR equations reflecting the impact of clock jitter. Oversampling converters employ a clock frequency that is M times higher. We must then determine how this clock's jitter affects the performance.

⁷ If the op amp input capacitance is neglected.

We know from Section 23.1.2 that the act of converting oversampled data to a Nyquist-rate stream can be roughly approximated by replacing every M samples with their average. Thus, the corruption due to the clock jitter is also reduced.

Consider an input sinusoid expressed by $x(t) = A \sin \omega_0 t$. In the presence of jitter, Δt_j , each sample experiences an error equal to $(dx/dt)\Delta t_j$. If M consecutive samples are averaged, the total noise power due to jitter is given by

$$P_j = \frac{1}{M} \left(\frac{dx}{dt} \right)^2 \overline{\Delta t_j^2}, \quad (23.28)$$

where $\overline{\Delta t_j^2}$ denotes the square of the rms jitter. It follows that oversampling lowers the jitter-induced noise power by a factor of M . While this appears to be another advantage of oversampling, we should remark that generating higher clock frequencies typically requires greater power consumption. In other words, a Nyquist-rate clock and its oversampling counterpart do not necessarily exhibit the same rms jitter if their generators draw the same amount of power.

23.2 Problem of Idle Tones

A $\Delta\Sigma$ modulator generates a binary output whose average value tracks its analog input. We naturally ask, what exactly happens if the input is *constant*? The output then exhibits a constant average while still toggling between logical high and low levels. But, if the loop has no random behavior, we also suspect that the output waveform is *periodic*, thus displaying harmonics.

To delve into this point, consider the first-order CT loop shown in Fig. 23.18(a) where the DAC provides levels equal to $\pm V_{REF}$, and suppose the comparator is strobed by the rising edge of CK . For illustration purposes, we assume that the comparator output toggles between -1 and $+1$. If $V_{in} = 0$, then the circuit reduces to that in Fig. 23.18(b). We know that the high gain of the comparator creates a virtual ground at the integrator output—but only on average. As, for example, $+V_{REF}$ is integrated, V_{int} rises above zero, the comparator generates a low level, and the integrator senses $-V_{REF}$ so that V_{int} begins to fall. Plotted in Fig. 23.18(c), the circuit's waveforms reveal that both V_{int} and V_{out} vary periodically and at a rate equal to $f_{CK}/2$. The output then contains the odd harmonics of $f_{CK}/2$.

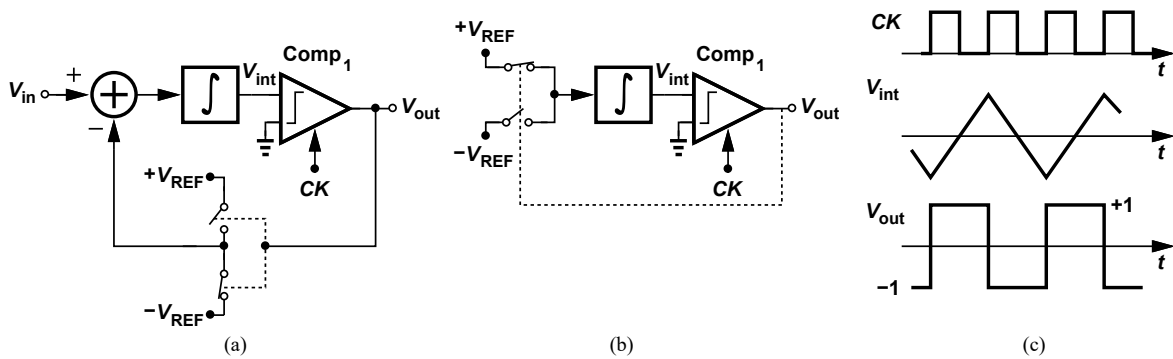


Figure 23.18 (a) First-order loop, (b) equivalent circuit with $V_{in} = 0$, and (c) loop waveforms.

Now, let us assume $V_{in} = +0.1V_{REF}$ in Fig. 23.18(a), recognizing that the integrator senses $+0.1V_{REF} - V_{REF} = -0.9V_{REF}$ or $+0.1V_{REF} + V_{REF} = 1.1V_{REF}$. For V_{out} to carry an average value of 0.1, it must remain positive for m cycles and negative for n cycles (Fig. 23.19) such that

$$\frac{mT_{CK} - nT_{CK}}{(m+n)T_{CK}} = 0.1. \quad (23.29)$$

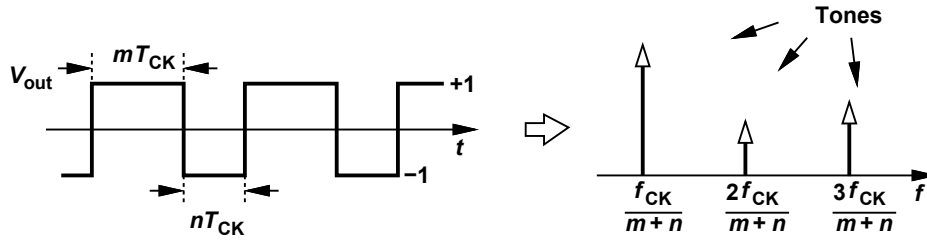


Figure 23.19 Periodic output in response to a dc input.

We have dealt with this situation in Section 22.3.2 and seen how m and n can be computed. Our principal concern here is that the modulator output is periodic, exhibiting a fundamental at $f_{CK}/(m+n)$ and its higher harmonics. These components are called “idle tones.”

Tones prove undesirable if they fall within the signal band. For example, a system processing audio signals must not produce any significant tones as the human ear is very sensitive to them.

We should make two remarks. First, tones appear in both DT and CT loops. Second, even if a $\Delta\Sigma$ modulator does not sense a constant input, it may still suffer from tones. This occurs if the input signal varies so slowly—compared to the clock—that the loop attempts to generate a periodic output whose average is roughly equal to the input.

We state without proof that the problem of tones is particularly acute in first-order modulators. For this reason, we opt for higher-order loops. In addition, it is possible to *randomize* the input signal so as to ensure the circuit does not display a periodic behavior. This is accomplished by adding a small random signal, called “dither,” to the input. Of course, dither corrupts the input.

For this reason, the dither added to the input must be eventually subtracted from the signal. Figure 23.20 shows a conceptual implementation, where random dither is added to V_{in} to ensure that the $\Delta\Sigma$ modulator does not fall into a periodic behavior. The dither is then subtracted from the output.

One can also envision a “high-pass” dither waveform that contains frequency components only outside the signal bandwidth.

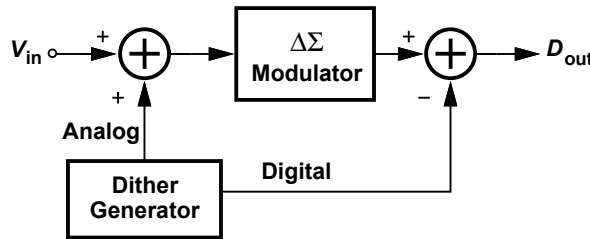


Figure 23.20 Use of dither to reduce tones.

23.3 Noise Folding Due to Nonlinearity

We have seen in previous chapters that $\Delta\Sigma$ modulators create a high-pass response for the quantization noise. While reducing the in-band noise, this mechanism also produces a peak at half of the sampling frequency, thus making the problem of “noise folding” particularly serious. We must first understand how nonlinearity translates high-frequency noise to low frequencies.

Suppose a nonlinear system described by $y(t) = \alpha_1 x(t) + \alpha_2 x^2(t)$ receives the sum of two sinusoids, $x(t) = A \cos \omega_1 t + A \cos \omega_2 t$. It follows that

$$y(t) = \alpha_1 A \cos \omega_1 t + \alpha_1 A \cos \omega_2 t + \alpha_2 A^2 \cos^2 \omega_1 t \quad (23.30)$$

$$2\alpha_2 A^2 \cos \omega_1 t \cos \omega_2 t + \alpha_2 A^2 \cos^2 \omega_2 t. \quad (23.31)$$

Of particular interest to us is the product term, which, upon expansion, leads to

$$y(t) = \alpha_2 A^2 \cos(\omega_1 + \omega_2)t + \alpha_2 A^2 \cos(\omega_1 - \omega_2)t + \dots \quad (23.32)$$

Depicted in Fig. 23.21(a) and called “folding,” this phenomenon generates a low-frequency output from two closely-spaced high-frequency inputs.

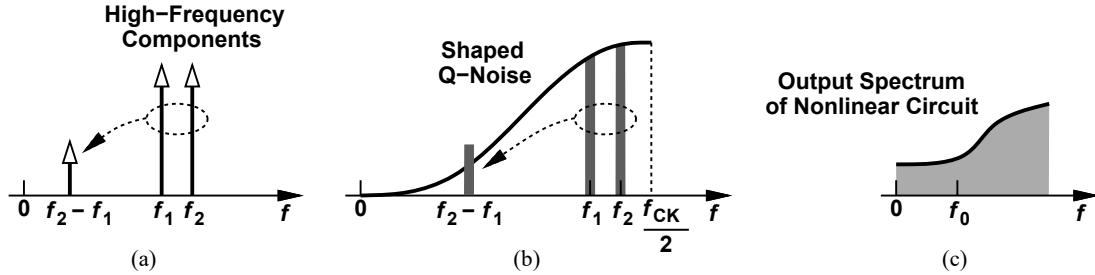


Figure 23.21 (a) Folding of a tone due to nonlinearity, (b) folding of continuous spectrum due to nonlinearity, and (c) resulting high noise floor at low frequencies.

Folding becomes problematic if strong high-frequency components are present, as is the case with shaped Q-noise. We observe from Fig. 23.21(b) that noise components near $f_{CK}/2$ introduce another at low frequencies. Considering different values for f_1 and f_2 , we arrive at the output spectrum shown in Fig. 23.21(c), where the in-band noise has risen due to folding.

Example 23.9

Does a system described by $y(t) = \alpha_1 x(t) + \alpha_3 x^3(t)$ exhibit noise folding?

Solution

We assume $x(t) = A_1 \cos \omega_1 t + A_2 \cos \omega_2 t$, noting that $y(t)$ now contains $3(A_1^2 \cos^2 \omega_1 t)(A_2 \cos \omega_2 t)$ and $3(A_1 \cos \omega_1 t)(A_2^2 \cos^2 \omega_2 t)$. Since $\cos^2 \theta = (1 + \cos 2\theta)/2$, these terms translate to components at $2\omega_1 \pm \omega_2$ and $2\omega_2 \pm \omega_1$ (why?) [Fig. 23.22(a)].⁸ Folding to low frequencies therefore does not occur if ω_1 and ω_2 are close to each other. One could select small values for both ω_1 and ω_2 so that $2\omega_1 - \omega_2$ and $2\omega_2 - \omega_1$ also fall near $\omega = 0$ [Fig. 23.22(b)], but the result is negligible in a shaped Q-noise spectrum because of its weak low-frequency components. However, if $2\omega_1$ is close to ω_2 , then $2\omega_1 - \omega_2$ lands in the signal band.

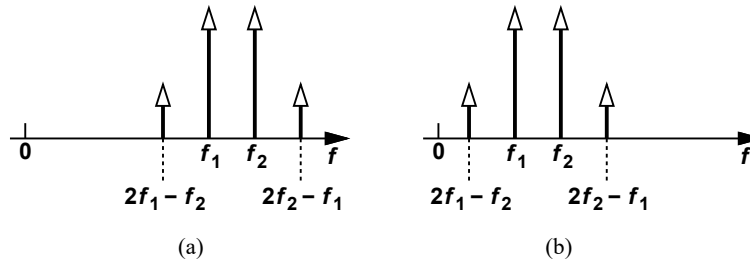


Figure 23.22 (a) Intermodulation between two tones, and (b) effect at low frequencies.

The foregoing study suggests that shaped Q-noise must avoid nonlinearity. But where exactly in a $\Delta\Sigma$ modulator can noise folding occur? We consider the first-order loop shown in Fig. 23.23, recognizing that

⁸In this figure, we have assumed $A_1 = A_2$.

the input subtractor senses the shaped noise. This spectrum also appears at the output of the subtractor. Thus, if this stage and/or the integrator exhibit nonlinearity, then the Q-noise is folded. In other words, the op amp used in the integrator must provide sufficient linearity to minimize noise folding. We typically design the op amp for a high open-loop gain so that its closed-loop linearity is adequate.

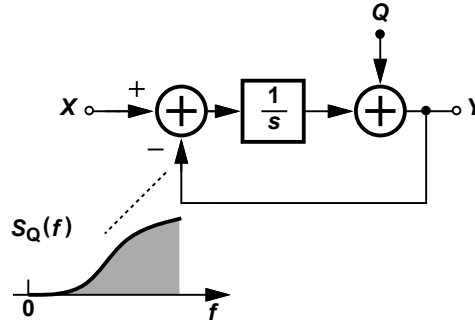


Figure 23.23 First-order loop including shaped Q-noise.

23.4 CT Modulator Nonidealities

Continuous-time Delta-Sigma modulator nonidealities The absence of sampling at the input of CT modulators avoids some of the DT loop imperfections but it also leads to a number of other issues. For example, effects such as capacitor mismatch and kT/C noise due to sampling are absent while issues such as the noise, finite gain, and nonlinearity of the op amp persist. Additionally, slewing in the op amp and clock jitter have a much more severe impact on CT loops. Also, the input-dependent delay of the comparator and the feedback DAC output noise corrupt the signal. We now study these phenomena.

23.4.1 Slewing at Op Amp Output

Our analysis of the DT integrator speed in Section 23.1.6 reveals that slewing at the output of the op amp introduces nonlinear settling. The integration time (i.e., the duration of the integration mode) must therefore be long enough for this effect to subside. Unfortunately, CT integrators do not offer this flexibility because they continuously integrate.

Consider the integrator shown in Fig. 23.24(a) and recall from Chapter 21 that the DAC's output average tracks V_{in}/R_1 . The circuit must generate the integral of V_{in} and I_{DAC} without slewing. An input of the form $V_{in} = V_0 \sin \omega_0 t$ causes a maximum output slope equal to $V_0/(R_1 C_1)$ (why?). Also, the DAC demands an output rate of I_0/C_1 . In the worst case, the op amp must support both and hence provide a slew rate greater than $V_0/(R_1 C_1) + I_0/C_1$. Otherwise, the integrator experiences nonlinearity, thereby folding high-frequency Q-noise (Section 23.3).

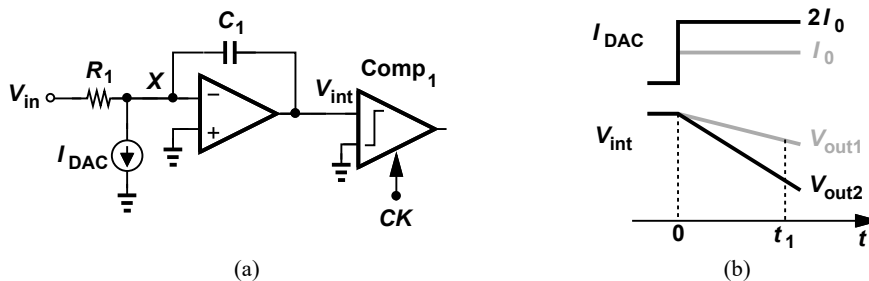


Figure 23.24 (a) CT $\Delta\Sigma$ modulator path, and (b) response of integrator to different DAC currents.

Unlike the DT integrator, the CT structure of Fig. 23.24(a) does not readily lend itself to linearity simulations. As a first step, we can set V_{in} to zero and apply steps of different heights for I_{DAC} [Fig. 23.24(b)]. As in the settling test described in Section 23.1.6, we obtain V_{out1} and V_{out2} for $I_{DAC} = I_0$ and $I_{DAC} = 2I_0$, respectively, form $2V_{out1} - V_{out2}$, and measure its maximum value between $t = 0$ and $t = t_1$. The value of t_1 is chosen such that $V_{out2}(t_1)$ is around the peak swing accommodated by the op amp. The maximum error must be sufficiently small. This test is incomplete as it does not include the input signal.

A more thorough test requires that we examine the integrator's output while the $\Delta\Sigma$ loop responds to an input. Using simulations and an actual op amp, we would study V_{int} and its spectrum to determine how much noise folding occurs.

23.4.2 Noise Sources

The CT integrator depicted in Fig. 23.24(a) contains three sources of electronic noise: R_1 , I_{DAC} , and the op amp. In contrast to the DT integrator, this circuit performs no sampling and does not exhibit kT/C noise directly, but the thermal noise of R_1 plays a similar role here. For an input signal bandwidth of f_0 , this noise amounts to $4kTR_1f_0$.

Example 23.10

Compare the noise contributed by R_1 in Fig. 23.24(a) to the kT/C noise in a DT integrator. Assume the integrators use equal feedback capacitors and the input signal occupies a bandwidth of f_0 .

Solution

From Section 23.1.2 the DT circuit exhibits an input-referred noise of $2kT/(MC_2)$, where C_2 is the sampling capacitor and M the oversampling ratio. Let us consider the simple topology shown in Fig. 23.25, recognizing

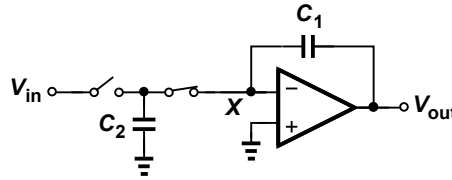


Figure 23.25 Simple integrator.

that the equivalent input resistance is approximately equal to $1/(f_{CK}C_2)$. For a fair comparison of CT and DT topologies, we assume they have equal input resistances: $1/(f_{CK}C_2) = R_1$. That is,

$$\frac{1}{C_2} = R_1 f_{CK}. \quad (23.33)$$

We now utilize this result to write the noise of the CT topology as $4kTR_1f_0 = 4kTR_1f_{CK}/(2M)$, which amounts to $2kT/(MC_2)$. The noise contributions are therefore the same in the CT and DT structures. However, the transient currents drawn by C_2 when the DT integrator enters the sampling mode make the design of the preceding buffer more difficult.

The feedback DAC in a CT loop contributes significant noise. Consider the simplified model shown in Fig. 23.26, where the current-steering DAC is lumped into one transistor having a current equal to $I_D = V_{REF}/R_1$ (Chapter 21). The thermal noise current spectrum of M_1 is given by

$$\overline{I_{n,DAC}^2} = 4kT\gamma g_m \quad (23.34)$$

$$= 4kT\gamma \frac{2I_D}{V_{GS} - V_{TH}} = \frac{8kT\gamma V_{REF}}{R_1(V_{GS} - V_{TH})}. \quad (23.35)$$

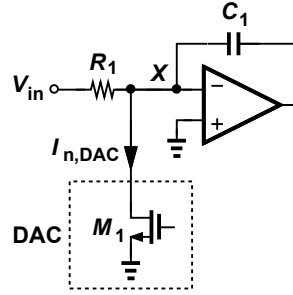


Figure 23.26 DAC representation for noise study.

This should be compared to the noise current produced by R_1 : $\overline{I_{n,R1}^2} = 4kT/R_1$. It follows that

$$\frac{\overline{I_{n,DAC}^2}}{\overline{I_{n,R1}^2}} = \frac{2\gamma V_{REF}}{V_{GS} - V_{TH}}. \quad (23.36)$$

Unfortunately, this ratio is typically greater than unity. For example, we have $\gamma \approx 1$, $V_{REF} = 0.5$ V, and $V_{GS} - V_{TH} = 0.2$ V.

Example 23.11

Repeat the above analysis for a differential CT loop.

Solution

Depicted in Fig. 23.27, the differential counterpart incurs thermal noise equal to $8kTR_1$ due to the resistors.

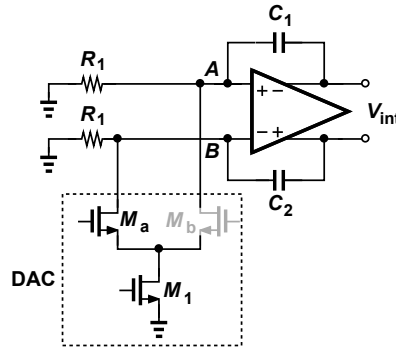


Figure 23.27 Effect of DAC noise in a differential CT integrator.

The noise current of M_1 flows through C_1 or C_2 , corrupting the output. Note that this noise is *not* a common-mode effect as it travels through only M_a or M_b . Thus,

$$\frac{\overline{I_{n,DAC}^2}}{\overline{I_{n,R1}^2}} = \frac{\gamma V_{REF}}{V_{GS} - V_{TH}}. \quad (23.37)$$

The ratio is lower here simply because the input resistors contribute more. As explained in Chapter 22, the DAC may incorporate a PMOS differential pair as well. The reader is encouraged to repeat the analysis for that case.

The noise of a CT integrator's op amp can be modeled as shown in Fig. 23.28, where, using superposition, we have

$$V_{out} = \frac{-V_{in}}{R_1 C_1 s} + \left(1 + \frac{1}{\frac{C_1 s}{R_1}} \right) V_n \quad (23.38)$$

$$= \frac{-V_{in}}{R_1 C_1 s} + \frac{V_n}{R_1 C_1 s} + V_n. \quad (23.39)$$

Since the input voltage and the op-amp input-referred noise experience approximately the same transfer function, we can say that V_n simply adds to V_{in} . That is, V_n is integrated across the input signal bandwidth.

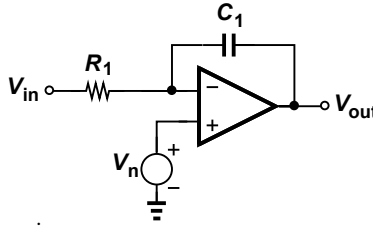


Figure 23.28 CT integrator with op-amp noise.

23.4.3 Resistive Load at Op Amp Output

Unlike DT modulators, high-order CT loops present a resistive load to their integrator(s), making op-amp design more difficult. For example, the second-order system shown in Fig. 23.29 requires that the first op

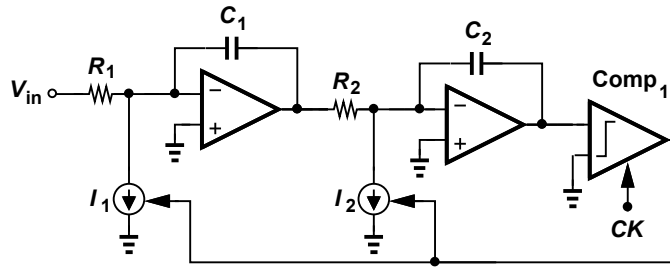


Figure 23.29 Second-order CT loop.

amp drive R_2 while maintaining adequate open-loop gain and linearity. Note that this issue is much less problematic in the DT counterpart of Fig. 23.4(a) because, after the second integrator's input capacitors charge, they draw little current from the first op amp.

In order to minimize this loading effect, we must select a large value for R_2 in Fig. 23.29, but only to the extent that its thermal noise (referred to the first integrator's input with first-order shaping) is negligible. We then design the first op amp such that it can drive R_2 .

23.4.4 Clock Jitter

As we have seen in Chapter 3 and Section 23.1.9, clock jitter corrupts sampled signals. In CT $\Delta\Sigma$ modulators, jitter creates an *additional* difficulty. Consider the loop illustrated in Fig. 23.30(a), assuming the sampling of the signal occurs within the comparator. That is, when CK goes high, $Comp_1$ makes a decision based on the instantaneous value of V_{int} and drives I_{DAC} accordingly. Suppose the DAC delivers a current to the integrator's capacitor for one clock cycle [Fig. 23.30(b)].⁹ In the presence of clock jitter, Δt_j , the charge

⁹The comparator delay is neglected in this illustration.

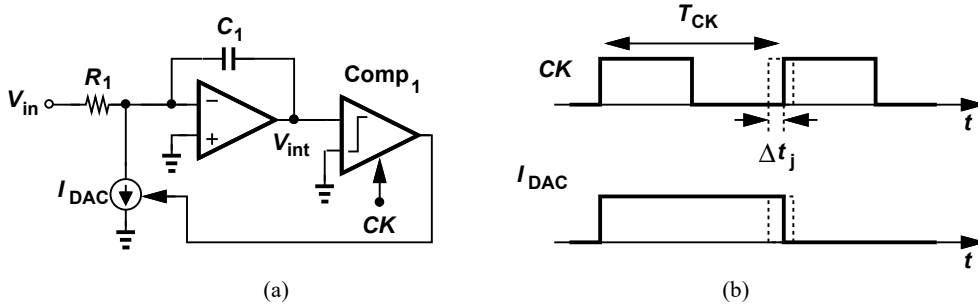


Figure 23.30 (a) First-order CT loop, (b) effect of comparator clock jitter on DAC current.

provided by the DAC in one clock cycle is corrupted by an amount equal to $I_0\Delta t_j$. The equivalent current corruption is given by $I_0\Delta t_j/T_{CK}$, and the input-referred rms noise voltage by $I_0R_1\Delta t_j/T_{CK}$ if Δt_j denotes the rms jitter. Thus, the overall input-referred noise due to this jitter mechanism emerges as

$$V_{n,in} = \frac{I_0R_1\Delta t_j}{T_{CK}} \quad (23.40)$$

$$= \frac{V_{REF}\Delta t_j}{T_{CK}}. \quad (23.41)$$

Let us assume that this noise occupies a bandwidth equal to $f_{CK}/2$, and that its power outside $f_0 = f_{CK}/(2M)$ is removed by the digital filter following the modulator:

$$V_{n,jit}^2 = \frac{V_{REF}^2}{M} \left(\frac{\Delta t_j}{T_{CK}} \right)^2. \quad (23.42)$$

The following example shows how serious clock jitter can be.

Example 23.12

A 14-bit ADC must digitize a 20-MHz signal. If $M = 64$, determine the maximum tolerable jitter for an SNR penalty of 1 dB.

Solution

For a 1-dB SNR penalty, we write the total noise with and without jitter and select the ratio to be 1 dB:

$$10 \log \frac{\frac{\Delta^2}{12} + V_{n,jit}^2}{\frac{\Delta^2}{12}} = 1 \text{ dB}, \quad (23.43)$$

where $\Delta = V_{REF}/2^{14}$ and $V_{n,jit}^2 = (V_{REF}\Delta t_j/T_{CK})^2/M$. Thus,

$$V_{n,jit}^2 = 0.259 \frac{\Delta^2}{12} \quad (23.44)$$

and hence

$$\Delta t_j^2 = \frac{0.259}{12 \times 2^{28}} \times (64T_{CK}^2). \quad (23.45)$$

We have $T_{CK} = (1/20 \text{ MHz})/(2M) = 390 \text{ ps}$ and

$$\Delta t_j = 28 \text{ fs}_{\text{rms}}. \quad (23.46)$$

Such a low jitter is difficult to achieve for clock generators. (But recent work on low-jitter phase-locked loops has demonstrated jitter values around $20 \text{ fs}_{\text{rms}}$.) The reader is encouraged to compute the tolerable jitter for a DT counterpart and note that it can be much larger.

The problem of clock jitter is greatly alleviated if the amount of charge delivered by the DAC in Fig. 23.30(a) becomes independent of the clock period. This can be accomplished as depicted in Fig. 23.31(a), where the DAC employs a capacitor whose left plate swings between zero and V_{REF} according to the comparator's decision. Viewing C_D , C_1 , and the op amp as an amplifier and using superposition, we recognize

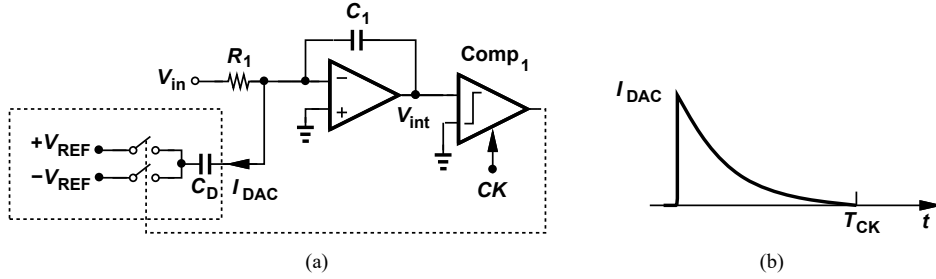


Figure 23.31 (a) Use of a capacitor DAC to suppress effect of clock jitter, and (b) DAC current waveform.

that V_{int} changes by $\pm(C_D/C_1)V_{\text{REF}}$ and remains free from the effect of clock jitter. In such a case, the area under the curve becomes independent of T_{CK} if C_D discharges (almost) completely.

The value of C_D in Fig. 23.31(a) is chosen as follows. We return to the basic topology in Fig. 23.30(a) and observe that the change in V_{int} is given by $I_{\text{DAC}}T_{\text{CK}}/C_1$ and $I_{\text{DAC}} = \pm V_{\text{REF}}/R_1$. Equating the changes,

$$\frac{I_{\text{DAC}}T_{\text{CK}}}{C_1} = \frac{C_D}{C_1}V_{\text{REF}} \quad (23.47)$$

and replacing I_{DAC} with V_{REF}/R_1 , we have

$$C_D = \frac{T_{\text{CK}}}{R_1}. \quad (23.48)$$

We identify two issues here. First, the transient current drawn by C_D partially flows through R_1 (because the op amp may not be able to provide all of it), disturbing the preceding stage. Second, C_D lowers the integrator loop gain.

23.4.5 Comparator Delay

As explained in Chapter 6, comparators take a finite time to generate a valid logical output after they are clocked. This delay degrades the performance of CT modulators.

Let us first assume a constant comparator delay, ΔT_C , and construct the model shown in Fig. 23.32 for a first-order loop. How does ΔT_C affect the loop's response? The transfer function of this delay unit is given by $\exp(-s\Delta T_C)$ (why?), which we approximate by $1 - s\Delta T_C$ if the delay is sufficiently small (i.e., much

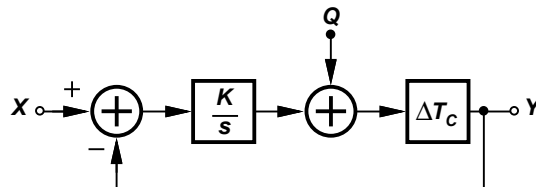


Figure 23.32 First-order loop with comparator delay.

less than the clock period). It follows that

$$\left[(X - Y) \frac{K}{s} + Q \right] (1 - s\Delta T_C) = Y. \quad (23.49)$$

We thus have

$$Y(s) = \frac{K(1 - s\Delta T_C)}{(1 - K\Delta T_C)s + K} X(s) + \frac{s(1 - s\Delta T_C)}{(1 - K\Delta T_C)s + K} Q(s). \quad (23.50)$$

The signal and noise transfer functions are slightly altered. Both contain a zero at $1/\Delta T_C$, a very high frequency. Moreover, their pole magnitude is raised to $K/(1 - K\Delta T_C)$. These changes are fairly benign.

In higher-order modulators, however, comparator delay can prove problematic. We recognize from the approximation $\exp(-s\Delta T_C) \approx 1 - s\Delta T_C$ that this phenomenon introduces an open-loop zero in the *right half plane*, degrading the phase margin as a left-half-plane pole would. This is seen by writing the transfer function as $1 - j\omega\Delta T_C$ and its phase as $-\tan^{-1}(\omega\Delta T_C)$. Alternatively, since $1 - \epsilon \approx 1/(1 + \epsilon)$ for $|\epsilon| \ll 1$, we have $1 - s\Delta T_C \approx 1/(1 + s\Delta T_C)$, as if the system contained a pole at $-1/\Delta T_C$. The key point here is that an excessive comparator delay can lead to unwanted peaking in the signal and noise transfer functions and even to instability. It can also cause large voltage swings at the output of the integrator(s).

The foregoing thoughts point to a more general issue called the “excess loop delay.” Any circuit block within the loop that introduces delay can cause peaking or instability, requiring attention. From the above formulation, we note that the total excess delay, ΔT , must be small enough so that the phase contributed by the zero at $1/\Delta T$ negligibly affects the phase margin.

Another difficulty arises from the input-dependent delay of the comparator. We know from Chapter 6 that regenerative amplification takes a longer time for smaller input differences. In a CT loop, therefore, the comparator drives the DAC with different delays in different cycles. Illustrated in Fig. 23.33, this phenomenon causes I_{DAC} to switch with a delay of ΔT_{C1} in the first cycle and ΔT_{C2} in the next. That is, the DAC pulsewidth is modulated, as if the clock had jitter.

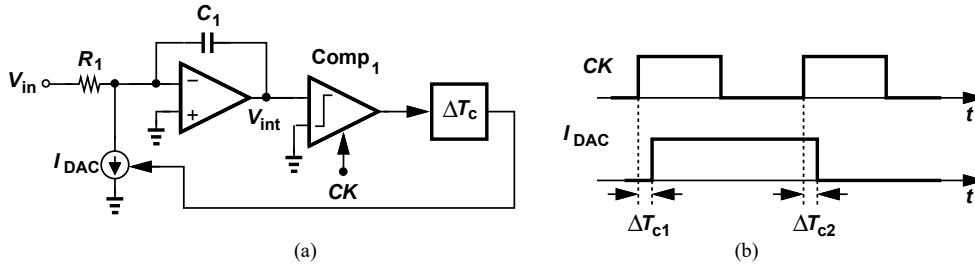


Figure 23.33 Effect of comparator delay on a first-order loop.

Does this error translate to noise or distortion? The delay is a function of the input difference sensed by Comp_1 , which itself contains the integral of both V_{in} and the Q-noise. We must then ponder how this effect can be formulated.

Consider the simpler scenario depicted in Fig. 23.34, where the delay, ΔT_C , is a function of the input. For

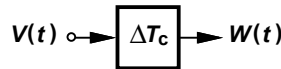


Figure 23.34 Simple delay stage.

example, $\Delta T_C = g[V(t)]$. The output is expressed as $W(t) = V(t - \Delta T_C) \approx V(t) - \Delta T_C dV/dt$ if ΔT_C is small.¹⁰ It follows that

$$W(t) \approx V(t) - g[V(t)] \frac{dV}{dt}, \quad (23.51)$$

¹⁰Note that this approximation is the same as $\exp(-s\Delta T_C) \approx 1 - s\Delta T_C$ (why?).

revealing that $W(t)$ is a nonlinear function of $V(t)$ even if $g[V(t)]$ is linear. Returning to the $\Delta\Sigma$ loop of Fig. 23.33, we observe that this nonlinearity leads to two issues. First, the signal component at the modulator's output experiences distortion. Second, more importantly, the high-frequency components present in V_{int} , i.e., those due to Q-noise, are folded down.

Example 23.13

Does the DAC of Fig. 23.31(a) alleviate the problem of input-dependent comparator delay?

Solution

Yes, it does. Since the charge delivered to the integrator is independent of ΔT_C , the nonlinearity does not affect the loop. This DAC is therefore preferable here.

It is possible to avoid this issue by updating the DAC some time after the comparator is clocked. For example, the comparator and the DAC can operate on opposite edges of the main clock.

23.5 Stability Considerations

Our development of $\Delta\Sigma$ modulators has mostly assumed that the quantizer exhibits a unity gain and simply adds quantization noise. For stability analysis, however, we must take into account its highly nonlinear behavior. For example, we know from Chapter 21 that the quantizer gain drops as its input signal amplitude increases.

In this context, we must distinguish among three phenomena. First, as explained in Section 23.2, a $\Delta\Sigma$ loop can enter a periodic behavior and generate “tones,” e.g., with a dc input. We may or may not consider this situation “unstable.” Second, the integrator(s) can experience “overload,” whereby their output swing becomes excessively large. This can be alleviated through the use of integrator gain scaling (Chapter 22). Third, the loop may begin to oscillate and the internal amplitudes grow to the point of causing heavy nonlinearity in op amps.

The analysis of stability is beyond the scope of this book, but first-order and second-order loops are generally considered more stable in both their DT and CT embodiments than higher-order loops.

Problems

- 23.1. Upon examining Eq. (23.4), a student reasons that the noise-shaping function can be of *second* order if $\Delta C = C$. Is that correct? Calculate the signal transfer function in such a case.
- 23.2. Suppose the feedback path around the second integrator in Fig. 23.4(b) is omitted, i.e., Y returns to only the first subtractor. Determine the input-referred contribution of V_n .
- 23.3. In a hypothetical design, the input signal bandwidth, f_0 , is *less* than the zero frequency in Fig. 23.7(d). Compute the integrated noise in this case by neglecting the effect of the zero. Simplify the result if $A_0 = 2M$.
- 23.4. Determine the minimum acceptable A_0 if the penalty in Eq. (23.23) must be no more than 5%. If $M = 64$, how much is the required op-amp gain?
- 23.5. Consider a second-order discrete-time $\Delta\Sigma$ loop. Suppose the op amp in the second integrator exhibits an input-referred offset equal to V_{OS} . Refer this offset to the input of the overall modulator. For simplicity, assume all of the capacitors in the integrators are equal.
- 23.6. Determine the slew rate in Fig. 23.12(a) if the left plate of C_2 jumps from zero to positive voltage V_1 .
- 23.7. Suppose we double the value of C_2 in Fig. 23.12(a). Explain what happens to the nonlinear and linear settling times of V_{out} .
- 23.8. Repeat the previous problem if we double only C_1 .
- 23.9. The DAC in Fig. 23.15(a) exhibits an offset equal to V_1 at its output. Determine the effect of this offset on the $\Delta\Sigma$ loop behavior.
- 23.10. Redraw the waveforms shown in Fig. 23.18(c) if $V_{in} = 0.2V_{REF}$.
- 23.11. A nonlinear (differential) system is characterized by $y(t) = \alpha \tanh[\beta x(t)]$. Does the folding effect depicted in Fig. 23.21(b) occur in such a system? Explain.
- 23.12. Compute the total kT/C noise at the output of the integrator shown in Fig. 23.17(a).
- 23.13. If $V_{in} = V_0 \sin \omega_0 t$ in Fig. 23.24(a) and I_{DAC} jumps from $-V_{REF}/R_1$ to $+V_{REF}/R_1$, determine the maximum slew rate required of the op amp.
- 23.14. We halve the value of R_1 in Fig. 23.26 and double the DAC current. Explain how the input-referred noise changes.
- 23.15. Consider the simple CT integrator shown in Fig. 23.28, assuming that the op amp can be modeled by a Norton equivalent having a transconductance, G_m , and an output resistance, R_{out} . Determine the transfer function of the circuit, sketch its magnitude, and compare the results to the case of an ideal op amp.
- 23.16. A 16-bit CT modulator must digitize a 1-MHz signal. If $M = 128$, determine the maximum tolerable clock jitter for an SNR penalty of 2 dB.
- 23.17. Consider the comparator delay model shown in Fig. 23.32, but assume a second-order CT modulator. Determine Y/X and Y/Q if the delay transfer function is approximated by $1 - s\Delta T$.
- 23.18. Repeat the previous problem if the integrator in Fig. 23.32 is replaced with K/s^2 . Sketch the open-loop magnitude and phase of Y/X .
- 23.19. Can the loop shown in Fig. 23.32 become unstable if ΔT becomes arbitrarily large? For large ΔT , the approximation $\exp(-s\Delta T_C) \approx 1 - s\Delta T_C$ does not hold. Model the delay by the exponential, determine the open-loop transfer function, replace s with $j\omega$, and note that $\exp(-j\omega\Delta T_C) = \cos(\omega\Delta T_C) - j\sin(\omega\Delta T_C)$. Now, sketch the magnitude and phase of the open-loop transfer function and study the stability.
- 23.20. If $V(t) = V_1 \cos \omega_1 t + V_2 \cos \omega_2 t$ in Fig. 23.34 and $\Delta T_C = \alpha V(t)$, determine the folded component (at $\omega_1 - \omega_2$) at the output. Assume ΔT_C is small.

Advanced Oversampling ADCs

The previous three chapters have laid the foundation for the analysis and design of oversampling ADCs. In this chapter, we study advanced methods that improve their performance. These include cascaded loops, modulators with multibit quantizers, and feedforward techniques. We also deal with “decimation filtering,” the operation that removes high-frequency quantization noise and converts the oversampled output to Nyquist-rate data.

24.1 Cascaded Modulators

In our analysis of $\Delta\Sigma$ modulators, we have seen that Q-noise can be reduced using two techniques, namely, by raising the order of the loop and/or by increasing the oversampling ratio, M . The former faces difficulties such as instability and large integrator swings, and the latter is limited by the speed of the op amps. In this section, we study a third method of lowering Q-noise.

24.1.1 Basic Idea

The output of a $\Delta\Sigma$ loop contains the shaped Q-noise. Is it possible to *cancel* this noise by recreating and subtracting it from the output?! Illustrated in Fig. 24.1(a), this approach requires that we reconstruct the Q-noise such that it is *free* from the signal. The subtraction then removes only the noise.

We now turn to the first-order loop shown in Fig. 24.1(b) and ask how the Q-noise can be reconstructed. Since the comparator adds this noise to V_{int} , we expect that $Q = Y - V_{int}$ and consider the arrangement of Fig. 24.1(c). But one question arises here. The integrator output is an analog signal but Y is a digital stream. How do we subtract these two signals?

The question is readily addressed by noting that the DAC output in Fig. 24.1(c) is an analog copy of Y and can be combined with V_{int} so as to reproduce Q [Fig. 24.1(d)]. We expect that this subtractor’s implementation is similar to that receiving X . However, we recognize another difficulty: Y is a digital quantity but Q an analog signal. How do we subtract Q from Y ? This necessitates that we digitize Q before combining it with Y , as depicted in Fig. 24.1(e). We say the first modulator and the second ADC are “cascaded.” Such topologies are also called “**multistage-noise shaping**”(MASH) architectures.

Example 24.1

A student looking at Fig. 24.1(c) realizes that the Q output represents the noise added by Comp_1 , i.e., the *unshaped* Q-noise, whereas the noise present in Y is *shaped*. How can these two outputs be combined so as to cancel the noise?

Solution

Indeed, in a discrete-time loop, we have $Y = z^{-1}X + (1 - z^{-1})Q$. The reconstructed Q output must then

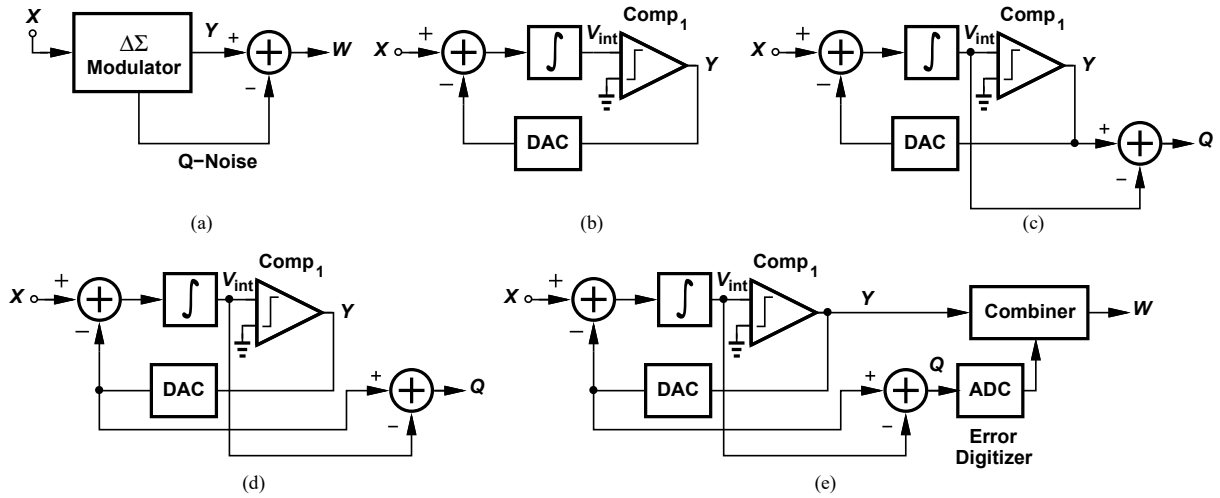


Figure 24.1 (a) Conceptual Q-noise cancellation, (b) first-order loop, (c) arrangement for Q-noise cancellation, (d) subtraction of analog quantities, and (e) complete Q-noise cancellation scheme.

be multiplied by $1 - z^{-1}$ (i.e., applied to a digital differentiator) before it is subtracted from Y . We return to this point below.

Example 24.2

Rather than follow the architecture of Fig. 24.1(e), a student surmises that one could digitize V_{int} in Fig. 24.1(b) and then subtract Y from it. Is that possible?

Solution

Yes, it is. However, this method must digitize both the *signal* and Q-noise, thus necessitating a wider dynamic range for this ADC.

While the second ADC in Fig. 24.1(e) can assume any architecture, we prefer to realize it as a $\Delta\Sigma$ modulator, too. In this spirit, let us construct the “1-1 cascade” shown in Fig. 24.2, where the term “1-1” refers to the orders of the two modulators. (The DACs are not shown for simplicity.) Since the “error digitizer” senses $Y - V_{int1} = Q_1$, we have

$$Y(s) = \frac{K_1}{s + K_1} X(s) + \frac{s}{s + K_1} Q_1(s), \quad (24.1)$$

$$B(s) = \frac{K_2}{s + K_2} Q_1(s) + \frac{s}{s + K_2} Q_2(s). \quad (24.2)$$

Let us assume $K_1 = K_2$. For proper cancelation of $Q_1(s)$, we must multiply Eq. (24.2) by s/K_2 and subtract the result from Eq. (24.1). As explained in Example 24.1, this means that the output of the second modulator must be differentiated. It follows that Y and B must be combined according to

$$Y(s) - \frac{s}{K_2} B(s) = \frac{K_1}{s + K_1} X(s) - \frac{s^2}{K_2(s + K_2)} Q_2(s). \quad (24.3)$$

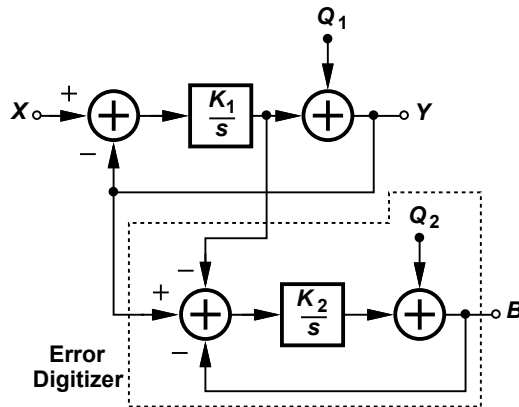


Figure 24.2 Cascade of first-order loops.

Interestingly, the cascaded first-order modulators, along with this combining action, provide *second-order* noise shaping.

Example 24.3

Construct a discrete-time counterpart of the 1-1 cascade shown in Fig. 24.2(b).

Solution

We must first decide how Y should be subtracted from V_{int1} . Consider the integrator shown in Fig. 24.3(a), where $C_4 = C_5$ and the left plates of C_4 and C_5 swing from V_{int1} and $-V_{DAC}$ to zero, respectively.

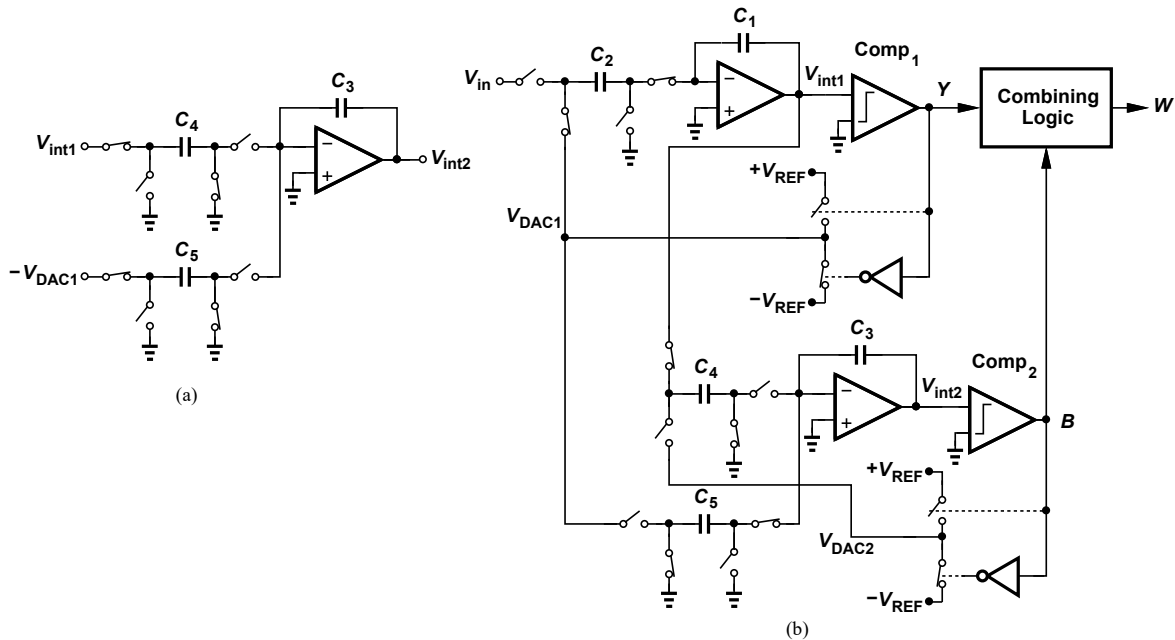


Figure 24.3 (a) Subtraction of two voltages, and (b) 1-1 cascade example.

Thus, V_{int2} rises by an amount equal to $(C_4/C_3)(V_{int1} - V_{DAC1})$, signifying subtraction. Shown in Fig. 24.3(b) is the 1-1 realization. Note that the left plate of C_5 switches from zero to V_{DAC1} . The circuit operates as follows. After the first loop enters the integration mode, V_{int1} settles and Comp_1 is clocked. During this time, C_4 and C_5 track V_{int1} and zero, respectively. Next, the second loop begins integration while C_4 senses V_{DAC2} and C_5 switches to V_{DAC1} . The second integrator output thus changes by

$(V_{DAC2} - V_{int1})(C_4/C_3) + V_{DAC1}(C_5/C_3)$. Selecting $C_4/C_3 = C_5/C_3$, we observe that V_{int2} carries $-V_{int1} + V_{DAC1}$ and hence the Q-noise of Comp_1 . This component finds its way to B and is removed by the combining logic.

As explained in Example 24.1, the second modulator's output must be shaped by a $1 - z^{-1}$ function before it is combined with Y . We deal with the details in Section 24.1.3.

24.1.2 Higher-Order Cascades

The developments in the previous section encourage us to consider cascades of higher-order $\Delta\Sigma$ modulators so as to achieve sharper noise shaping. We limit our scope here to second-order loops as they face less serious stability issues than those having higher orders.

Consider the 2-1 MASH architecture illustrated in Fig. 24.4. The reader can express the output of the first

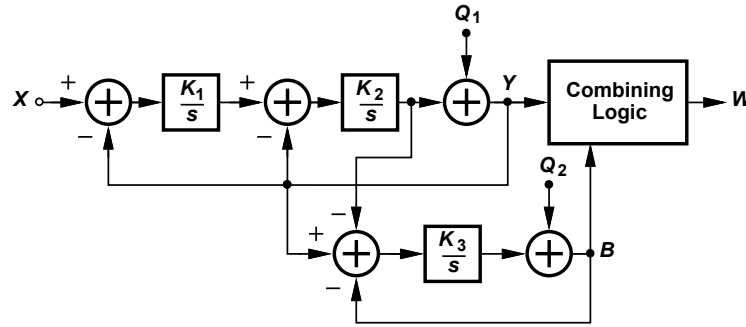


Figure 24.4 Cascade of second-order and first-order loops.

modulator as

$$Y(s) = \frac{K_1 K_2}{s^2 + K_2 s + K_1 K_2} X(s) + \frac{s^2}{s^2 + K_2 s + K_1 K_2} Q_1(s). \quad (24.4)$$

To reconstruct $Q_1(s)$, we subtract Y from the second integrator output. We then apply the result to the second $\Delta\Sigma$ loop, which, according to Eq. (24.2) produces:

$$B(s) = \frac{K_3}{s + K_3} Q_1(s) + \frac{s}{s + K_3} Q_2(s). \quad (24.5)$$

In order to cancel $Q_1(s)$, we must multiply $B(s)$ by $(s + K_3)s^2$, divide it by $K_3(s^2 + K_2 s + K_1 K_2)$ and subtract it from $Y(s)$. It follows that

$$W(s) = \frac{K_1 K_2}{s^2 + K_2 s + K_1 K_2} X(s) - \frac{s^3}{K_3(s^2 + K_2 s + K_1 K_2)} Q_2(s). \quad (24.6)$$

The cascade therefore exhibits third-order noise shaping. Of course, we must ponder how multiplication by $K_3^{-1}(s + K_3)s^2/(s^2 + K_2 s + K_1 K_2)$ is accomplished in the digital domain. This is a formidable challenge, especially for CT realizations because their integrator gains, of the form $1/(RC)$, vary with PVT conditions. In DT cascades, on the other hand, the gains are given by the ratio of capacitors and are much less PVT-dependent.

Let us now study the 1-2 cascade shown in Fig. 24.5. The first modulator's output is written as

$$Y(s) = \frac{K_1}{s + K_1} X(s) + \frac{s}{s + K_1} Q_1(s) \quad (24.7)$$

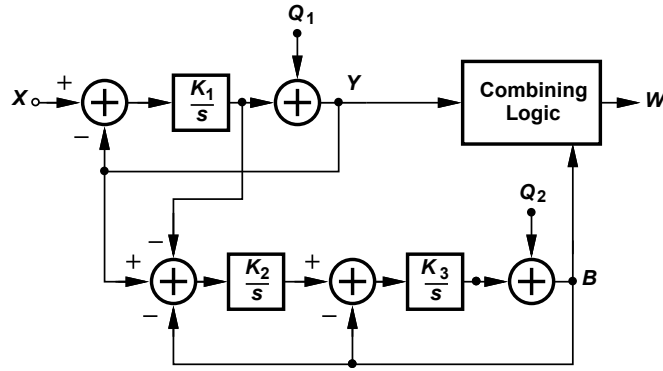


Figure 24.5 Cascade of first-order and second-order loops.

and that of the second as

$$B(s) = \frac{K_2 K_3}{s^2 + K_3 s + K_2 K_3} Q_1(s) + \frac{s^2}{s^2 + K_3 s + K_2 K_3} Q_2(s). \quad (24.8)$$

In this case, $B(s)$ must be multiplied by $[(s^2 + K_3 s + K_2 K_3)/(K_2 K_3)][s/(s + K_1)]$ and subtracted from $Y(s)$. We then have

$$W(s) = \frac{K_1}{s + K_1} X(s) - \frac{s^3}{K_2 K_3 (s + K_1)} Q_2(s). \quad (24.9)$$

Alternatively, we can multiply $Y(s)$ by $K_2 K_3/(s^2 + K_3 s + K_2 K_3)$ and $B(s)$ by $s/(s + K_1)$ and subtract the results.

Given that the cascades in Figs. 24.4 and 24.5 both offer third-order noise shaping, the reader may wonder which one is preferable. This relates to the problem of gain error, studied next.

24.1.3 Problem of Gain Error

Our treatment of 1-1, 2-1, and 1-2 cascades has assumed that the Q-noise of the first modulator is exactly canceled. In reality, analog imperfections lead to finite cancellation and hence degraded performance.

CT Cascade Let us begin with the 1-1 cascade shown in Fig. 24.2 and recall that cancellation of Q_1 requires that $K_1 = K_2$. That is, the integrators in the two stages must provide equal gains. For a simple CT realization and ideal op amps, this means that the RC values of the two integrators must match with high precision, a reasonable condition. In practice, the RC values exhibit mismatches well below 1%.

However, another phenomenon enters the picture that is much more severe. Recall from Example 24.1 that the second modulator output in Fig. 24.2 must be *differentiated* before it is combined with Y (Fig. 24.6). How is this implemented in the digital domain? Since the output, $B(s)$, is a digital bit stream, we surmise that

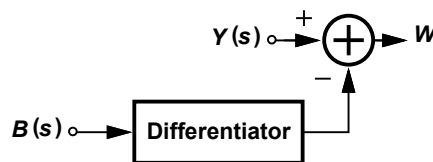


Figure 24.6 Processing of $B(s)$ before combining.

differentiation assumes a transfer function of the form $1 - z^{-1}$. To see how well this expression approximates multiplication by s , we treat z^{-1} as a one-cycle delay and write it as $\exp(-sT_{CK}) \approx 1 - sT_{CK}$. Thus, $1 - z^{-1} \approx sT_{CK}$, yielding a “gain” of T_{CK} for the differentiator. In Eq. (24.2), on the other hand, $B(s)$ must

be multiplied by s/K_2 , and hence we must have

$$\frac{1}{K_2} = T_{CK}. \quad (24.10)$$

This condition proves far more difficult to implement than the integrator gain match seen above. This is because Eq. (24.10) demands that the *absolute* value of the second integrator's time constant match the clock period. The challenge is then to calibrate the RC value so that it is sufficiently close to T_{CK} . We view this effect as a gain error between the analog integrator and the digital differentiator.

Example 24.4

How does the finite gain of the op amp affect the Q-noise cancelation?

Solution

We have seen in Chapter 23 that, with an op-amp gain of A_0 , the Q-noise is shaped as follows:

$$\frac{Y}{Q}(s) = \frac{1}{1 + A_0} \frac{(1 + A_0)R_1C_1s + 1}{R_1C_1s + 1} \quad (24.11)$$

$$= \frac{1}{1 + A_0} \frac{(1 + A_0)s + K_1}{s + K_1}, \quad (24.12)$$

where $K_1 = 1/(R_1C_1)$. This transfer function differs from the nominal, $s/(s + K_1)$, because its zero is located at $K_1/(1 + A_0)$ rather than at the origin. In other words, $K_1/(1 + A_0)$ must be sufficiently small to obtain reasonable cancelation. The reader is encouraged to repeat this example for the finite bandwidth of the op amp, e.g., if its open-loop transfer function is equal to $A_0/(1 + s/\omega_0)$.

The foregoing “analog-digital mismatch” issue becomes more serious in the 2-1 and 1-2 cascades described previously. The former requires that $B(s)$ be multiplied by $(s + K_3)s^2/(s^2 + K_2s + K_1K_2)$ in the digital domain. For proper cancelation, K_1 , K_2 , and K_3 must be adjusted in the analog domain. Similar considerations apply to the 1-2 cascade as well.

DT Cascade We return to the DT implementation of Fig. 24.3(b), assume $C_1 = C_2$ and $C_3 = C_4 = C_5$, and note that $Y = z^{-1}V_{in} + (1 - z^{-1})Q_1$ and $B = z^{-1}Q_1 + (1 - z^{-1})Q_2$. We must then multiply B by $(1 - z^{-1})/z^{-1}$ and subtract the result from Y . Alternatively, we can delay Y by one clock cycle so that it assumes the form $z^{-2}V_{in} + (1 - z^{-1})z^{-1}Q_1$, and subtract $(1 - z^{-1})B = (1 - z^{-1})z^{-1}Q_1 + (1 - z^{-1})^2Q_2$ from it. As expected, Q_2 experiences second-order shaping.

The nominal integrator gains in Fig. 24.3 are given by C_2/C_1 and C_4/C_3 , quantities that are much more accurately set than the absolute RC product in CT realizations. Nevertheless, the unity gain that we have assumed incurs errors due to two mechanisms. First, the op amp yields a closed-loop gain equal to $1/(1 + 2/A_0)$ (Section 23.1.4), requiring that A_0 be sufficiently large. For example, a suppression of 40 dB calls for $A_0 \approx 200$. Second, the incomplete settling of the op amp also translates to gain error, an effect that we have considered tolerable in a single $\Delta\Sigma$ modulator (Section 23.1.6).

Example 24.5

Suppose the first and second integrators in Fig. 24.3 suffer from gain error due to capacitor mismatch, finite op amp gain, and incomplete settling. We thus write their transfer functions as $(1 + \alpha)z^{-1}/(1 - z^{-1})$ and $(1 + \beta)z^{-1}/(1 - z^{-1})$, respectively, where α and β denote the gain errors. Determine the cascade's output Q-noise after combining.

Solution

The first loop provides an input-output transfer function equal to

$$\frac{Y}{V_{in}} = \frac{(1 + \alpha)z^{-1}}{1 - z^{-1}} \quad (24.13)$$

$$= \frac{(1 + \alpha)z^{-1}}{1 + \alpha z^{-1}}. \quad (24.14)$$

The loop also shapes its own Q-noise by a factor of $(1 - z^{-1})/(1 + \alpha z^{-1})$. The outputs of the two stages in Fig. 24.3 can thus be expressed as

$$Y(z) = \frac{(1 + \alpha)z^{-1}}{1 + \alpha z^{-1}} V_{in} + \frac{1 - z^{-1}}{1 + \alpha z^{-1}} Q_1, \quad (24.15)$$

$$B(z) = \frac{(1 + \beta)z^{-1}}{1 + \beta z^{-1}} Q_1 + \frac{1 - z^{-1}}{1 + \beta z^{-1}} Q_2. \quad (24.16)$$

The nominal combining operation multiplies $Y(z)$ by z^{-1} and $B(z)$ by $1 - z^{-1}$, and subtracts the latter result from the former. It follows that

$$W(z) \approx \frac{(1 + \alpha)z^{-2}}{1 + \alpha z^{-1}} V_{in} + \frac{z^{-1}(1 - z^{-1})[(\beta - \alpha)z^{-1} - \beta]}{1 + (\alpha + \beta)z^{-1}} Q_1 - \frac{(1 - z^{-1})^2}{1 + \beta z^{-1}} Q_2. \quad (24.17)$$

Neglecting the $(\alpha + \beta)z^{-1}$ term, we recognize that Q_1 is delayed by one cycle and shaped by $(1 - z^{-1})[(\beta - \alpha)z^{-1} - \beta]$ as it appears in the final output. In other words, the original shaped Q_1 is simply reduced by this factor. In a well-designed cascade, we must ensure that this contribution is much less than $(1 - z^{-1})^2 Q_2$. Note that the first-integrator gain error, α , negligibly affects the performance.

24.1.4 Remedies for Gain Mismatch

The problem of gain mismatch proves serious, especially if we wish to push the speed by employing low-gain, wideband op amps and also exploit linear incomplete settling (Section 23.1.6). Example 24.5 suggests that the first stage's Q-noise is scaled down by a factor equal to the second integrator's gain error. We then surmise that the issue is alleviated if Q_1 is reduced by selecting a *higher-order* loop for the first modulator.

This point brings us to the 2-1 cascade shown in Fig. 24.7, where the first stage is realized as the second-order loop developed in Chapter 21. As a simple case, suppose the first loop incorporates ideal integrators while the second behaves according to the results in the previous section, i.e., it multiplies its input by $(1 + \beta)z^{-1}/(1 + \beta z^{-1})$ and its Q-noise by $(1 - z^{-1})/(1 + \beta z^{-1})$. Thus,

$$Y(z) = z^{-2}X + (1 - z^{-1})^2 Q_1, \quad (24.18)$$

$$B(z) = \frac{(1 + \beta)z^{-1}}{1 + \beta z^{-1}} Q_1 + \frac{1 - z^{-1}}{1 + \beta z^{-1}} Q_2. \quad (24.19)$$

We multiply $Y(z)$ by z^{-1} and $B(z)$ by $(1 - z^{-1})^2$ and subtract the latter result from the former:

$$W(z) \approx z^{-3}X - \beta(1 - z^{-1})^2 z^{-1} Q_1 - (1 - z^{-1})^3 Q_2, \quad (24.20)$$

where βz^{-1} in the denominator is neglected. We observe that the residual Q_1 is shaped by $(1 - z^{-1})^2$ and is therefore less significant.

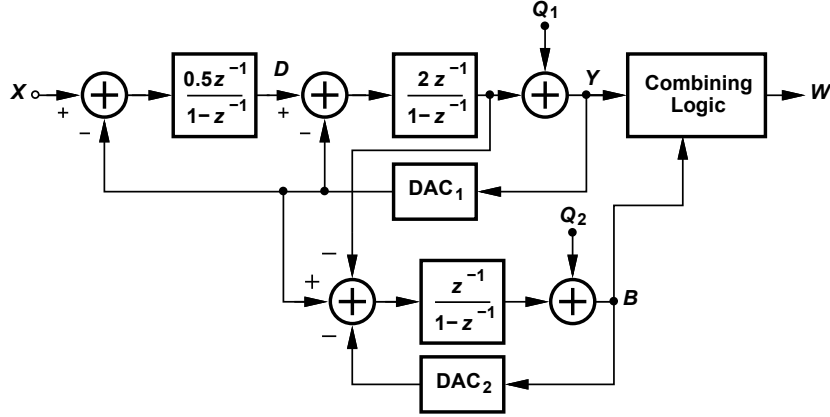


Figure 24.7 Discrete-time 2-1 MASH architecture.

Example 24.6

How do 2-1 and 1-2 cascades compare in view of the gain error problem?

Solution

Let us first study a second-order loop that suffers from gain error. Depicted in Fig. 24.8(a), the model yields

$$\frac{B(z)}{A(z)} = \frac{(1 + \alpha)^2 z^{-2}}{(1 + \alpha z^{-1})^2}, \quad (24.21)$$

where the two integrator gain errors are assumed equal.

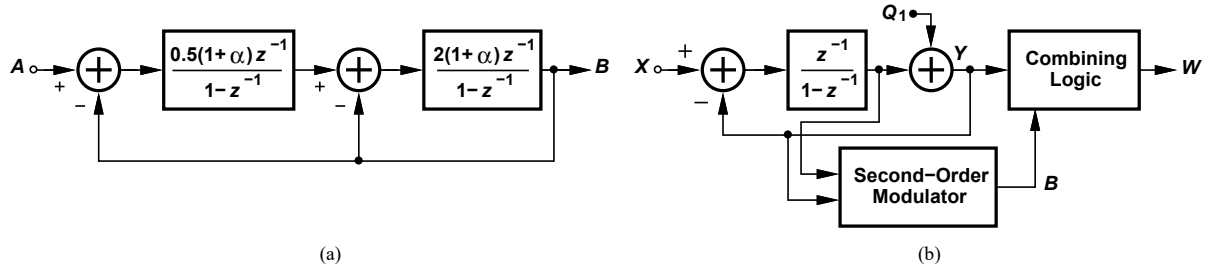


Figure 24.8 (a) Discrete-time modulator with gain error, and (b) use of such a modulator as the second stage of a cascade.

Now, we consider the cascade shown in Fig. 24.8(b) and, assuming no gain error in the first stage, write

$$Y(z) = z^{-1}X + (1 - z^{-1})Q_1, \quad (24.22)$$

$$B(z) = \frac{(1 + \alpha)^2 z^{-2}}{(1 + \alpha z^{-1})^2} Q_1 + H(z)Q_2, \quad (24.23)$$

where $H(z)$ denotes the noise-shaping function of the second-order loop. Upon multiplying $Y(z)$ by z^{-2} and $B(z)$ by $1 - z^{-1}$ and subtracting the results, we obtain

$$W(z) \approx z^{-3}X - 2\alpha(1 - z^{-1})z^{-2}Q_1 - H(z)(1 - z^{-1})Q_2. \quad (24.24)$$

As expected, Q_1 reaches the final output with only first-order shaping, a point of contrast to Eq. (24.20). For this reason, the 2-1 cascade is preferable.

Another approach to reducing the Q-noise of the first stage in a cascade is to raise the resolution of its quantizer. We deal with this concept in the next section.

24.2 $\Delta\Sigma$ Modulators with Multibit Quantizers

Our study of $\Delta\Sigma$ modulators has thus far assumed a 1-bit quantizer. We have sought to increase the resolution by increasing the order or the oversampling ratio of a single loop or by cascading, but we have not asked whether we can simply employ a multibit (e.g., flash) quantizer. Since the one-sided Q-noise spectral density is given by $\Delta^2/(12f_{CK})$ for a sampling rate of f_{CK} , we expect a proportional rise in the SNR as the quantizer resolution is raised and hence Δ is reduced.

24.2.1 Basic Idea

Figure 24.9(a) shows a simple realization of the idea. The integrator is followed by a flash stage, and the resulting thermometer code directly drives a multibit thermometric DAC. The flash quantizer output carries the digitized signal along with shaped Q-noise. We should make two remarks. First, the flash stage's full-scale (reference) voltage, V_{FS} , must be chosen no less than the maximum swing produced by the integrator. This generally requires careful simulations with various input signal waveforms. Second, the flash thermometer code can be converted to a binary output, which must then be processed to create Nyquist-rate data.

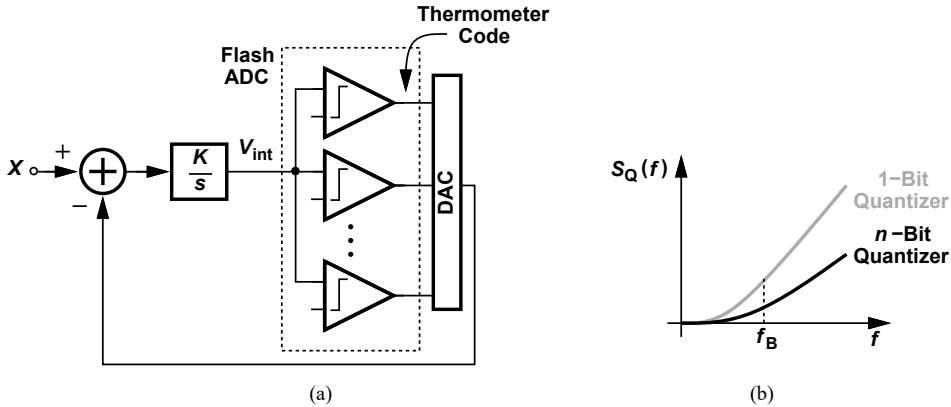


Figure 24.9 (a) First-order loop with multibit quantizer, and (b) quantization noise behavior for 1-bit and multibit quantizers.

Let us examine the shaped Q-noise in Fig. 24.9(a). We know that the loop gives $Y/Q = s/(s + K)$, where Y denotes the thermometer value and Q the noise introduced by the flash quantizer. Our treatment in Chapter 21 has approximated $|Y/Q|^2$ by $\omega^2 = 4\pi^2 f^2$ and multiplied it by $\Delta^2/(12f_{CK})$, yielding $S_Q = \pi^2 \Delta^2 f^2 / (3f_{CK})$. The same results apply here as well, except that Δ is now equal to $V_{FS}/2^n$, where n denotes the flash stage's resolution (the number of bits) [Fig. 24.9(b)]. If V_{FS} is approximately equal to the main input range, then S_Q falls by a factor of 2^{2n} , and so does the Q-noise integrated across the signal bandwidth, f_B .

In addition to a direct rise in the SNR, multibit quantizers offer several other advantages as well. First, since the DAC in Fig. 24.9(a) delivers smaller steps to the integrator, the latter settles faster *and* generates *smaller* output swings. (Recall that V_{int} includes both V_{in} and the Q-noise.) The op amp design is therefore relaxed. Second, the quantizer becomes more linear, improving the loop stability. To understand this point, we note from Chapter 21 that the gain of a 1-bit quantizer falls in proportion to its input amplitude [Fig. 24.10(a)]. With a higher resolution, on the other hand, we can assume that the gain is relatively constant across the input range [Fig. 24.10(b)]. We state without proof that this gain behavior makes the modulator more stable.

The third advantage manifests itself in continuous-time modulators: the smaller DAC output steps reduce the effect of clock jitter proportionally. From Section 23.4.4, we note that, if the DAC output current is equal

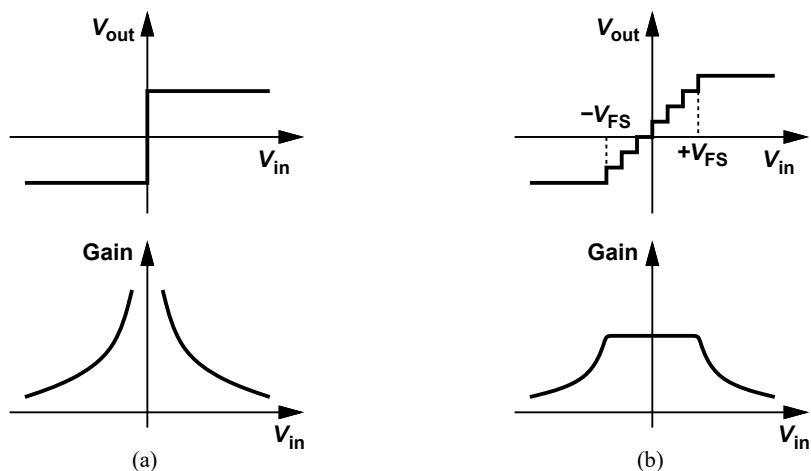


Figure 24.10 Input-output and gain characteristics of (a) 1-bit, and (b) multibit quantizers.

to $I_{FS}/2^n$, where I_{FS} is the full-scale value, then a clock jitter of Δt_j corrupts the charge delivered to the integrator by $(I_{FS}/2^n)\Delta t_j$.

The fourth advantage of multibit quantization relates to cascaded modulators: if the first stage employs such an approach, then the Q-noise leakage due to gain errors is smaller.

We thus observe that a flash quantizer having even a resolution of 3 or 4 bits can dramatically improve the performance of $\Delta\Sigma$ modulators. Unfortunately, however, the actual design is not straightforward. Of course, the flash stage presents a load to the integrator, degrading its settling speed. But, more importantly, the multibit DAC in Fig. 24.9(a) proves problematic, as explained below.

24.2.2 Problem of DAC Nonlinearity

In our analysis of previous $\Delta\Sigma$ modulators, we were not concerned with DAC nonlinearity because a 1-bit realization is perfectly linear. For example, a DAC delivering $V_{REF}^+ = 0.9$ V and $V_{REF}^- = 0.2$ V to the integrator(s) introduces no nonlinearity even if these voltages depart from their nominal values. This is because, with only two output levels, we can always pass a straight line through them [Fig. 24.11(a)]. With more than two, however, this is not possible unless the DAC exhibits sufficient matching among its constituent components [Fig. 24.11(b)].

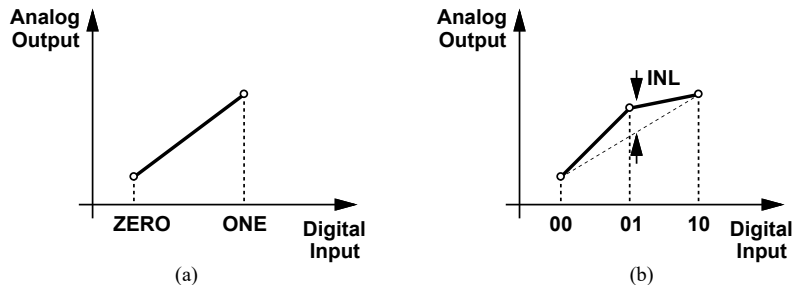


Figure 24.11 Input-output characteristics of (a) 1-bit and (b) multibit DACs.

Example 24.7

How can we express the output of a nonlinear DAC?

Solution

We can assume that each output level incurs a certain departure from its ideal value. For example, suppose a

resistor-ladder DAC is tied between zero and V_{REF} and provides 2^n levels. Ideally, we have $V_j = jV_{REF}/2^n$, but in the presence of resistor mismatches, we write $V_j = (jV_{REF}/2^n) + \Delta V_j$, where ΔV_j denotes the error. This “additive” view of nonlinearity proves useful below.

Let us now determine how DAC nonlinearity affects the performance of $\Delta\Sigma$ modulators. Considering the simple loop depicted in Fig. 24.12 and assuming that the DAC’s errors can be represented by ΔV_j , we observe that the subtractor produces $V_{in} - jV_{REF}/2^n - \Delta V_j = (V_{in} - \Delta V_j) - jV_{REF}/2^n$. Thus, DAC nonlinearity is simply referred to the main input, i.e., it equivalently distorts V_{in} .

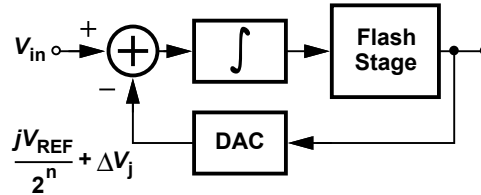


Figure 24.12 Representation of nonlinearity at DAC output.

The key point here is that, even though the DAC has a *resolution* of 3 or 4 bits, its *linearity* must be commensurate with the overall ADC’s performance. For example, a 14-bit design requires that ΔV_j in Fig. 24.12 be less than $2^{-14} = 0.006\%$ of the input full scale.

Another critical implication of DAC nonlinearity is the folding of high-frequency quantization noise. Recall from Section 23.3 that nonlinearity mixes high-frequency components, thereby raising the noise within the signal bandwidth. Thus, as the quantizer’s noise in Fig. 24.12 travels through the DAC, its high-frequency components fold and then subtract from V_{in} (Fig. 24.13). For these reasons, a multibit quantizer/DAC chain demands that the DAC nonlinearity be corrected. One approach is based on “data-weighted averaging” (DWA).

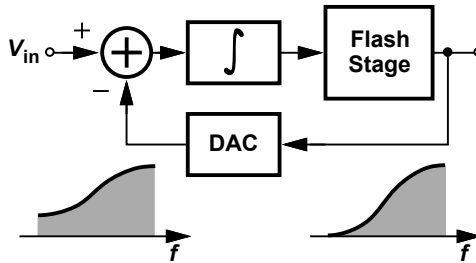


Figure 24.13 Q-noise folding due to DAC nonlinearity.

24.2.3 Data-Weighted Averaging

Suppose the feedback DAC in a $\Delta\Sigma$ modulator is directly driven by an n -bit flash stage. The DAC consists of 2^n or $2^n - 1$ units (capacitors, resistors, or current sources) and they incur mismatches. For illustration purposes, we assume a current-mode topology.

We begin with a three-level DAC. Shown in Fig. 24.14(a), the circuit employs two nominally identical current sources. For a mismatch of, say, $2\Delta I$, it is simpler to represent I_1 and I_2 by $I_0 + \Delta I$ and $I_0 - \Delta I$, respectively.¹ We recognize two possible code assignments here: as the thermometer code rises from 00 to 01 to 11, we can turn on I_1 first or I_2 first. As depicted in Fig. 24.14(b), the first scenario yields a *negative* INL,

¹We denote the mismatch by $2\Delta I$ rather than ΔI for the sake of simplicity.

and the second a *positive* INL.² We then surmise that these two errors “average out” if the code assignment changes over time: when the quantizer generates an output equal to 01, I_{out} is chosen to be $I_0 - \Delta I$ in some cycles and equal to $I_0 + \Delta I$ in some other cycles [1].

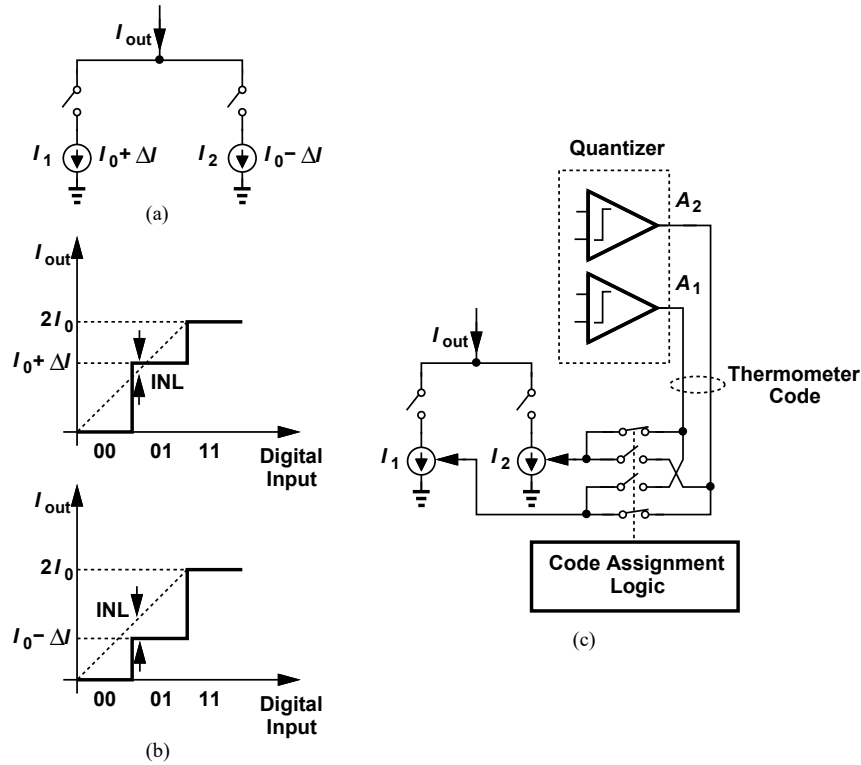


Figure 24.14 (a) A three-level DAC, (b) its behavior for two different code assignments, and (c) error randomization network.

Figure 24.14(c) illustrates the implementation of the idea. If $A_2A_1 = 00$, both current sources are off, and if $A_2A_1 = 11$, both are on. That is, the code assignments are unique. However, for $A_2A_1 = 01$, we swap A_1 and A_2 occasionally. We now decide how often the swapping must occur. To avoid accumulation of ΔI_0 , it is best to swap A_1 and A_2 every time the quantizer produces $A_2A_1 = 01$, hence the code assignment logic must “remember” the last assignment of $A_2A_1 = 01$ to I_1 and I_2 . This method is called “data-weighted averaging” (DWA) [1].

Mismatch Shaping Data-weighted averaging in essence “shapes” the spectrum of the DAC mismatch, moving it to high frequencies. To appreciate this point, suppose the flash quantizer in Fig. 24.14(c) senses a constant input and is clocked at a rate of f_{CK} while producing $A_2A_1 = 01$. The code assignment logic turns on I_1 in one cycle and I_2 in the next. The DAC output current thus behaves as shown in Fig. 24.15(a), containing an average equal to I_0 and a square wave toggling at $f_{CK}/2$. Writing I_{out} as a Fourier series, we have

$$I_{out} = I_0 + \frac{4}{\pi} \Delta I \cos(\pi f_{CK} t) - \frac{4}{3\pi} \Delta I \cos(3\pi f_{CK} t) + \dots \quad (24.25)$$

Depicted in Fig. 24.15(b) is the resulting spectrum. We say the mismatch, ΔI , is “shaped” because it appears only at high frequencies.

²Recall that the INL is given by the straight line passing through the end points minus the actual characteristic.

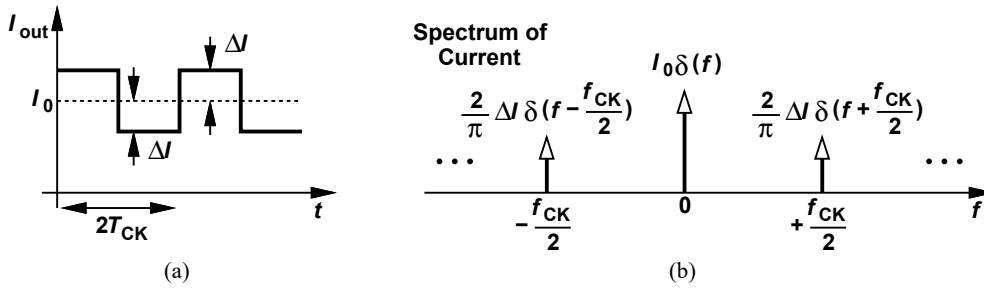


Figure 24.15 (a) DAC output current, and (b) resulting spectrum.

Extension to Higher Resolutions In order to extend the DWA concept to a greater number of DAC levels, we rephrase the strategy as follows: each new thermometer code selects an *unused* DAC unit [1]. This is indeed the case in Fig. 24.14(c) if $A_2A_1 = 01$ occurs in two consecutive cycles: in one cycle, I_1 is used and in the next, I_2 . Consider a DAC employing three nominally equal current sources, I_1 , I_2 , and I_3 , to generate four levels. We wish to assign the codes to these current sources so as to minimize mismatch accumulation.

Let us assume a nominal value of I_0 for each current source. Figure 24.16(a) demonstrates that the nominal output level I_0 (Level 2) can be established by any of the three current sources. Similarly, $2I_0$ (Level 3) can be generated by selecting two out of the three. The two ends of the characteristic, however, are unique and equal to zero and $I_1 + I_2 + I_3$. As explained in Chapter 8, we pass a line through these points and obtain the INL profile by subtracting the actual characteristic from it [Fig. 24.16(b)]. We recognize that (a) the departure of $I_1 + I_2 + I_3$ from $3I_0$ leads to gain error rather than INL (Chapter 8), and (b) only Level 2 and Level 3 can contribute nonlinearity.

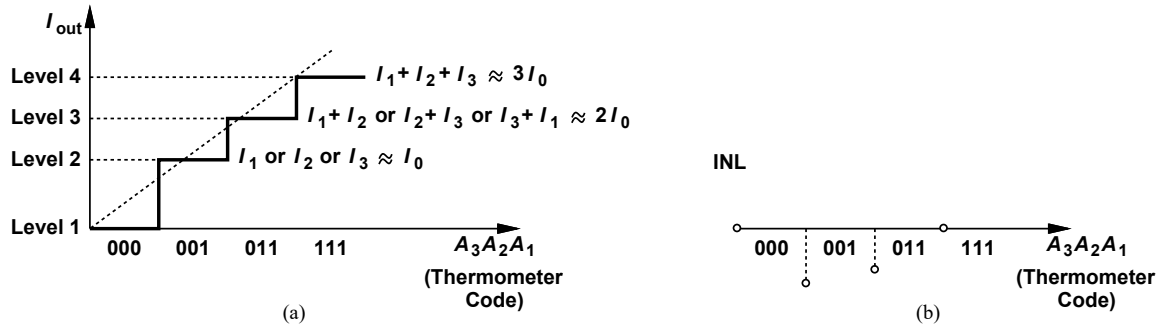


Figure 24.16 (a) Four-level DAC characteristic, and (b) its INL profile.

Now, suppose current source I_j incurs an error, ΔI_j . From Fig. 24.17(a), the straight line is expressed as

$$I_{out}(k) = \frac{3I_0 + \Delta I_1 + \Delta I_2 + \Delta I_3}{3} k, \quad (24.26)$$

where k denotes the digital input. If the DAC were linear, then $k = 1$ ($A_3A_2A_1 = 001$) would yield the “ideal” point P :

$$I_{out}(k = 1) = I_0 + \frac{\Delta I_1 + \Delta I_2 + \Delta I_3}{3}. \quad (24.27)$$

But the actual current can be equal to $I_1 = I_0 + \Delta I_1$, $I_2 = I_0 + \Delta I_2$, or $I_3 = I_0 + \Delta I_3$. Equation (24.27) reveals that the ideal output can be obtained by averaging I_1 , I_2 , and I_3 . That is, if the digital input is constant and equal to $A_3A_2A_1 = 001$, we select I_1 in one clock cycle, I_2 in the next, I_3 in the one after [Fig. 24.17(b)]. This agrees with the principle that “each new code selects an *unused* unit.” We now plot the difference between these levels and the ideal values given by Eq. (24.26) [Fig. 24.17(c)], noting that they add

up to zero. This waveform thus exhibits a zero average between t_a and t_b , repeats every three clock cycles, and hence displays the spectrum shown in Fig. 24.17(d). We say the current mismatches are “shaped” and moved to high frequencies.

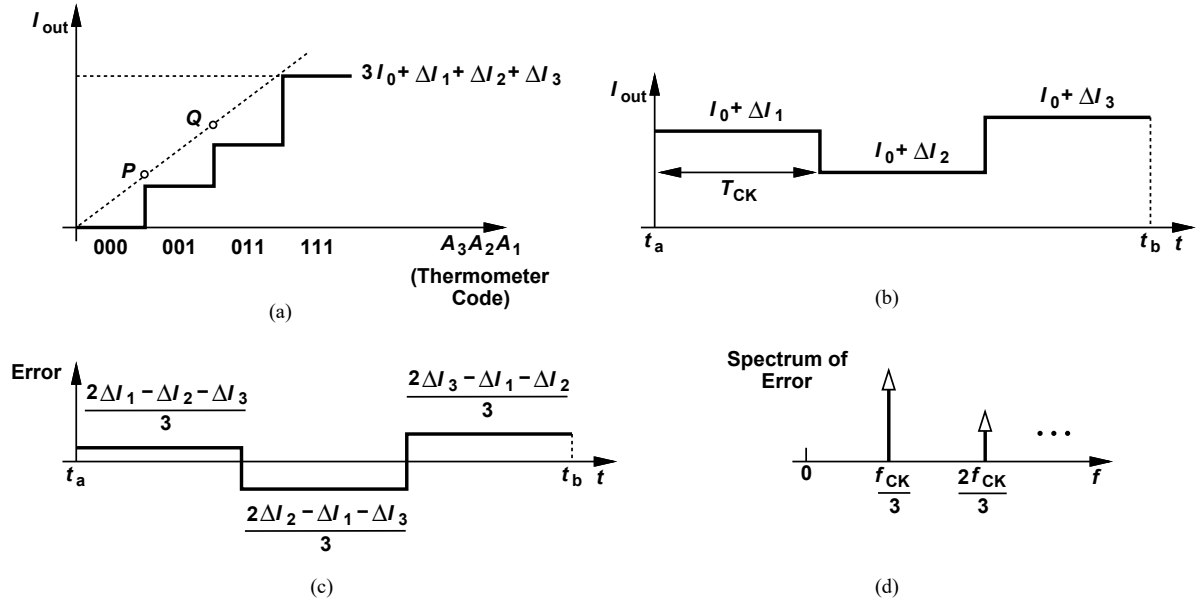


Figure 24.17 (a) DAC characteristic with nonlinearity, (b) current switching with randomization, (c) resulting error waveform, and (d) corresponding error spectrum.

Example 24.8

Repeat the foregoing analysis if the DAC input is constant and equal to $k = 2$ ($A_3A_2A_1 = 011$).

Solution

The ideal point Q in Fig. 24.17(a) is characterized by

$$I_{out}(k = 2) = 2I_0 + \frac{2\Delta I_1 + 2\Delta I_2 + 2\Delta I_3}{3}. \quad (24.28)$$

In three consecutive clock cycles, we select $I_1 + I_2 = 2I_0 + \Delta I_1 + \Delta I_2$, $I_2 + I_3 = 2I_0 + \Delta I_2 + \Delta I_3$, and $I_3 + I_1 = 2I_0 + \Delta I_3 + \Delta I_1$ [Fig. 24.18(a)], recognizing that their average is equal to that given by Eq. (24.28). The error varies as depicted in Fig. 24.18(b) and exhibits a zero average. Its spectrum is then similar to that in Fig. 24.17(d).

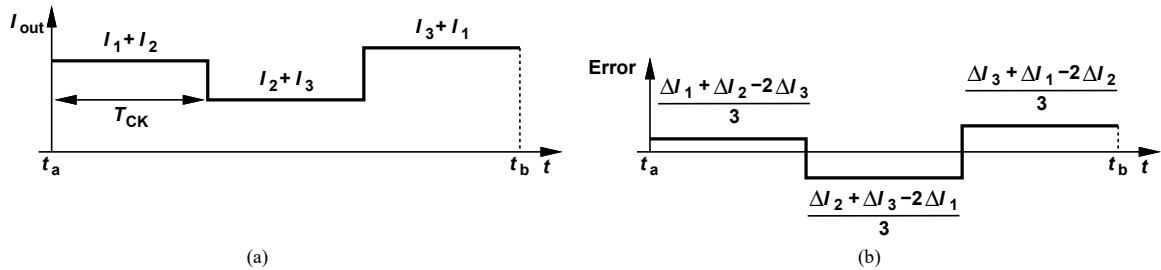


Figure 24.18 (a) DAC output behavior for $k = 2$, and (b) resulting error waveform.

Switching Sequence for Random Inputs Let us ponder how the codes are assigned to the DAC units as the digital input varies randomly. This is more easily visualized by means of a “rotation” diagram [2, 3]. We consider a DAC employing eight unit current sources, I_1 - I_8 , and arrange them in a circle (Fig. 24.19). Suppose the decimal equivalent of the digital input over four clock cycles is equal to 2, 4, 3, and 6. For the first digital input, we begin with I_1 and select the proper number of current sources, e.g., I_1 and I_2 . In the next cycle, to ensure the principle that “each new code selects an unused unit,” we enable the next four units, I_3 - I_6 . Similarly, in the third cycle, we activate I_7 , I_8 , and I_1 , and in the fourth, we enable I_2 - I_7 . In this spirit, we can call this method “rotational element selection” [3]. The key point here is that the rotation diagram provides a clear and unique procedure for selecting the DAC units in consecutive cycles.

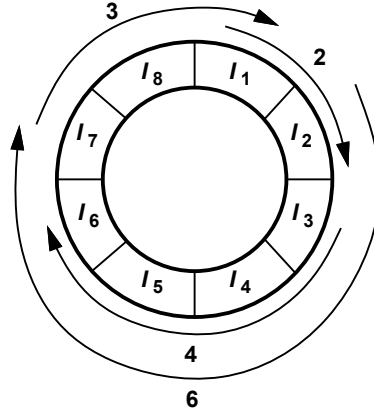


Figure 24.19 Illustration of current source selection in consecutive clock cycles.

DWA Issues DWA reduces the accumulation of mismatches over time, thus pushing their effect to high frequencies so that they can be removed along with Q-noise. DWA, however, faces two difficulties. First, it produces tones at low frequencies, even within the signal band [1]. A number of techniques have been devised to deal with this issue [2, 4]. Second, if the logic necessary for DWA appears between the multibit quantizer and the DAC, it inevitably introduces excess delay and degrades the loop stability. This proves particularly serious in high-speed continuous-time $\Delta\Sigma$ modulators (Chapter 23). The implementation of the logic for minimal loop delay is described in [1, 3].

24.3 Modulators with Feedforward

We have seen in Chapter 22 that the integrators in a $\Delta\Sigma$ loop experience large output swings, thereby challenging their op amps. The use of feedforward can alleviate this issue. It can also improve the loop stability.

Consider the second-order system shown in Fig. 24.20, where two memoryless feedforward paths allow the input to travel to the outputs of the integrators. These paths' gains, α and β , are so chosen as to yield a proper signal transfer function. We note that the first integrator output is equal to $(X - Y)z^{-1}/(1 - z^{-1})$, to which we must add $\alpha X - 2Y$. The result is then multiplied by $z^{-1}/(1 - z^{-1})$ and summed with $\beta X + Q$ to yield Y . It follows that

$$Y(z) = [(1 - \alpha + \beta)z^{-2} + (\alpha - 2\beta)z^{-1} + \beta]X(z) + (1 - z^{-1})^2Q(z). \quad (24.29)$$

Interestingly, the feedforward paths do not alter the noise transfer function. This is because, to find Y/Q , we can set X to zero and obtain $\alpha X = 0$ and $\beta X = 0$, i.e., the feedforward action vanishes. For $X(z)$ to experience a flat response, we set the coefficients of z^{-2} and z^{-1} to zero, obtaining $\alpha = 2$ and $\beta = 1$. Thus,

$$Y(z) = X(z) + (1 - z^{-1})^2Q(z). \quad (24.30)$$

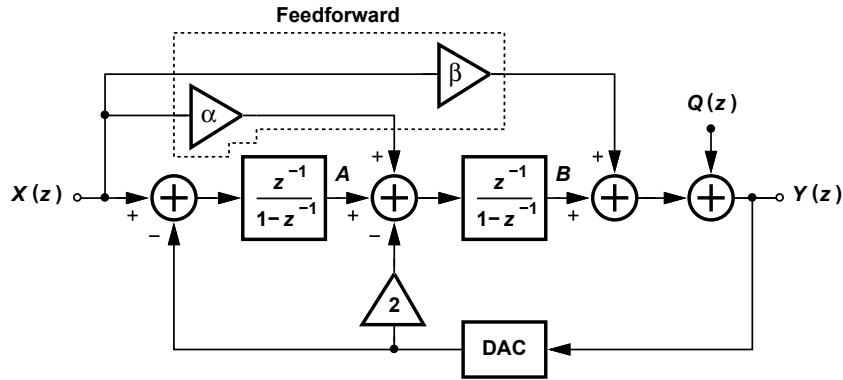


Figure 24.20 $\Delta\Sigma$ modulator with feedforward paths.

We must now examine the outputs of the two integrators in Fig. 24.20. We write

$$A(z) = (X - Y) \frac{z^{-1}}{1 - z^{-1}} \quad (24.31)$$

$$= -(1 - z^{-1})z^{-1}Q(z). \quad (24.32)$$

Thus, $A(z)$ does not contain the *input* signal and hence exhibits smaller voltage swings. For $B(z)$, we have $B(z) + \beta X(z) + Q(z) = Y(z)$, which from Eq. (24.30) yields

$$B(z) = -z^{-1}(2 - z^{-1})Q(z). \quad (24.33)$$

This output, too, is free from X . We conclude that feedforward elegantly reduces the integrator swings.

Example 24.9

Consider the summation with αX in Fig. 24.20. How is this implemented in a switched-capacitor environment?

Solution

Figure 24.21 depicts a possible approach where summation is realized within the second integrator. While C_1 captures the first integrator output, C_3 senses X . They are then both switched into the second integrator; hence their charges travel through the virtual ground and are summed on C_2 .

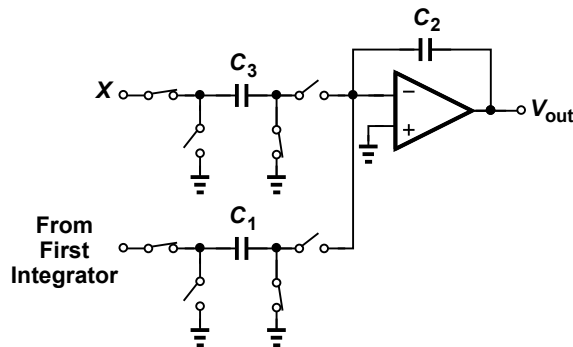


Figure 24.21 Summer implementation in feedforward path.

The principal difficulty with feedforward is the need for summation at the *last* integrator output, i.e., at the quantizer input. Since no virtual ground is present at this interface, summation poses a challenge, especially because linearity is still somewhat critical before the result reaches the quantizer. For this reason, we must employ an active summer, as illustrated in Fig. 24.22. Here, S_F is on while X and the last integrator outputs are sampled, and turns off before the left plates of C_a and C_b swing to zero.³ The additional op amp consumes more power. The summer also degrades the loop stability by introducing its own delay.

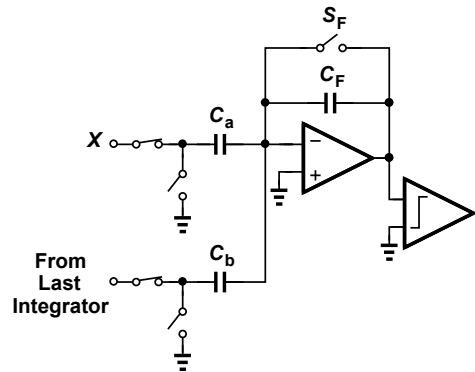


Figure 24.22 Active summer necessary before the quantizer.

24.4 Decimation Filtering

24.4.1 Basic Idea

The oversampled output of $\Delta\Sigma$ modulators must be processed in the digital domain for two purposes. First, the noise outside the signal bandwidth must be removed (Fig. 24.23). For example, a third-order $\Delta\Sigma$ modulator may require a fourth-order filter so as to suppress the Q-noise. Second, the high-speed output must be converted to Nyquist-rate data, an operation called “downsampling.” Both functions are realized by a “decimation” filter. We assume an oversampling ratio of M and a 1-bit quantizer.

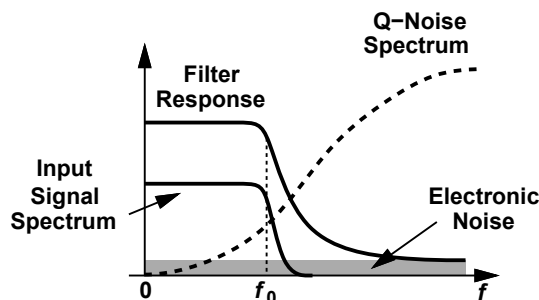


Figure 24.23 Filtering required after oversampling ADC.

We postulated in Chapter 23 that averaging and downsampling can be carried out by simply replacing every M consecutive samples by their average (Fig. 24.24). In fact, if normalization to M is disregarded, the processing can be viewed as counting the number of ONEs among every M output bits. For example, $M = 64$ yields an output ranging from zero to 64. But where does the *resolution* come into the picture? We observe that $M = 64$ provides no more than 6 bits of resolution regardless of the order of the $\Delta\Sigma$ modulator! To answer this question, we must modify our approach.

³The circuit is similar to the MDAC studied in Chapter 17. It is *not* an integrator.

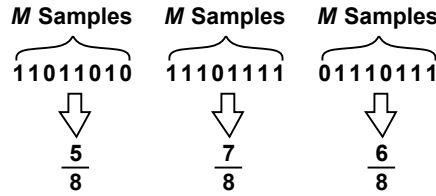


Figure 24.24 Simple filtering and decimation by averaging groups of bits.

Let us still perform averaging but without downsampling. That is, we average samples 1 through M , then samples 2 through $M + 1$, etc. [Fig. 24.25(a)]. Called the “running” average, this quantity can be formed by summing the delayed replicas of the digital signal [Fig. 24.25(b)]. We have

$$B(z) = Y(z)[1 + z^{-1} + \dots + z^{-(M-1)}]. \quad (24.34)$$

This structure exemplifies a finite impulse response (FIR) filter. Note that the filter operates at a rate equal to the oversampling clock frequency, potentially consuming high power.

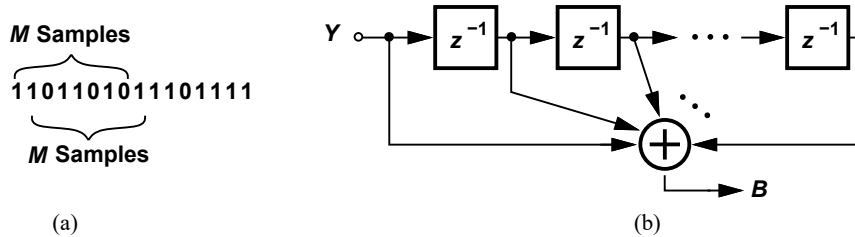


Figure 24.25 (a) Filtering by a running average, and (b) its implementation.

It is possible to simplify the logic by noting that

$$[1 + z^{-1} + \dots + z^{-(M-1)}](1 - z^{-1}) = 1 + z^{-1} + \dots + z^{-(M-1)} - z^{-1} - \dots - z^{-M} \quad (24.35)$$

$$= 1 - z^{-M}. \quad (24.36)$$

That is,

$$H(z) = 1 + z^{-1} + \dots + z^{-(M-1)} = \frac{1 - z^{-M}}{1 - z^{-1}}. \quad (24.37)$$

It can be shown that the magnitude of this transfer function has a sinc shape (Fig. 24.26) and hence the arrangement in Fig. 24.25(b) is called a first-order “sinc filter.”

Equation (24.37) suggests an alternative implementation: we simply cascade an integrator providing $1/(1 - z^{-1})$ and a delay-and-subtract circuit realizing $1 - z^{-M}$ [Fig. 24.27(a)]. Employing M delay stages, the latter is also called a “comb filter” as its magnitude response exhibits peaks at odd integer multiples of $1/(MT_{CK})$ [Fig. 24.27(b)].⁴ We can also view this circuit as a differentiator.

The principal drawback of the first-order filter is the gradual rolloff of its sinc magnitude and hence insufficient suppression of the quantization noise. That is, the area under $S_Q(f)|H(z)|^2$ across the signal BW in Fig. 24.26 is still rather large. We thus raise the filter’s order as follows:

$$H(z) = \left(\frac{1 - z^{-M}}{1 - z^{-1}} \right)^k. \quad (24.38)$$

⁴This can be proved by writing $z = e^{j2\pi f T_{CK}}$ and taking the magnitude of $1 - z^{-M}$.

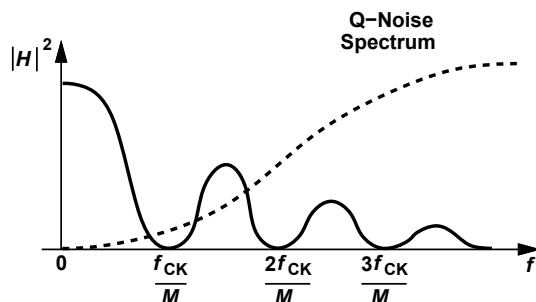


Figure 24.26 Sinc filter response.

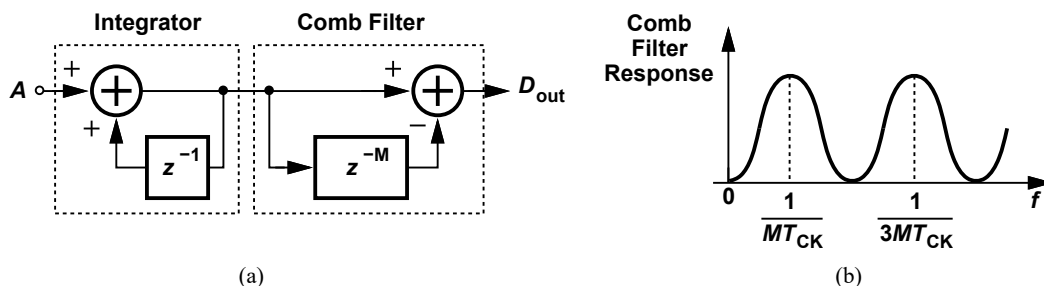


Figure 24.27 (a) Filter realized as a cascade of an integrator and a comb filter, and (b) comb filter response.

The result is called a “cascaded-integrator-comb” (CIC) filter and can be realized as shown in Fig. 24.28. Note that the output is still at the oversampled rate. We then envision the arrangement shown in Fig. 24.29 as

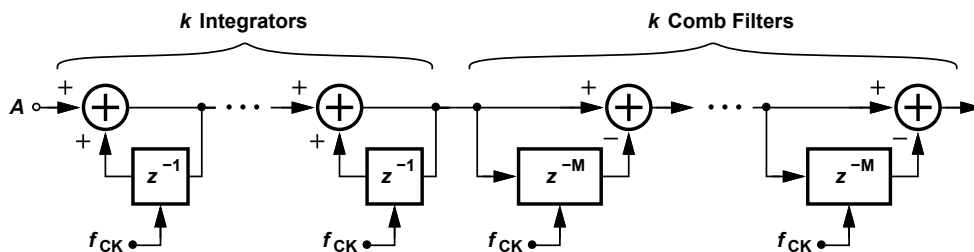


Figure 24.28 High-order filter consisting of cascaded integrators and comb filters.

the complete solution: the $\Delta\Sigma$ modulator drives a CIC filter, whose output is “downsampled” (“decimated”) by a factor of M so as to produce Nyquist-rate data. The downsampler simply passes only one out of every M samples produced by the filter.

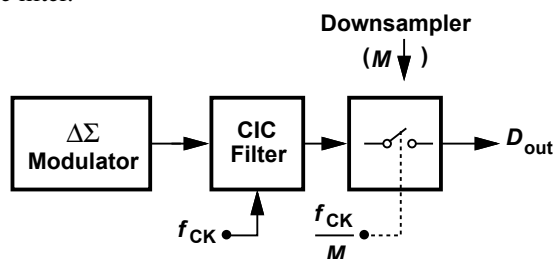


Figure 24.29 $\Delta\Sigma$ modulator followed by a filter and a downsampler.

Example 24.10

A student decides to swap the filter and the downsampler in Fig. 24.29 [Fig. 24.30(a)], reasoning that the former consumes more power and becomes more efficient if clocked at a lower frequency. Can this be done?

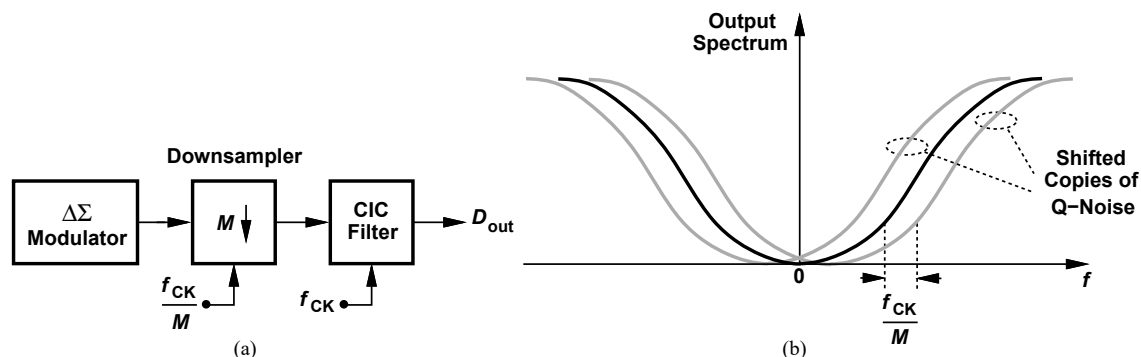


Figure 24.30 (a) $\Delta\Sigma$ modulator followed by a downsampler and a filter, and (b) resulting aliasing of Q-noise.

Solution

No, it cannot. The downsampler now *directly* samples the $\Delta\Sigma$ modulator output at a rate of f_{CK}/M whereas the Q-noise spectrum at this interface reaches a peak at $f_{CK}/2$. Thus, significant aliasing occurs and the Q-noise folds down to the signal band [Fig. 24.30(b)].

Example 24.11

The student in the foregoing example attempts another variant: how about downsampling between the integrators and the differentiators?

Solution

Illustrated in Fig. 24.31, this solution proves promising because the high-frequency Q-noise is suppressed by the cascade of integrators. Downsampling thus experiences negligible aliasing. This popular architecture saves considerable power in the flipflops and the adder in each comb filter.

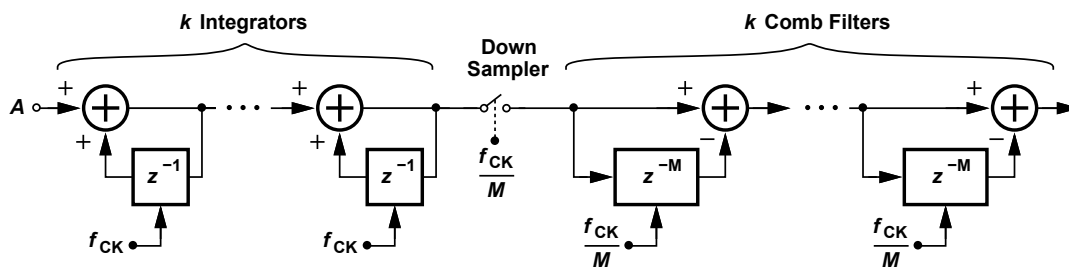


Figure 24.31 Cascaded full-rate integrators followed by downsampling and slower comb filters.

While suppressing the Q-noise by a greater factor, a high-order CIC filter also exacerbates the in-band sinc “droop,” removing some of the *signal* energy (Fig. 24.32). For this reason, the order of the filter, k , must be chosen carefully.

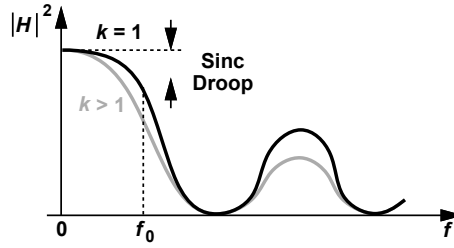


Figure 24.32 Sinc droop at the edge of the signal band.

24.4.2 Counter-Based Design

In addition to FIR and CIC implementations, one can opt for a counter-based sinc filter structure [3], its advantage being that it does not require a digital adder in some cases. In its simplest form, however, the topology operates with only a 1-bit input, i.e., it is suited to $\Delta\Sigma$ loops employing 1-bit quantizers. The circuit is realized as depicted in Fig. 24.33(a): a counter begins from zero and counts the number of ONEs for MT_{CK} seconds [Fig. 24.33(b)], with the result captured by the register. The counter and the register are then reset. This is called an “accumulate-and-dump” operation because the counter accumulates (integrates) the input and is reset after it delivers the data to the register. If, for example, $M = 64$, then the counter and the register produce 6-bit binary outputs. In our mind, we normalize this value to M to find the average.

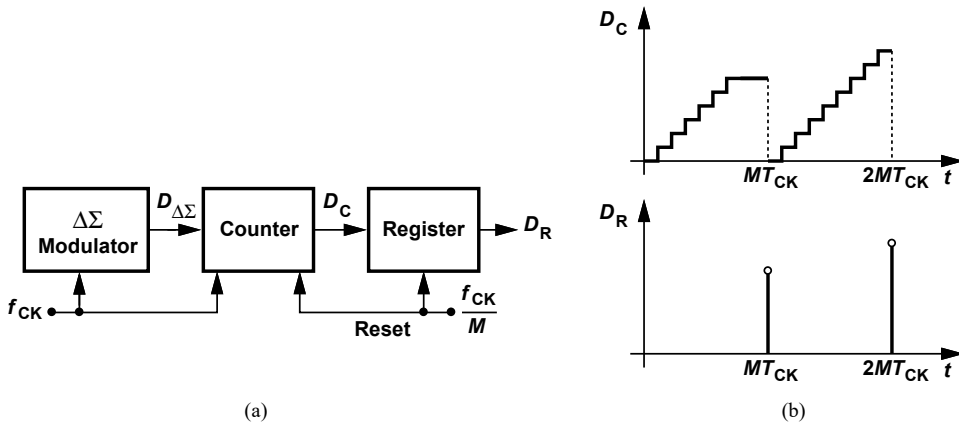


Figure 24.33 (a) Counter-based filter, and (b) counter and register outputs.

We must now ask what type of transfer function the foregoing circuit implements. After all, we must still *filter* the noise accompanying the signal. Let us study the analog counterpart for simplicity. Consider the “integrate-and-dump” circuit shown in Fig. 24.34(a), where the input, I_{in} , is integrated by C_1 for T_1 seconds, and the output is then reset. We wish to determine the circuit’s frequency response. We assume $I_{in} = I_0\delta(t)$, noting that the first integration interval produces a pulse [Fig. 24.34(b)], but, upon reset, V_{out} remains equal to zero after $t = T_1$. The magnitude response is given by the Fourier transform of the impulse response (why?) and is thus a sinc function.

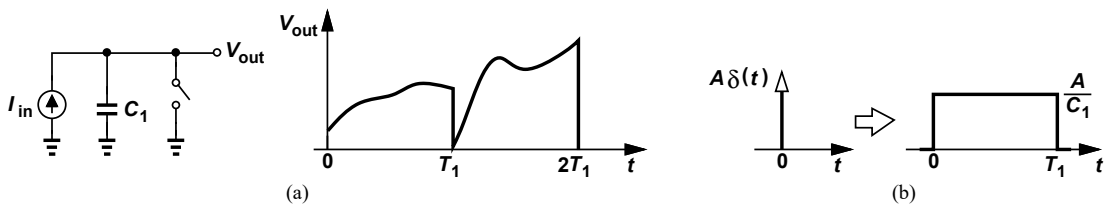


Figure 24.34 (a) An analog integrate-and-dump circuit, and (b) its impulse response.

References

- [1]. R. Biard and T. Fiez, "Linearity enhancement of multibit $\Delta\Sigma$ A/D and D/A converters using data weighted averaging," *IEEE Transactions on Circuits and Systems II*, vol. 42, pp. 753-762, Dec. 1995.
- [2]. K. Vleugels, S. Rabii, and B. A. Wooley, "A 2.5-V sigma-delta modulator for broadband communications applications," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 1887-1889, Dec. 2001.
- [3]. S. Pavan, R. Schreier, and G. C. Temes, *Understanding Delta-Sigma Converters*, Second Ed., Wiley, 2017.
- [4]. I. Fujimori, A. Nogi, and T. Sugimoto, "A multibit delta-sigma audio DAC with 120-dB dynamic range," *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 1066-1073, Aug. 2000.

Problems

- 24.1. Can we reconstruct the quantization noise in Fig. 24.1(b) by combining the subtractor output with another signal in the loop?
- 24.2. We have noted that proper combining of Y and Q in Fig. 24.2 requires that B be differentiated. Is the derivative of B present anywhere within the error digitizer?
- 24.3. Our analysis of Fig. 24.2 indicates that, at low frequencies, $Y/X \approx 1$ and $Y/Q_2 \approx -s^2/(K_2K_1)$. Compare these results to those of a second-order CT $\Delta\Sigma$ modulator.
- 24.4. Consider the 1-1 cascade shown in Fig. 24.3(b), assuming $V_{in} = 0$. Calculate the kT/C noise at the output of the first and second integrators. Assume $V_{DAC1} = V_{DAC2} = 0$.
- 24.5. Suppose the noise-shaping function obtained for Fig. 24.4 exhibits two real poles at ω_{p1} and ω_{p2} . Sketch the magnitude of the NTF. Do the poles increase or decrease the Q-noise? Can we reduce $|\omega_{p1}|$ and $|\omega_{p2}|$ arbitrarily?
- 24.6. Sketch the magnitude of the NTF for the 1-2 cascade shown in Fig. 24.5 and compare it to that obtained in the previous problem.
- 24.7. We wish to implement a discrete-time version of the 2-1 cascade shown in Fig. 24.4. Determine how we select delaying or nondelaying integrators. Next, compute $Y(z)$ and $B(z)$ and explain the functions that must be realized by the combining logic.
- 24.8. Repeat the previous problem for the 1-2 cascade shown in Fig. 24.5.
- 24.9. Suppose the differentiator in Fig. 24.6 provides a transfer function of the form $1 - z^{-1} \approx sT_{CK}$ whereas the second integrator in Fig. 24.2 has a gain, K_2 , that does not quite match $1/T_{CK}$. That is, $K_2 = (1/T_{CK}) + \epsilon$. Compute $Y(s) - (s/K_2)B(s)$ for Fig. 24.2.
- 24.10. If the second integrator in Fig. 24.7 suffers from gain error, we can express its transfer function as $2(1 + \alpha)z^{-1}/(1 - z^{-1})$. Determine $Y(z)$ and $B(z)$. Next, compute the combining logic output, i.e., $z^{-1}Y(z) - (1 - z^{-1})^2B(z)$.
- 24.11. Repeat the previous problem if only the first integrator incurs a gain error.
- 24.12. From the conceptual diagram in Fig. 24.9, construct a second-order $\Delta\Sigma$ modulator with an n -bit quantizer. Estimate the noise-shaping function.
- 24.13. Repeat the study in Fig. 24.15 if the flash quantizer in Fig. 24.14(c) senses a constant input and generates $A_2A_1 = 10$.
- 24.14. In a DAC employing three current sources, we have $I_1 = I_2$ but $I_3 \neq I_1$. Repeat the analysis depicted in Figs. 24.16 and 24.17.
- 24.15. A DAC employs four current sources, $I_1 = I_0 + \Delta I_1$, $I_2 = I_0 + \Delta I_2$, etc. Using a rotation diagram, determine how the current sources are selected as the digital input varies in this sequence: 1, 3, 2, 4, 2, 1. Sketch the output current and the error versus time.
- 24.16. Determine the input of the integrator in Fig. 24.20 in terms of X and Q . Simplify the result if $\alpha = 2$ and $\beta = 1$.
- 24.17. Repeat the previous problem for the output of the multiply-by-2 stage (which appears between the DAC and the second subtractor).
- 24.18. Suppose we replace the multiply-by-2 stage in the feedback path of Fig. 24.20 with a unity gain and also change the first integrator to a delaying topology. Is it possible to select α and β so as to obtain an acceptable signal transfer function? Assume $\alpha, \beta \neq 0$.
- 24.19. The active summer depicted in Fig. 24.22 is used in Fig. 24.20. Determine the summer output in terms of X and Q (in the z -domain).
- 24.20. Sketch the magnitude of the transfer function of Fig. 24.28.

Index

- 1-Bit quantizer properties, 505
- 1.5-bit-per-stage MDAC, 425

- Acquisition time, 53
- ADC, 1
- ADC DNL, 242
- ADC ENOB, 247
- ADC figure of merit, 248
- ADC gain error, 241
- ADC INL, 243
- ADC offset, 241
- ADC SFDR, 248
- ADC SNDR, 246
- Adder-based code conversion, 258
- Advanced pipelining techniques, 418
- Advanced SAR ADCs, 330
- Advanced SAR techniques, 418
- Aliased components, 440
- Aliasing, 21
- Aliasing of harmonics, 62
- Amplifier with sampling, 363
- Amplitude error due to phase mismatch, 449
- Analog multiplexer, 159
- Analog offset cancelation, 119
- Analog-to-digital conversion, 1
- Asynchronous SAR ADCs, 330

- Background calibration, 460
- Bandwidth mismatch in interleaved ADCs, 452
- Binary search, 284, 353
- Binary search tree, 284
- Binary-weighted CDACs, 212
- Binary-weighted IDACs, 180
- Bootstrapped sampling circuit, 43
- Bootstrapped sampling switch, 43
- Bootstrapping capacitor, 44
- Bootstrapping imperfections, 49
- Bottom-plate sampling ADC, 299
- Bubble correction, 272
- Bubbles in thermometer code, 261
- Bulk connections, 48

- Capacitance density, 12
- Capacitance nonlinearity, 314
- Capacitor DACs, 210
- Capacitor mismatch, 534
- Capacitor mismatch in MDAC, 412
- Capacitor parasitics, 218
- Capacitor scaling, 519
- Capacitor structures, 10, 217
- Cascaded modulators, 559
- Cascaded-integrator-comb filter, 577
- Cascode current source, 191
- Cascode device, 48

- CDAC properties, 211
- CDAC speed limitations, 214
- CDACs, 210
- Channel area, 105
- Charge conservation, 6
- Charge injection, 36
- Charge injection in MDAC, 392
- Charge injection in SAR ADC, 301
- Charge sharing, 8, 49, 453
- Charge sharing in interleaved ADC, 453
- Charge-redistribution ADC, 290
- Clock boosting, 311
- Clock feedthrough, 35, 41
- Clock generation, 307, 334, 471, 531
- Clock jitter, 55, 546, 553
- Clocked comparator, 72
- Clocking in DACs, 154
- Coarse and fine delay control, 469
- Code transition, 236
- Column thermometer code, 193
- Common-mode generation by resistors, 299
- Comparator delay, 555
- Comparator design, 70
- Comparator design parameters, 84
- Comparator in a flash ADC, 253
- Comparator noise, 87, 137
- Comparator offset, 74, 252, 418, 545
- Comparator offset in flash ADCs, 260
- Comparator response, 322
- Comparator speed, 85
- Comparator waveforms, 131
- Complementary sampling switches, 31
- Continuous-time Delta-Sigma loop, 495
- Continuous-time first-order Delta-Sigma loop, 498
- Control logic for switches, 307
- Controlled positive feedback, 80
- Corruption due to phase mismatch, 451
- Current-steering DACs, 174

- DAC, 142
- DAC complexity reduction, 346
- DAC gain error, 145
- DAC glitch impulse, 151
- DAC kT/C noise, 316
- DAC offset, 144
- DAC output in SAR ADC, 286
- DAC performance parameters, 143
- DAC settling time, 150
- DAC SFDR, 152
- DAC SNDR, 152
- DAC waveforms in SAR ADC, 290
- DAC with supply inductance, 176
- Data-weighted averaging, 569

- DC nonlinearity, 318
- Decimation filter using a counter, 579
- Decimation filtering, 575
- Decision threshold, 236
- Decoupling capacitor in RDACs, 169
- Delta-Sigma modulator, 487
- Delta-Sigma modulator design, 509
- Delta-Sigma modulator nonidealities, 534
- Delta-Sigma modulator performance metrics, 504
- Delta-Sigma modulators with multibit quantizers, 567
- Deterministic mismatches in IDACs, 178
- Device stress, 66
- Differential bottom-plate sampling DC, 303
- Differential CDAC, 213
- Differential comparators, 274
- Differential delay control, 467
- Differential flash ADCs, 273
- Differential IDAC, 188
- Differential integrator, 510
- Differential nonlinearity, 145
- Differential pipelined ADC, 357
- Differential RDACs, 162
- Differential sampler, 60
- Differential sampling, 40
- Differential switching in IDACs, 176
- Differential top-plate sampling ADC, 298
- Digital codes in DACs, 153
- Digital offset cancelation, 123
- Digital-to-analog conversion, 4
- Digital-to-analog converter, 142
- Discrete-time Delta-Sigma modulator, 488
- Discrete-time integrator, 488
- Distortion analysis, 33
- Distortion due to asymmetry, 190
- Distortion due to switch resistance, 29
- Divide-by-2 circuit, 307
- DNL profile, 195
- Dummy cells in matrix architecture, 193
- Dummy switches, 42
- Dynamic errors in IDACs, 202
- Dynamic nonidealities, 345
- Dynamic nonlinearity, 55
- Dynamic offset, 93
- Edge effects in MOSFETs, 178
- Effect of offset on dynamic range, 445
- Exponential growth, 74
- Feedforward in Delta-Sigma modulators, 573
- Feedthrough to ladder, 269
- First-order Delta-Sigma modulator, 492
- First-order Delta-Sigma modulator design, 509, 522
- Flash ADC, 251
- Flash ADC characteristic, 251
- Flash ADC front end, 253
- Flash ADC nonidealities, 260
- Flip-around amplifier, 364
- Flip-around MDAC, 368
- Folded ladder, 163
- Foreground calibration, 460
- Four-channel interleaved ADC, 441
- Fringe capacitors, 12, 222
- Full adder, 258
- Full scale, 236
- Gain calibration, 462
- Gain error, 14, 36, 211
- Gain error in cascaded modulators, 563
- Gain error in SAR ADC, 317
- Gain mismatch in frequency domain, 446
- Gain mismatch in interleaved ADCs, 446
- Gaussian surface, 8
- Gradients in IDACs, 194
- Gradients in RDACs, 163
- Gray code, 252
- Gray coding in flash ADC, 264
- Ground line voltage drop, 197
- Ground resistance, 199
- Ground tapering in IDACs, 199
- Half adder, 258
- Harmonic distortion, 15, 29, 60
- Harmonic distortion in IDAC, 187
- Harmonics, 15
- High-impedance node, 8
- High-order cascades, 562
- Hold mode, 45
- Hybrid flash/SAR ADCs, 338
- Hybrid SAR/pipelined ADCs, 432
- IDAC implementation, 178
- IDAC INL calculation, 182
- IDAC nonlinearity, 186, 188
- IDAC speed limitations, 200
- IDAC unit cell, 192
- IDACs, 174
- Ideal interleaved sampling, 439
- Ideal sampler, 20
- Idle tones, 547
- INL computation for IDAC, 182
- Input and output spectra of interleaved channels, 440
- Input buffer, 268
- Input offset, 84
- Input offset storage, 122
- Integral nonlinearity, 147
- Integrator swing reduction, 516
- Integrator transfer function, 490
- Interchannel crosstalk, 453
- Interface between StrongArm latch and decoder, 255
- Interleaved ADCs, 439
- Interleaved channels with different offsets, 444
- Interleaved samplers, 452

- Interleaving advantages, 442
- Interleaving disadvantages, 443
- Interpolation in flash ADCs, 281
- Kickback noise, 88, 98, 137, 252, 265
- Kickback noise in flash ADC, 265
- Kickback noise reduction, 267
- kT/C noise, 38, 61, 535
- kT/C noise in MDAC, 396
- Ladder model, 268
- Lateral capacitors, 221
- Linearity, 54
- Local logic including a latch, 196
- MASH, 562
- Matched transistors, 70
- Matching issues in IDACs, 177
- Matrix architecture, 231
- Matrix IDACs, 192
- MDAC, 361, 366
- MDAC nonidealities, 383
- Memory effect in IDACs, 206
- Metastability, 107, 134, 262, 323
- Metastability in current-steering comparators, 107
- Metastability in SAR ADCs, 333
- Metastability in StrongArm comparator, 111
- Mismatch, 71
- Mismatch calibration, 457
- Mismatch shaping, 570
- Mismatches in interleaved ADCs, 443
- Missing code, 260
- Monotonic switching, 346
- MOS cap, 217
- MOS capacitor, 10
- Multibit pipelined stages, 376
- Multilayer capacitors, 219
- Multiply-by-2 stage, 361
- MUX, 159
- MUX decoder, 256
- Noise folding, 548
- Noise in a feedback system, 483
- Noise in amplification mode, 115
- Noise in StrongArm comparator, 112
- Noise in time domain, 113
- Noise shaping, 485
- Noise shaping in time domain, 487
- Noise transfer function, 483
- Non-flip-around amplifier, 362
- Non-flip-around MDAC, 366
- Nonidealities in pipelined ADCs, 383
- Nonlinear capacitance, 35
- Nonlinear settling, 543
- Nonlinearity, 14
- Nonmonotonicity, 149
- Nonoverlapping clocks, 531
- Nyquist zones, 62
- Nyquist's theorem, 21
- Nyquist-rate ADCs, 236
- Offset, 13
- Offset, 70
- Offset calibration, 460
- Offset cancelation, 119
- Offset cancelation in StrongArm comparator, 125
- Offset in a differential pair, 13
- Offset in MDAC, 386
- Offset in SAR ADC, 317
- Offset mismatch in interleaved ADCs, 444
- Offset optimization, 132
- Offset voltage in current sources, 179
- On-resistance variation, 34
- ONEs-counter decoder, 257
- Op amp models, 357
- Op-amp gain in MDAC, 401
- Op-amp noise, 537
- Op-amp noise in MDAC, 407
- Op-amp nonidealities in MDAC, 396
- Op-amp offset, 391
- Op-amp slewing, 550
- Open-loop residue amplification, 431
- Output offset storage, 119
- Output resistance in IDACs, 174
- Overlap and redundancy, 346, 418
- Oversampling ADCs, 479
- Oversampling advantages, 503
- Oversampling disadvantages, 504
- Parallel-plate capacitor, 10
- Phase calibration, 463
- Phase mismatch in interleaved ADCs, 447
- Pipelined ADCs, 353
- Pipelined as assembly line, 355
- Pipelined chain, 370
- Pipelining in flash ADC, 263
- Pointer register in SAR ADC, 288
- Positive feedback, 74
- Power distribution in iDACs, 196
- Power-speed trade-off, 442
- Quantization, 2, 237, 284
- Quantization error, 237
- Quantization noise, 3, 239, 479
- Quantization noise calculation, 497
- Quantization noise model, 483
- Quantization noise spectrum, 482
- Quantizer, 2
- Quantizer output spectrum, 479
- R-2R IDACs, 183
- Random asymmetries, 70
- Random mismatches in CDACs, 216
- Random mismatches in IDACs, 179

- Random mismatches in RDACs, 165
- RDAC speed limitations, 160
- RDCAs, 158
- Reference bounce in CDACs, 222
- Reference bounce in IDACs, 200
- Reference bounce in pipelined ADCs, 414
- Reference bounce in RDACs, 168
- Reference bounce in SAR ADC, 319
- Reference buffer, 170, 223
- Reference noise, 320
- Reference settling, 545
- Regeneration, 77
- Regeneration time constant, 78
- Residue, 286
- Residue in hybrid flash/SAR ADC, 422
- Residue in pipelined ADC, 357, 373
- Residue manipulations, 383
- Resistor ladder, 158
- Resistor-ladder DACs, 158
- Ring counter, 307
- ROM decoder, 254
- Row thermometer code, 193
- RS latch, 98, 133

- Sample-and-hold function, 21
- Sampler input current, 67
- Sampler time constant, 24
- Sampling circuit design, 60
- Sampling circuit design procedure, 61
- Sampling circuits, 20
- Sampling in flash ADCs, 253
- Sampling rate, 20
- SAR ADC, 284
- SAR ADC nonidealities, 310
- SAR ADC speed limitations, 322
- SAR ADC structure, 284
- SAR ADC using CDAC, 290
- SAR ADC with data register, 284
- SAR ADCs with multibit quantizers, 334
- SAR and pipelined comparison, 380
- SAR logic, 287, 307
- SAR logic using RS latch, 288
- SAR offset, 423
- SAR register, 284
- Second-order Delta-Sigma modulator, 499
- Second-order Delta-Sigma modulator design, 518
- Segmented IDACs, 184
- Signal distribution in IDACs, 196
- Signal-to-noise ratio, 3, 60, 239
- Simulation of comparator noise, 117
- Sinc droop, 22
- SNR calculation, 517, 520
- Source follower, 269
- Sources of mismatch, 448
- Speed optimization, 132
- Speed-power trade-off, 442

- Split CDACs, 224
- Stability, 557
- Static nonidealities, 342
- Static nonlinearity, 55
- StrongArm comparator, 89, 94, 96, 129
- Switch junction capacitance, 312
- Switch resistance, 310
- Switched-capacitor circuits, 6
- Switching action in IDACs, 175

- Tapering in SAR ADCs, 326
- Thermometer code, 180
- Thermometer to 1-of- n decoder, 254
- Thermometer-to-binary conversion, 254, 259
- Thermometer-to-Gray code conversion, 265
- Thermometric CDACs, 212
- Thermometric DAC, 180
- Thermometric IDACs, 180
- Three-level sampling, 348
- Time trellis, 287
- Time-interleaved channels, 439
- Timing mismatch, 422
- Timing mismatch in flash ADC, 270
- Timing mismatch in pipelined ADCs, 415
- Tones due to gain mismatch, 451
- Tones due to offset mismatch, 444
- Tones due to phase mismatch, 451
- Top-plate sampling, 290
- Top-plate sampling ADC, 293
- Total harmonic distortion, 17, 55
- Track mode, 45
- Track-and-hold circuit, 23
- Track-mode distortion, 62
- Transistor mismatch, 102
- Transistor sizing in IDCs, 192
- Two-rank interleaving, 455

- U-shaped flash ADC, 280
- Uniform sampling, 2

- Variable-delay line, 465
- Virtual ground, 10

- Zero-order hold, 21

