

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

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

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

Dudley R. Kay, *Director of Book Publishing*
John Griffin, *Senior Editor*
Lisa Dayne, *Assistant Editor*
Linda Matarazzo, *Editorial Assistant*
Savoula Amanatidis, *Production Editor*

IEEE Circuits & Systems Society, *Sponsor*
CAS-S Liaison to IEEE Press, Jaime Ramirez-Angulo

Also of Interest from IEEE Press . . .

Oversampling Delta-Sigma Data Converters: Theory, Design and Simulation
edited by James C. Candy, *AT&T Bell Laboratories* and Gabor C. Temes, *Oregon State University*
1992 Hardcover 512 pp IEEE Order No. PC2741 ISBN 0-87942-285-8

Clock Distribution Networks in VLSI Circuits and Systems
edited by Eby G. Friedman, *University of Rochester*
1995 Hardcover 544 pp IEEE Order No. PC4127 ISBN 0-7803-1058-6

Nonvolatile Semiconductor Memories: Technologies, Design, and Applications
edited by Chenming Hu, *University of California, Berkeley*
1991 Hardcover 496 pp IEEE Order No. PC2634 ISBN 0-87942-269-6

Monolithic Phase-Locked Loops and Clock Recovery Circuits: Theory and Design
edited by Behzad Razavi, *AT&T Bell Laboratories*
1996 Hardcover 512 pp IEEE Order No. PC5620 ISBN 0-7803-1149-3

Routing in the Third Dimension: From VLSI Chips to MCMs
Naveed A. Sherwani, Siddharth Bhingarde, and Anand Panyam, *Microprocessor Division, Intel Corporation*
1995 Hardcover 376 pp IEEE Order No. PC4473 ISBN 0-7803-1089-6

Circuits and Systems Tutorials
Chris Toumazou, Editor; Nick Battersby and Sonia Porta, Assistant Editors
1996 Softcover 700 pp IEEE Order No. PP5596 ISBN 0-7803-1170-1

Delta-Sigma Data Converters

*Theory, Design,
and Simulation*

Edited by

Steven R. Norsworthy
Motorola

Richard Schreier
Oregon State University

Gabor C. Temes
Oregon State University



IEEE
PRESS

IEEE Circuits & Systems Society, *Sponsor*

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

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

IEEE Press Marketing
Attn: Special Sales
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331
Fax: (908) 981-9334

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

© 1997 by the Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394

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

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-7803-1045-4

IEEE Order Number: PC3954

Library of Congress Cataloging-in-Publication Data

Delta-sigma data converters : theory, design, and simulation / edited
by Steven R. Norsworthy, Richard Schreier, Gabor C. Temes ; IEEE
Circuits & Systems Society, sponsor.
p. cm.
Includes index.
ISBN 0-7803-1045-4
1. Analog-to-digital converters. 2. Digital-to-analog converters.
3. Modulators (Electronics)--Design. I. Norsworthy, Steven R.
(date) . II. Schreier, Richard (date) . III. Temes, Gabor C.
(date) . IV. IEEE Circuits & Systems Society.
TK7887.6.D45 1996
621.3815'322--dc20 96-14774
CIP

Contents

Preface xv
Introduction xvii

Chapter 1 An Overview of Basic Concepts 1
J. C. Candy

- 1.1 Introduction 1
- 1.2 Digital Modulation 3
 - 1.2.1 Quantization 3
 - 1.2.2 Delta-Sigma Modulation 5
 - 1.2.2.1 First-Order Feedback Quantizer 5
 - 1.2.2.2 Modulation Noise in Busy Signals 7
 - 1.2.2.3 Pattern Noise from $\Delta\Sigma$ Modulation with dc Inputs 8
 - 1.2.2.4 Dead Zones in $\Delta\Sigma$ Modulation 10
 - 1.2.2.5 Influence of Circuit Parameters on $\Delta\Sigma$ Modulation 11
 - 1.2.3 High-Order Modulation 14
 - 1.2.3.1 Predicting In-Band Values of Quantization Error 14
 - 1.2.3.2 Noise in High-Order $\Delta\Sigma$ Modulation 14
 - 1.2.3.3 Dynamic Range of the Modulators 16
 - 1.2.3.4 Influence of Circuit Parameters on Second-Order Modulators 19
 - 1.2.3.5 Limit Cycles in Third-Order $\Delta\Sigma$ Modulators 20
 - 1.2.3.6 Noise Shaping Using Filters with Nonmonotonic Transfer Functions 22
 - 1.2.4 Some Alternative Modulator Structures 23
 - 1.2.4.1 Error Feedback 23
 - 1.2.4.2 Cascaded Modulators 24
 - 1.2.4.3 Delta Modulation 26

1.3	Decimating the Modulated Signal	28
1.3.1	Multistage Decimation	28
1.3.2	Design of the First-Stage Decimator	29
1.3.3	Implementing sinc Decimators	32
1.3.4	The Low-Pass Filter	35
1.4	Oversampling D/A Converters	36
1.4.1	Demodulating Signals at Elevated Word Rates	36
1.4.2	Interpolating with sinc ^K -Shaped Filter Functions	37
1.4.3	Demodulator Stage	38
1.4.3.1	Quantizing the Digital Signal	38
1.4.3.2	Quantization with Error Feedback	38
1.4.3.3	Cascaded Demodulators	40
1.4.3.4	Circuit Design for $\Delta\Sigma$ Demodulation	40
1.5	Conclusion	41
	References	41

Chapter 2 Quantization Noise in $\Delta\Sigma$ A/D Converters 44

Robert M. Gray

2.1	Introduction	44
2.2	Uniform Quantization	45
2.3	Additive White-Noise Approximation	46
2.4	Characteristic Function Method	53
2.5	Pulse Code Modulation Quantization Noise	55
2.6	Dithered PCM	58
2.7	Single-Loop $\Delta\Sigma$ Modulation	59
2.8	Two-Stage (Cascade or MASH) $\Delta\Sigma$ Modulation	64
2.9	Second-Order $\Delta\Sigma$ Modulation	66
2.10	Some Extensions	68
2.10.1	Dithered Single-Loop $\Delta\Sigma$ Modulation	68
2.10.2	Multistage and Higher Order $\Delta\Sigma$ Modulation	68
2.10.3	Leaky Integrating $\Delta\Sigma$ Modulation	69
2.10.4	Multibit Quantizer, Single-Bit Feedback	69
2.10.5	Related Work	69
2.11	Conclusion	70
	Acknowledgments	70
	References	70

Chapter 3 Quantization Errors and Dithering in $\Delta\Sigma$ Modulators 75

Steven R. Norsworthy

3.1	Introduction	75
3.1.1	Problems with Empirically Based Reports on $\Delta\Sigma$ Modulators	77
3.1.2	Steps Taken to Ensure Accuracy of Results	77
3.2	Basic Structures and Terminology	78

3.3	Observability of Periodic Sequences	80
3.4	Tones in Single-Stage $\Delta\Sigma$ Modulators	84
3.4.1	Second-Order Modulator	85
3.4.2	Third-Order Modulator	88
3.4.3	Fifth-Order Modulator	92
3.4.4	Baseband Demodulation of Tones Near $f_s/2$	95
3.4.5	Higher-Order and Multibit Single-Stage Modulators	97
3.5	Tones in Multistage $\Delta\Sigma$ Modulators	98
3.6	Tones in $\Delta\Sigma$ Converter Hardware	100
3.6.1	Third-Order Digital Modulator Test	101
3.6.2	Fifth-Order Digital Modulator Test	102
3.6.3	Multistage Modulator Test	104
3.7	Dither in PCM Quantizers	104
3.7.1	Nonsubtractive Dither	104
3.7.2	Subtractive Dither	105
3.8	Dither Topologies for $\Delta\Sigma$ Modulators	107
3.8.1	Dither Topologies for Single-Stage Modulators	107
3.8.2	Dither Topologies for Multistage Modulators	109
3.9	Empirical Studies of Noise-Shaped Dithering	112
3.9.1	Second-Order Modulator	112
3.9.2	Third-Order Modulator	116
3.9.3	Fifth-Order Modulator	118
3.9.4	Effect of Dither on Tones Near $f_s/2$	119
3.9.5	Multistage Modulators	120
3.10	Dither Generation	121
3.11	Dither in A/D Modulators	121
3.11.1	Single-Stage A/D Modulator Example	121
3.11.2	Multistage A/D Modulators	121
3.12	Subtractive Noise-Shaped Dithering	123
3.13	Dynamic Noise-Shaped Dithering	124
3.13.1	Theory of Dynamic Dither	124
3.13.2	Implementation Considerations of Dynamic Dither	127
3.14	Dithered Multibit Noise-Shaping Coders	130
3.14.1	Stability Test with Dither	130
3.15	Chaos versus Noise-Shaped Dither	131
3.16	Other Techniques	134
3.17	Conclusion	135
	References	136

Chapter 4 Stability Theory for $\Delta\Sigma$ Modulators 141

Robert W. Adams and Richard Schreier

4.1	Introduction	141
4.2	Linear Analysis	142
4.2.1	The Linear Model	142
4.2.2	Root Locus of a High-Order Modulator	144
4.2.3	Describing Function Method	145

- 4.3 First- and Second-Order Modulators 147
 - 4.3.1 First-Order Modulator 148
 - 4.3.2 Second-Order Modulator 149
- 4.4 Practical Design Methodology 152
 - 4.4.1 Cookbook Design Procedure 152
 - 4.4.2 SNR Limits 153
 - 4.4.3 Sixth-Order NTF 154
 - 4.4.4 Design Trade-Offs 156
- 4.5 Continuous-Time Design 158
- 4.6 Nonlinear Stabilization Techniques 162
- 4.7 Conclusion 163
 - Acknowledgments 163
 - References 163

Chapter 5 The Design of High-Order Single-Bit $\Delta\Sigma$ ADCs 165

Robert W. Adams

- 5.1 Introduction 165
- 5.2 Motivation for Using High-Order Single-Bit Loops 166
- 5.3 Design Choices: SC or Active-RC? 167
- 5.4 Stability 170
 - 5.4.1 Stability and the Uncontrolled Input Signal:
A Practical Guide to Safe Operation 170
 - 5.4.2 Transient Input Signals and Stability:
The Case for Mild Prefiltering 170
- 5.5 Choices for the NTF 172
 - 5.5.1 n th-Order Pure Differentiation 172
 - 5.5.2 Butterworth High-Pass Response 173
 - 5.5.3 Complex Zeros on the Unit Circle (Inverse Chebyshev) 174
- 5.6 Comparison of Loop Topologies 174
 - 5.6.1 Chain of Integrators with Weighted Feedforward Summation 176
 - 5.6.2 Chain of Integrators with Feedforward Summation and Local
Resonator Feedbacks 177
 - 5.6.3 Chain of Integrators with Distributed Feedback 178
 - 5.6.4 Chain of Integrators with Distributed Feedback and Distributed
Feedforward Inputs 179
 - 5.6.5 Error Feedback Only 180
- 5.7 Nonlinear Global Stabilization Techniques 183
- 5.8 Practical Measures for Preventing Idle Tones 185
- 5.9 Practical Implementation of a Stereo 18-Bit $\Delta\Sigma$ ADC IC 186
 - 5.9.1 Noise-Shaping Modulator IC 186
 - 5.9.2 Switched-Capacitor Loop Filter Design 186
 - 5.9.3 Circuit Noise Considerations 189
 - 5.9.4 Stabilization Using Integrator Reset 190
 - 5.9.5 Op-Amp Design 190
 - 5.9.6 Results and Comments 191
 - References 192

Chapter 6 The Design of Cascaded $\Delta\Sigma$ ADCs 193

Mike Rebeschini

- 6.1 Introduction 193
- 6.2 System Design 195
 - 6.2.1 Comparison of Single-Loop and Cascaded Designs 195
 - 6.2.1.1 Single-Loop Designs 195
 - 6.2.1.2 Cascaded Designs 196
 - 6.2.2 Analytical Linearized Modeling 196
 - 6.2.3 Software Simulations 197
- 6.3 Analysis of Specific Cascaded Architectures 199
 - 6.3.1 Third-Order (1–1–1) Modulator 199
 - 6.3.2 Third-Order (2–1) Modulator 203
- 6.4 Circuit Topologies for Third-Order (1–1–1) Cascade 204
 - 6.4.1 Autozeroed Integrator 204
 - 6.4.2 First Modulator of Third-Order (1–1–1) Cascade 206
 - 6.4.3 Second and Third Modulators of Third-Order (1–1–1) Cascade 207
- 6.5 Sources of Error for the Third-Order (1–1–1) Cascade 209
- 6.6 Experimental Results for the Third-Order (1–1–1) Cascade 211
- 6.7 Continuous-Time Cascaded $\Delta\Sigma$ Modulators 213
- 6.8 Conclusion 217
 - References 218

Chapter 7 High-Speed Cascaded $\Delta\Sigma$ ADCs 219

Brian Brandt

- 7.1 Introduction 219
- 7.2 $\Delta\Sigma$ Modulation at Low Oversampling Ratios 220
- 7.3 A Cascaded Multibit $\Delta\Sigma$ Modulator 222
 - 7.3.1 Interstage Coupling 225
- 7.4 Implementation of the Cascaded Multibit Modulator 229
 - 7.4.1 Gain Error 230
 - 7.4.2 Incomplete Settling 232
 - 7.4.3 Integrator Leakage 232
- 7.5 Design of the Cascaded Multibit Modulator 233
- 7.6 Experimental Results 239
- 7.7 Summary 242
 - References 242

Chapter 8 Delta-Sigma ADCs with Multibit Internal Converters 244

Richard L. Carley, Richard Schreier, and Gabor C. Temes

- 8.1 Introduction 244
- 8.2 Multibit Noise-Shaping Modulator Architectures 245
- 8.3 DAC Architectures for Improved Linearity 247

8.3.1	Internal DAC Topology	247
8.3.2	Element-Trimming Approaches	249
8.3.2.1	One-Time Trimming Methods	249
8.3.2.2	Repeated Trimming Methods	251
8.3.2.3	Other Element-Matching Methods	251
8.3.3	Dynamic Element Matching	251
8.3.3.1	Dynamic Element Randomization	253
8.3.3.2	Dynamic Element Rotation—Barrel Shifter	256
8.3.3.3	Individual Level Averaging	259
8.3.3.4	Noise-Shaped Element Usage	260
8.4	Digital Correction Techniques	264
8.4.1	$\Delta\Sigma$ ADC Architectures with Error-Storing Random-Access Memory	264
8.4.2	The Calibration of the Digitally Corrected $\Delta\Sigma$ ADC	265
8.4.3	An Improved Digital Correction System	267
8.4.4	Cascade $\Delta\Sigma$ ADC Systems Using Digital Correction	269
8.4.5	Digitally Corrected $\Delta\Sigma$ ADC with Companding Quantizer	270
8.5	Dual-Quantizer ADC Architectures	273
8.5.1	The Leslie–Singh Architecture	273
8.5.2	Dual-Quantization Cascade ADC Architectures	275
8.5.3	Dual-Feedback Single-Path ADC Architecture	276
8.6	Conclusion	277
	References	278

Chapter 9 The Design of Bandpass $\Delta\Sigma$ ADCs 282

Stephen Jantzi, Richard Schreier, and Martin Snelgrove

9.1	Introduction	282
9.2	Bandpass $\Delta\Sigma$ Transfer Function Design	284
9.2.1	The Linear Model	285
9.2.2	Band Location	285
9.2.3	Low-Pass Prototype Method	286
9.2.4	Design by Generalized Filter Approximator	287
9.2.4.1	The Design of $H(z)$	287
9.2.4.2	The Design of $G(z)$	288
9.2.4.3	An Example Modulator	288
9.2.5	Modulator Performance	289
9.2.5.1	Linear Model Predictions	289
9.2.5.2	Simulations	290
9.2.5.3	SNR versus Modulator Order and Oversampling Ratio	291
9.3	Bandpass $\Delta\Sigma$ Modulator Design	292
9.3.1	Standard Switched-Capacitor Design	292
9.3.2	Switched-Capacitor N -Path Design	294
9.3.3	Practical Considerations in Discrete-Time Systems	296
9.3.3.1	Capacitor and $1/f$ Noise	296
9.3.3.2	Op-amp Speed	297
9.3.3.3	Sample-and-Hold Circuits	297
9.3.4	Continuous-Time Design	297
9.4	Decimation for Bandpass Modulators	300

9.5	Experimental Results	301
9.5.1	Reported Implementations of Bandpass $\Delta\Sigma$ Modulators	301
9.5.1.1	September 1990	301
9.5.1.2	September 1991	301
9.5.1.3	May 1992	302
9.5.1.4	June 1992	302
9.5.1.5	February 1993	303
9.5.1.6	May 1994	304
9.5.2	Performance Summary	304
9.5.3	Comments on Bandpass $\Delta\Sigma$ Modulator Performance	304
9.6	The Future of Bandpass $\Delta\Sigma$ Modulation	305
9.6.1	High-Frequency Converters	305
9.6.2	Bandpass $\Delta\Sigma$ DACs	306
9.7	Conclusion	306
	References	306

Chapter 10 Architectures for $\Delta\Sigma$ DACs 309

Gabor C. Temes, Shaofeng Shu, and Richard Schreier

10.1	Introduction	309
10.2	Architectures for the Noise-Shaping Loop	311
10.2.1	Delta–Sigma Loop	311
10.2.2	Error Feedback Structure	313
10.2.3	Cascade Structure	315
10.2.4	Multibit Quantizer Loops	316
10.3	Design Example 1: A Fifth-Order Single-Bit Noise-Shaping Loop	321
10.3.1	The Noise Transfer Function	321
10.3.2	The Modulator Structure	322
10.4	Design Example 2: A Third-Order (2+1) Multibit Cascade Noise-Shaping Loop	324
10.5	Conclusion	331
	References	332

Chapter 11 Analog Circuit Design for $\Delta\Sigma$ ADCs 333

Brian Brandt, Paul F. Ferguson, and Mike Rebeschini

11.1	Introduction	333
11.2	Architectural Considerations	334
11.3	Building Blocks	336
11.3.1	Input Integrator	336
11.3.2	Specifications	337
11.3.3	Fully Differential SC Integrator	337
11.3.4	Op-Amp	343
11.4	Circuit Nonidealities	348
11.4.1	Effects of Component Nonidealities on the Integrator Performance	348
11.4.2	Nonlinear Effects	350
11.4.3	Intrinsic Noise	353

- 11.5 Modulator Component Design Considerations 356
 - 11.5.1 The Feedback DAC 356
 - 11.5.1.1 Reference Nonidealities 356
 - 11.5.1.2 Charge-Taking Nonidealities 357
 - 11.5.1.3 Charge-Delivery Nonidealities 358
 - 11.5.2 The Comparator 360
 - 11.5.3 The Clock Generation Circuitry 361
- 11.6 System-Level Considerations 361
 - 11.6.1 Dynamic Range Considerations 361
 - 11.6.2 Clock Jitter 363
 - 11.6.3 Input Impedance 363
- 11.7 Layout Considerations 365
 - 11.7.1 Signal Paths 365
 - 11.7.2 Busses 365
 - 11.7.3 RF Coupling 366
 - 11.7.4 Interfacing to the ADC 367
- 11.8 Design Examples 369
 - 11.8.1 Second-Order Single-Stage $\Delta\Sigma$ Modulator 369
 - 11.8.2 Second-Order Cascaded Modulator (1–1) 373
- 11.9 Conclusion 378
- References 378

Chapter 12 Analog Circuit Design for $\Delta\Sigma$ DACs 380
Mike Rebeschini and Paul F. Ferguson, Jr.

- 12.1 Introduction 380
- 12.2 Building Blocks 381
 - 12.2.1 The Low-Resolution Input DAC 382
 - 12.2.2 Voltage References 384
 - 12.2.3 Reconstruction Filter 386
 - 12.2.3.1 Specifications 386
 - 12.2.3.2 Switched-Capacitor Reconstruction Using Biquads 387
 - 12.2.3.3 Noise-Shaping Filter 389
 - 12.2.3.4 Analog Decimation Filters for $\Delta\Sigma$ DACs 390
 - 12.2.3.5 Effect of Nonideal Integrator Transfer Function 394
 - 12.2.4 Discrete-Time/Continuous-Time Interface 395
 - 12.2.4.1 Final Discrete-Time Stage 395
 - 12.2.4.2 Continuous-Time Reconstruction Filter 396
 - 12.2.5 Other Nonideal Effects 397
 - 12.2.5.1 Intrinsic Noise 397
 - 12.2.5.2 Dynamic Range Considerations 398
- 12.3 Layout Considerations 398
 - 12.3.1 Differences from the ADC 398
 - 12.3.2 Layout Influences on the Architecture 399
- 12.4 Design Examples 399
 - 12.4.1 A One-Bit High-Order $\Delta\Sigma$ DAC 399
 - 12.4.2 A MASH DAC 403
 - 12.4.3 Recent Developments 404
- References 404

Chapter 13 Decimation and Interpolation for $\Delta\Sigma$ Conversion 406
Steven R. Norsworthy and Ronald E. Crochiere

- 13.1 Introduction 406
- 13.2 Scope of Design Trade-Offs and Alternatives 408
- 13.3 Basic Principles of Sampling Rate Conversion—Algorithm Issues 408
 - 13.3.1 Decimation by M 410
 - 13.3.2 Interpolation by L 411
 - 13.3.3 Duality 411
 - 13.3.4 Fractional Rate Changing 413
- 13.4 Multistage Conversion 413
- 13.5 Filter Design Considerations 416
 - 13.5.1 sinc^K Filters 416
 - 13.5.2 Half-Band Filters 418
 - 13.5.3 Ternary-Encoded FIR Filters 419
 - 13.5.4 Combining sinc^K Filters with FIR and IIR Filters 420
 - 13.5.5 Minimum-Phase FIR Filters 421
 - 13.5.6 Compensation Techniques 421
- 13.6 Digital Filter Structures 422
 - 13.6.1 Direct-Form and Transpose Direct-Form Decimators and Interpolators 424
 - 13.6.2 Polyphase Architectures for Decimators and Interpolators 426
 - 13.6.3 Multistage Architectures 429
- 13.7 Hardware Implementation—Architectural Issues 432
 - 13.7.1 Historical Background 432
 - 13.7.2 Architectural Features and Styles 432
 - 13.7.3 Arithmetic Processing Issues 433
 - 13.7.3.1 Bit-Serial and Digit-Serial Arithmetic 433
 - 13.7.3.2 Parallel Multiplication 435
 - 13.7.3.3 Combined Bit-Serial and Bit-Parallel Architectures 435
 - 13.7.3.4 Single-Multiplier Architectures 436
 - 13.7.4 DSP and Programmable Implementations 438
 - 13.7.4.1 Multirate Filtering Efficiency on DSPs 440
 - 13.7.4.2 Multistage Implementation Including sinc^K Filters 441
 - 13.7.4.3 Data Transfers and Buffering Between the Stages 441
 - 13.7.5 Mixed Analog and Digital Implementations 443
- 13.8 Conclusion 443
- Acknowledgments 444
- References 444

Chapter 14 CAD for the Analysis and Design of $\Delta\Sigma$ Converters 447
Christopher Wolff, John G. Kenney, and L. Richard Carley

- 14.1 Introduction 447
- 14.2 Multibit Converter Design 447
 - 14.2.1 Accumulation of Quantization Error 448
 - 14.2.2 Formulation and Solution of the Optimization Problem 449

- 14.2.2.1 Quantization Noise 450
 - 14.2.2.2 Constraints 450
 - 14.2.2.3 Representation of Closed-Loop Poles for the Optimizer 451
 - 14.2.3 Example Results 451
- 14.3 Simulation Based on Difference Equations 452
- 14.4 Simulation Based on the Quantizer Transfer Function 453
 - 14.4.1 Statistical Average Quantizer Transfer Function 453
 - 14.4.2 Distortion 455
- 14.5 Simulation Approaches 458
 - 14.5.1 Overview 458
 - 14.5.2 Model Comparison 459
 - 14.5.3 Efficient Macromodel Simulation 460
- 14.6 Conclusion 463
 - References 464
 - For Further Reading 466

Index 469

About the Editors 475

Preface

This book has the distinction of being the first original text on the subject of delta-sigma data conversion. Because of this, we thought our readers might appreciate a little background on how this book came into being. In 1992, the IEEE Press approached us about authoring a book on this subject. At the time, we felt that in order to do the job right, it would be necessary to enlist a number of authors renowned for their expertise in areas related to delta-sigma modulation. After all, the field of delta-sigma modulation encompasses many diverse disciplines: systems and control theory, digital and analog signal processing, VLSI integrated circuit design, and knowledge of consumer and communications applications. We did not feel qualified by ourselves to write about all of these. Therefore, we quickly concluded that we should gather additional experts who could join us in thoroughly covering these specialties and then treat the chapters more or less independently. With this style of book, there is always a tendency for the authors to overlap material, but we tried to orchestrate the overall effort to minimize overlap and to maximize original content. Nevertheless, if the book comes across to the readers as a collection of chapters, we confess that it really is just that! Nonetheless we hope that this text is useful and authoritative, and that it will quickly become a standard for both industry and academia in this important area.

We are grateful to the contributing authors for their excellent material, and to the reviewers (Professors Ian Galton and Stanley Lipshitz, Dr. David Rich, and others) for their valuable comments and suggestions.

*Steven R. Norsworthy
Richard Schreier
Gabor C. Temes*

Introduction

HISTORY (G. C. TEMES)

The basic concept underlying both delta modulators and delta-sigma converters is the use of feedback for improving the effective resolution of a coarse quantizer. An early description of this concept was given in a patent by Cutler [1], which was filed in 1954 and granted in 1960. His system was based on generating and subtracting from the input signal the quantization error of the low-resolution quantizer placed in the forward path of a feedback loop. A detailed analysis of Cutler's system, along with methods for improving and optimizing its performance, was given in 1962 by Spang and Schultheiss [2]. They also proposed the addition of a finite impulse response (FIR) loop filter in the feedback path. The resulting system aims to predict and correct the next quantization error value. Its output contains the original signal, plus the quantization error filtered essentially by the inverse transfer function of the FIR loop filter. This system is often called an *error feedback coder*.

A variant of the error feedback coder, the delta modulator, was proposed even earlier, in 1952, by de Jager [3]. It contained a quantizer (usually a single-bit one) in its forward path and a loop filter (in the simplest case, an integrator) in the feedback path. Thus, the signal and quantization error were both fed back, filtered, and subtracted from the input signal. The output therefore contained the input signal and the quantization error, both filtered (to a good approximation) by the inverse of the loop filter transfer function. Hence, the receiver had to duplicate the loop filter to restore the signal to its original form. When used as analog-to-digital converters (ADCs), both the error feedback coder and the delta modulator had serious practical problems. The error feedback structure needs high-accuracy analog subtractors in its feedback path, which are not readily

realizable. The delta modulator suppresses the low-frequency end of the signal spectrum, which needs to be restored at the receiver end. This causes the dynamic range to decrease with signal frequency and causes a cumulative error due to line noise.

It was proposed by Inose, Yasuda, and Murakami in 1962 [4] to add the loop filter to the front end of a delta modulator and then move it inside the loop. For the simple case of an integrator used as loop filter, the resulting system contained an integrator in the forward path, followed by the (1-bit) quantizer, and the feedback loop contained only a 1-bit digital-to-analog converter (DAC). Since this system contained a delta modulator and an integrator, they named it a delta-sigma modulator, where the “sigma” denoted the summation performed by the integrator. It was often called sigma-delta modulator by later workers. Today, both names are in use. The output of the modulator contains the original input signal plus the first difference of the quantization error, as was the case for the error feedback coder. Thus, both delta-sigma and error feedback coders are noise-shaping modulators. They suppress the error in the baseband and thus achieve improved dynamic range across this band independent of the signal frequency. The structure is free of the practical problems of the previous coders; the only component in the feedback loop is a 1-bit DAC, which can be made nearly ideal with careful design.

In the 34 years since its first description, the basic delta-sigma converter has been modified many times and in many ways. The first important change was suggested in 1977 by Ritchie [5]. He proposed using several integrators in cascade in the forward path to create a higher order loop filter, with each integrator receiving an additional input from the DAC. The latter was needed to prevent instability. In an influential paper published in 1985 [6], Candy gave extensive design information on the double-integrator loop. Even so, for more than two integrators in the loop, the stability was conditional and had to be verified by numerical simulation. In 1987, Lee and Sodini gave design techniques for stable higher order loops [7, 8]. Based on these, delta-sigma ADCs with fourth- and fifth-order loop filters containing several cascaded switched-capacitor (SC) integrators and resonators have been successfully produced by several integrated circuit (IC) companies.

A different approach for the design of stable high-order delta-sigma ADCs was proposed in 1986 by Hayashi et al. [9]. In their system (which they named MASH, for multistage noise shaping) a single-integrator delta-sigma ADC processes the signal, and the resulting large quantization error is converted by a second delta-sigma ADC into digital data. The digital outputs of the two ADCs are combined through a digital stage that cancels the quantization error of the first ADC and differentiates the quantization error of the second. The resulting digital signal contains a high-pass filtered replica of the quantization error; the order of filtering is the sum of the orders of the two loop filters. To prevent the leakage of unfiltered quantization noise to the output, the analog integrator in the first (signal) loop needs to function nearly ideally. This makes the practical realization somewhat difficult. The principle can be extended to the realization of high-order converters, and successful third- and fourth-order MASH ADCs have been reported.

Another way of enhancing the performance of a delta-sigma ADC is to use a multibit internal quantizer. This, however, necessitates the inclusion of a multibit DAC in the feedback loop. The linearity of this DAC limits the linearity of the complete ADC, and hence its design is a daunting task. Several techniques have been used to overcome this problem. In 1989, Carley suggested the use of dynamic element matching to reduce the effect of

DAC nonlinearity on the overall response [10]; also, Larson et al. [11] proposed the use of digital correction to cancel it. In 1990, Leslie and Singh introduced an architecture that used a single-bit DAC and a multibit ADC to achieve both good linearity and low quantization noise [12]. Other dual-quantization techniques were also proposed, as well as techniques that, instead of randomizing the noise due to the unavoidable nonlinearity error of the multibit DAC, high-pass filtered it (see Chapter 8 of this book for details).

The development of delta-sigma DACs followed a similar path. Ritchie et al. [13] proposed an interpolative DAC, based on an error feedback configuration, in 1974; in 1986, Candy and Huynh extended the concept to a second-order system, which used the MASH technique (proposed shortly before Hayashi et al.!) to combine the output signals of two first-order digital loops [14]. A 16-bit single-loop second-order delta-sigma DAC, using an error feedback topology, was described in 1987 by Naus et al. [15]. Higher order loops have followed; in 1991, Sooch et al. described an 18-bit DAC using a fifth-order loop filter [16]. Several techniques were also offered for realizing delta-sigma DACs with multibit internal DACs. Carley and Kenney [17], Schouwenaars et al. [18], and Xu et al. [19] described such systems.

Recently, great interest has developed in bandpass delta-sigma converters that offer efficient signal processing for digital wireless devices. The idea of constructing a delta-sigma ADC tuned to a nonzero center frequency was first disclosed in 1987 [20] but was independently reinvented a number of times [21–23]. Several realizations of bandpass delta-sigma converters have been reported (see Chapter 9), with the most recent involving the use of a sixth-order multistage (MASH) architecture to convert signals in the 3.25±0.1 MHz range with 11-bit accuracy [24]. All low-pass delta-sigma architectures (including multibit systems and DACs) have their bandpass counterparts.

THEORY (R. SCHREIER)

Although it is possible to divine some intuitive feel for the operation of a delta-sigma modulator from its behavior in the time domain, the easiest way to understand a delta-sigma modulator is in terms of the frequency-domain description of its “linear model.” In this model a nonlinear operation, quantization, is replaced by the addition of a noise signal. Linear system theory is then invoked to show that the output of the modulator is therefore the sum of the filtered input signal and the filtered quantization noise. The two components can be filtered independently, under the control of the designer. This allows delta-sigma modulators to separate spectrally the input signal from the noise introduced by quantization. Subsequent digital filtering then removes the out-of-band noise, leaving an accurate digital replica of the original analog input.

The understanding afforded by the linear model is sufficient to explain the most fundamental characteristic of a delta-sigma modulator, namely the shaping of quantization noise. The linear model is also accurate enough to estimate the performance of a wide variety of delta-sigma modulator architectures. Unfortunately, the linear model breaks down when assumptions regarding the nature of the quantization noise, such as whiteness or the possession of a particular probability density function, fail to hold [25]. Two notable instances of these failures of the linear model are the generation of idle-channel tones and instability in the modulator. Tones result when the quantization noise contains

discrete (or nearly discrete) spectral lines. Chapter 2 discusses the available theoretical results regarding the nature of the quantization noise in a delta-sigma modulator, while Chapter 3 deals with the practical resolution of the tone problem. Although the available theory is unable to answer the question "How bad are the tones?," the techniques of Chapter 3 are effective in eliminating idle-channel tones.

Stability is the second major unanswered theoretical question regarding delta-sigma modulation. For a number of years it was believed that modulators of order 3 or more were automatically unstable, but after Ritchie [5] and Lee [7] showed that high-order modulators could in fact be stabilized, high-order designs proliferated. Unfortunately, those designs were achieved through the application of ad hoc criteria (backed up by simulation or the construction of a prototype) rather than a rigorous design methodology. Judging by the fact that it took more than a decade for the stability of a second-order modulator to be rigorously established, and even then only for a constant input [26], it is likely that the question "Is my modulator stable?" will continue to be addressed by simulation and other approximate methods well into the next millennium. Chapter 4 presents the available theoretical results and ad hoc methods so that new designers may construct high-order modulators even though an iron-clad design methodology is not yet available.

APPLICATIONS (S. R. NORSWORTHY)

It took nearly three decades from the earliest works [1–4] on oversampled noise shaping and delta modulation techniques until the introduction of high-volume commercial products that used these principles. Semiconductor technology simply did not reach very large scale integrated (VLSI) proportions until the mid-1980s, and without the fine-line complementary metal-oxide-semiconductor (CMOS), the digital filtering required in delta-sigma converters for decimation and interpolation makes these circuits too expensive.

Noise shaping and oversampling principles lend themselves most favorably to applications that require one or more of the following attributes: relatively low frequency (<100 kHz) and high resolution (>12 bits), high-complexity high-integration single-chip designs using mixed analog and digital signal processing, and low-cost CMOS-only processing. Certainly the delta-sigma technique has not completely supplanted more traditional data conversion techniques. Even at the time of this writing, there are architectural techniques and designs that compete favorably with delta-sigma on the basis of cost and power dissipation.

The first (and probably largest) application of delta-sigma conversion is in the field of digital telephony. There are many subsets in this field with widely varying requirements. Voiceband codecs for the public switched telephone network require 13-bit linear (8-bit companded) resolution and 8 ksamp/s conversion rates. Echo-canceling modems such as CCITT V.32 and V.34 require 12–16-bit resolution and 8 ksamp/s rates. ISDN (Integrated Services Digital Network) U-interface transceivers require 13–16-bit linearity and 80–160-ksamp/s rates. Digital cellular telephones utilize delta-sigma both for voiceband speech coding and for IF-to-baseband radio interface data conversion, such as I/Q modulation and demodulation. Bandpass delta-sigma modulators (Chapter 9) are beginning to be explored as alternatives to low-pass modulators for the IF-to-baseband inter-

face, and additional work is being explored to make them produce high-enough resolution at relatively high sampling rates without consuming too much power.

Possibly the most obvious application that takes full advantage of the inherent qualities of delta-sigma conversion is digital audio [7–10, 15–18]. Delta-sigma conversion enables nearly unlimited high-resolution data conversion without precise matching of analog components, and certainly there is a never-ending thirst for more precision in digital audio applications! Data converters for digital audio fall into two basic categories: consumer grade, requiring 14–18-bit resolution, and professional grade, requiring 18–20-bit resolution. The higher quality audio converter designs are made possible through higher order and/or multibit noise-shaping architectures. Even higher resolution (22–24-bit) delta-sigma designs can be found for instrumentation applications; however, the conversion rates are quite low (< 1 ksamp/s).

As technology continues to advance into the next century, a variety of other applications of the delta-sigma technique is likely to be developed. Already we see the beginnings of this, as the principle of noiseshaping has been successfully applied to image coding, clock jitter reduction, and frequency synthesis. Furthermore, there is considerable interest in the use of highly accurate multibit quantization combined with noise shaping. As these techniques are perfected, delta-sigma data conversion could be an enabling technology for higher speed consumer applications such as digital video. The trend toward higher levels of integration, lower cost, and lower voltage will cause the designer to think of new and unusual ways of replacing traditional analog functions with digital implementations. Analog circuitry will be needed only for data conversion in which the chip must communicate with the outside world.

REFERENCES

- [1] C. C. Cutler, "Transmission system employing quantization," U. S. Patent No. 2,927,962, March 8, 1960 (filed 1954).
- [2] H. A. Spang III and P. M. Schultheiss, "Reduction of quantizing noise by use of feedback," *IRE Trans. on Commun. Syst.*, pp. 373–380, Dec. 1962.
- [3] F. de Jager, "Delta modulation—a method of PCM transmission using the one unit code," *Philips Res. Rep.*, vol. 7, pp. 442–466, 1952.
- [4] H. Inose, Y. Yasuda, and J. Murakami, "A telemetering system by code modulation— Δ - Σ modulation," *IRE Trans. Space Electron. Telemetry*, vol. SET-8, pp. 204–209, Sept. 1962.
- [5] G. R. Ritchie, "Higher order interpolation analog to digital converters," Ph.D. Dissertation, University of Pennsylvania, 1977.
- [6] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. 33, no. 3, pp. 249–258, March 1985.
- [7] W. L. Lee, "A novel higher order interpolative modulator topology for high resolution oversampling A/D converters," Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1987.
- [8] K. C. H. Chao, S. Nadeem, W. L. Lee, and C. G. Sodini, "A higher order topology for interpolative modulators for oversampling A/D conversion," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 309–318, March 1990.

- [9] T. Hayashi, Y. Inabe, K. Uchimura, and A. Iwata, "A multistage delta-sigma modulator without double integration loop," *ISSCC Dig. Techn. Pap.*, pp. 182–183, Feb. 1986.
- [10] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 267–273, April 1989.
- [11] L. E. Larson, T. Cataltepe, and G. C. Temes, "Multi-bit oversampled, $\Sigma\Delta$ A/D converter with digital error correction," *Electron. Lett.*, vol. 24, pp. 1051–1052, Aug. 1988.
- [12] T. C. Leslie and B. Singh, "An improved sigma-delta modulator architecture," *Proceedings of the 1990 IEEE Int. Symp. Circuits Syst.*, vol. 1, pp. 372–375, May 1990.
- [13] G. R. Ritchie, J. C. Candy, and W. H. Ninke, "Interpolative digital to analog converters," *IEEE Trans. Commun.*, vol. COM-22, pp. 1797–1806, Nov. 1974.
- [14] J. C. Candy and A. Huynh, "Double integration for digital-to-analog conversion," *IEEE Trans. Commun.*, vol. 34, no. 1, pp. 77–81, Jan. 1986.
- [15] P. J. Naus, E. C. Dijkmans, E. F. Stikvoort, A. J. McKnight, D. J. Holland and W. Brandinal, "A CMOS stereo 16-bit D/A converter for digital audio," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 390–395, June 1987.
- [16] N. S. Sooch, J. W. Scott, T. Tanaka, T. Sugimoto, and C. Kubomura, "18-bit stereo D/A converter with integrated digital and analog filters," presented at the 91st convention of the Audio Engineering Society, New York, Oct. 1991, preprint 3113.
- [17] R. Carley and J. Kenney, "A 16-bit 4th order noise-shaping D/A converter," *IEEE Proc. Custom Integrated Circuits Conf.*, pp. 21.7.1–21.7.4, May 1988.
- [18] H. J. Schouwenaars, D. W. J. Groeneveld, C. A. A. Bastiaansen, and H. A. H. Termeer, "An oversampled multibit CMOS D/A converter for digital audio with 115-dB dynamic range," *IEEE J. Solid-State Circuits*, vol. SC-26, pp. 1775–1780, Dec. 1991.
- [19] X. Xu, G. C. Temes, and R. Schreier, "The implementation of dual-truncation SD D/A converters," *Proc., IEEE Int. Symp. Circuits Systems*, pp. 597–600, May 10–13, 1992.
- [20] T. H. Pearce and A. C. Baker, "Analogue to digital conversion requirements for HF radio receivers," *Proceedings of the IEE Colloquium on System Aspects and Applications of ADCs for Radar, Sonar and Communications*, London, Nov. 1987, Digest No. 1987/92.
- [21] P. H. Gailus, W. J. Turney, and F. R. Yester, Jr., "Method and arrangement for a sigma delta converter for bandpass signals," U.S. Patent No. 4,857,928, filed Jan. 28, 1988, issued Aug. 15, 1989.
- [22] R. Schreier and W. M. Snelgrove, "Bandpass sigma-delta modulation," *Electron. Lett.*, vol. 25, no. 23, pp. 1560–1561, Nov. 9, 1989.
- [23] D. H. Horrocks, "A second-order oversampled sigma-delta modulator for bandpass signals," *Proc. 1991 IEEE Int. Symp. Circuits Systems*, vol. 3, pp. 1653–1656, June 1991.
- [24] A. Hairapetian, "An 81MHz IF receiver in CMOS," *ISSCC Dig. Techn. Pap.*, pp. 56–57, Feb. 1996.

- [25] R. M. Gray, "Oversampled sigma-delta modulation," *IEEE Trans. Commun.*, vol. 35, no. 5, pp. 481–489, May 1987.
- [26] S. Hein and A. Zakhor, "On the stability of sigma delta modulators," *IEEE Trans. Signal Proc.*, vol. 41, no. 7, pp. 2322–2348, July 1993.

An Overview of Basic Concepts *

1.1 INTRODUCTION

This chapter reviews the main properties of oversampling techniques that are useful for converting signals between analog and digital formats. Oversampling has become popular in recent years because it avoids many of the difficulties encountered with conventional methods for analog-to-digital and digital-to-analog (A/D, D/A) conversion, especially for those applications that call for high-resolution representation of relatively low-frequency signals.

Conventional converters, illustrated in Figure 1.1, are often difficult to implement in fine-line very large scale integration (VLSI) technology. These difficulties arise because conventional methods need precise analog components in their filters and conversion circuits and because their circuits can be very vulnerable to noise and interference. The virtue of the conventional methods is their use of a low sampling frequency, usually the Nyquist rate of the signal (i.e., twice the signal bandwidth).

A low-pass filter at the input to the encoder of Figure 1.1 attenuates high-frequency noise and out-of-band components of the signal that alias into the signal when sampled at the Nyquist rate. Properties of this filter are usually specified for each application. The A/D circuit can take a number of different forms, such as flash converters for fast operation, successive-approximation converters for moderate rates, and ramp converters for slow ones. At the decoder a filter smooths the sampled output of the D/A circuit; the amount of smoothing required is usually part of the specification of the system. The circuits of these conventional converters require high-accuracy analog components in order to achieve high overall resolution.

*This chapter is a rewrite of material from reference [1].

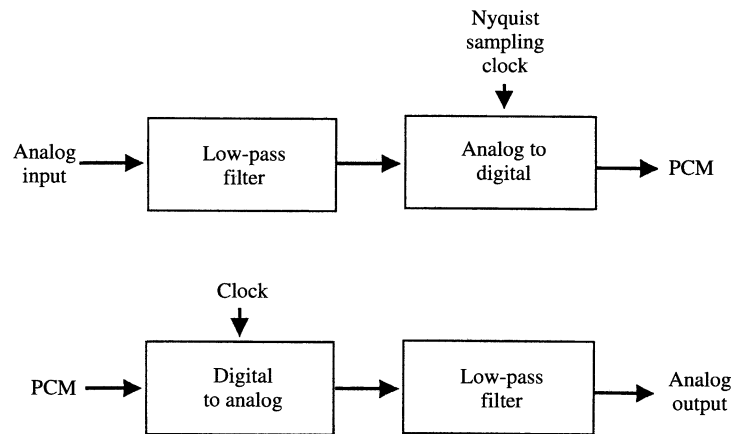


Figure 1.1 Conventional pulse code modulation (PCM), including analog filters for curtailing the aliasing noise in the encoder and for smoothing the output from the decoder.

Oversampling converters, illustrated in Figure 1.2, can use simple and relatively high-tolerance analog components to achieve high resolution, but they require fast and complex digital signal processing stages. These converters modulate the analog signal into a simple code, usually single-bit words, at a frequency much higher than the Nyquist rate.

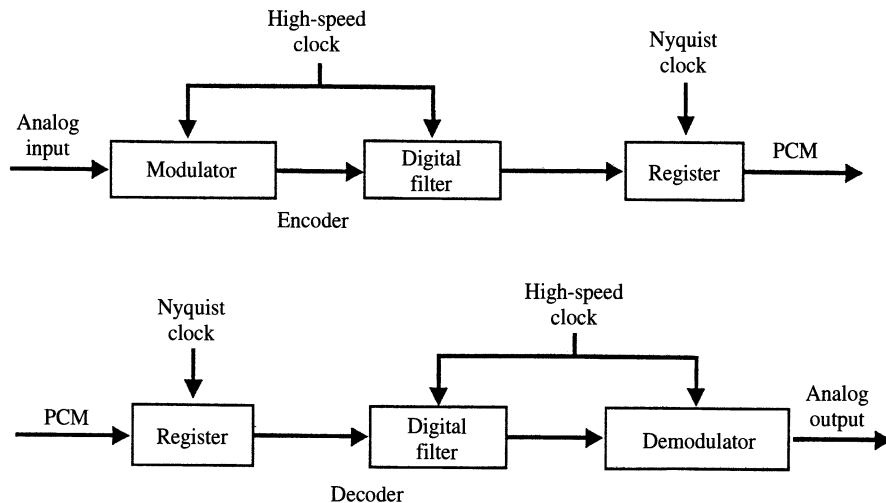


Figure 1.2 Oversampling pulse code modulation. The modulation and demodulation occur at sufficiently high sampling rate that digital filters can provide most for the antialiasing and smoothing functions.

We shall show that the design of the modulator can trade resolution in time for resolution in amplitude in such a way that imprecise analog circuits can be tolerated. The use of high-frequency modulation and demodulation eliminates the need for abrupt cutoffs in the analog antialiasing filter at the input to the A/D converter, as well as in the filters that smooth the analog output of the D/A converter. Digital filters are used instead as illustrated in Figure 1.2. A digital filter smooths the output of the modulator, attenuating noise, interference, and high-frequency components of the signal before they can alias into the signal band when the code is resampled at the Nyquist rate. Another digital filter interpolates the code in the decoder to a high word rate before it is demodulated to analog form.

Oversampling converters make extensive use of digital signal processing, taking advantage of the fact that fine-line VLSI is better suited for providing fast digital circuits than for providing precise analog circuits. Because their sampling rate usually needs to be several orders of magnitude higher than the Nyquist rate, oversampling methods are best suited for relatively low-frequency signals. They have found use in such applications as digital audio, digital telephony, and instrumentation. Future applications in video and radar systems are imminent as faster technologies become available.

An important difference between conventional converters and oversampling ones involve testing and specifying their performance. With conventional converters there is a one-to-one correspondence between input and output sample values, and hence one can describe their accuracy by comparing the values of corresponding input and output samples. In contrast there is no similar correspondence in oversampling converters because they inherently include digital low-pass filters, and hence each input sample value contributes to a whole train of output samples. Consequently, it has been useful to borrow techniques from communication technology to describe the performance of oversampling converters. Thus we measure their root-mean-square (rms) noise under various conditions, the distortion they introduce into sinusoidal signals, and their frequency responses. An important task in designing an oversampling converter is therefore the calculation of rms values of modulation noise and its spectral density. Examples of such calculations will be given in following sections.

This chapter is organized into four main sections. Following this introduction, Section 1.2 describes some basic properties of the quantization noise. It then introduces delta-sigma modulation as a technique for shaping the spectrum of quantization noise, moving most of the noise power to high frequencies, well outside the band of the signal, where it is removed by digital filtering. A number of other modulators are also described. Section 1.3 discusses the design of digital filters that decimate the modulated signal, converting it from a sequence of short digital words occurring at a high rate into long words occurring at the Nyquist rate. Section 1.4 describes oversampling D/A converters.

1.2 DIGITAL MODULATION

1.2.1 Quantization

Quantization of amplitude and sampling in time are at the heart of all digital modulators. Periodic sampling at rates more than twice the signal bandwidth need not introduce distortion, but quantization does, and our primary objective in designing modulators is to limit this distortion. We begin our discussion by describing some basic properties of

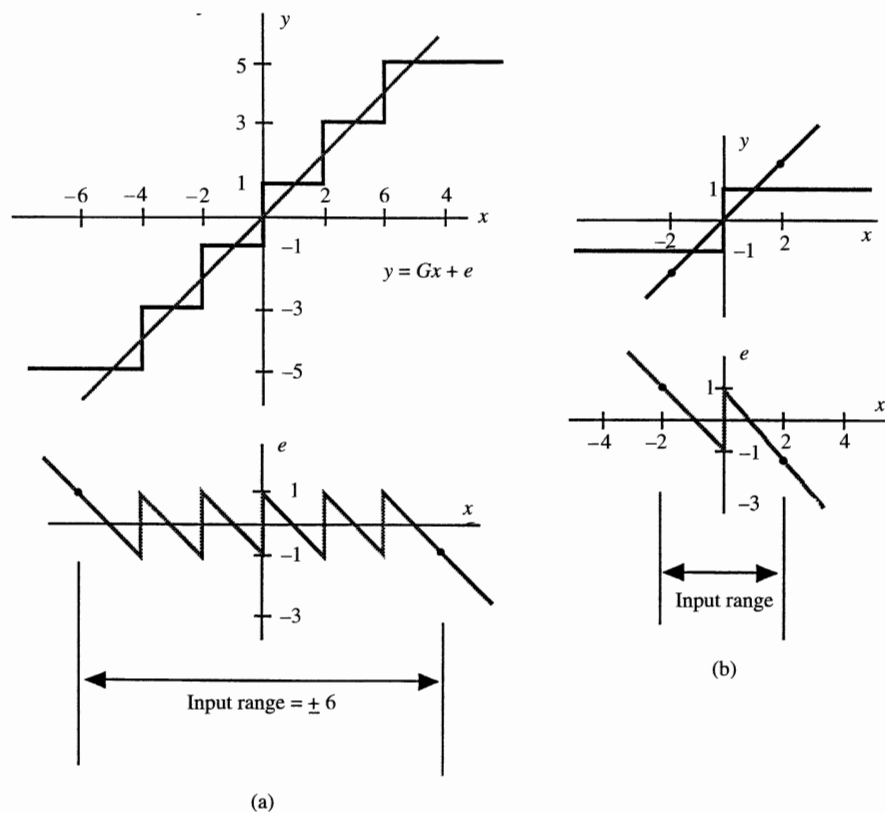


Figure 1.3 (a) An example of a uniform multilevel quantization characteristic that is represented by linear gain G and an error e . (b) For two-level quantization the gain G is arbitrary.

quantization that will be useful for specifying the noise from modulators. Figure 1.3(a) shows a uniform quantization that rounds off a continuous amplitude signal x to odd integers in the range ± 5 . In this example the level spacing Δ is 2. We will find it useful to represent the quantized signal y by a linear function Gx with an error e : that is,

$$y = Gx + e \quad (1.1)$$

The gain G is the slope of the straight line that passes through the center of the quantization characteristic so that, when the quantizer does not saturate (i.e., when $-6 \leq x \leq 6$), the error is bounded by $\pm\Delta/2$. Notice that the above consideration remains applicable to a two-level (single-bit) quantizer, as illustrated in Figure 1.3(b), but in this case the choice of gain G is arbitrary.

The error is completely defined by the input, but if the input changes randomly between samples by amounts comparable with or greater than the threshold spacing, without causing saturation, then the error is largely uncorrelated from sample to sample and has equal probability of lying anywhere in the range $\pm\Delta/2$. If we further assume that the error has statistical properties that are independent of the signal, then we can represent

it by a noise, and some important properties of modulators can be determined. In many cases experiments have confirmed these properties, but there are two important instances where they may not apply: when the input is constant, and when it changes regularly by multiples or submultiples of the step size between sample times, as can happen in feedback circuits.

When we treat the quantization error e as having equal probability of lying anywhere in the range $\pm\Delta/2$, its mean square value is given by

$$e_{\text{rms}}^2 = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} e^2 de = \frac{\Delta^2}{12} \quad (1.2)$$

For the ensuing discussion of spectral densities of the noise, we shall employ a one-sided representation of frequencies: that is, we assume that all the power is in the positive range of frequencies. When a quantized signal is sampled at frequency $f_s = 1/T$, all of its power folds into the frequency band $0 \leq f < f_s/2$. Then, if the quantization noise is white, the spectral density of the sampled noise is given by

$$E(f) = e_{\text{rms}} \sqrt{\frac{2}{f_s}} = e_{\text{rms}} \sqrt{2T} \quad (1.3)$$

We can use this result to analyze examples of oversampling modulators. Consider first ordinary pulse code modulation (PCM). A signal lying in the frequency band $0 \leq f < f_0$, to which a dither signal contained in the band $f_0 \leq f < f_s/2$ is added, is pulse code modulated at f_s . The oversampling ratio (OSR), defined as the ratio of the sampling frequency f_s to the Nyquist frequency $2f_0$, is given by the integer

$$\text{OSR} = \frac{f_s}{2f_0} = \frac{1}{2f_0 T} \quad (1.4)$$

If the dither is sufficiently large and busy to whiten and decorrelate the quantization error, the noise power that falls into the signal band will be given by

$$n_0^2 = \int_0^{f_0} e^2(f) df = e_{\text{rms}}^2 (2f_0 T) = \frac{e_{\text{rms}}^2}{\text{OSR}} \quad (1.5)$$

Thus we have the well-known result that oversampling reduces the in-band rms noise from ordinary quantization by the square root of the oversampling ratio. Therefore each doubling of the sampling frequency decreases the in-band noise by 3 dB, increasing the resolution by only half a bit.

1.2.2 Delta-Sigma Modulation

1.2.2.1 First-Order Feedback Quantizer. A more efficient oversampling quantizer is the delta-sigma ($\Delta\Sigma$) modulator shown in Figure 1.4(a). Although $\Delta\Sigma$ modulators usually employ two-level quantization, we commence our discussion by assuming the modulator contains a multilevel, uniform quantizer with unity gain $G = 1$. The input

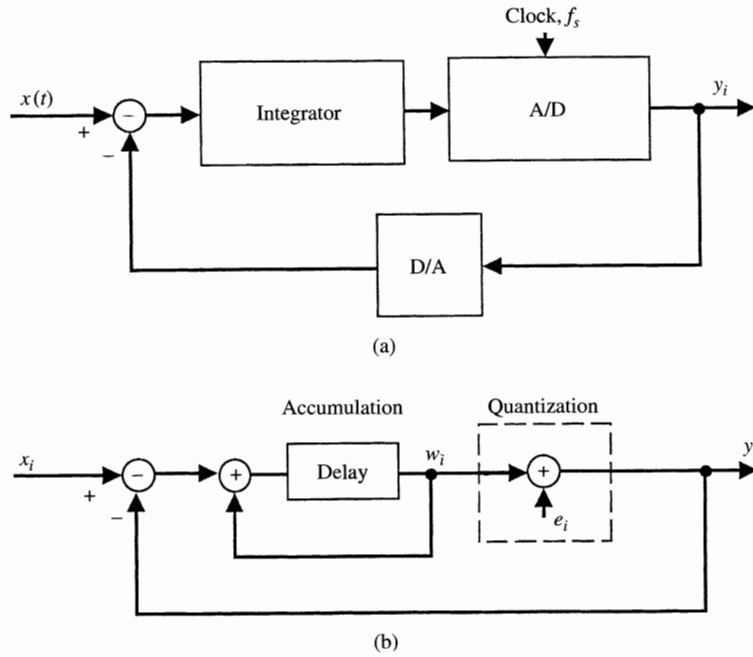


Figure 1.4 A block diagram of a $\Delta\Sigma$ quantizer and its sampled-data equivalent circuit.

to the circuit feeds to the quantizer via an integrator, and the quantized output feeds back to subtract from the input signal. This feedback forces the average value of the quantized signal to track the average input. Any persistent difference between them accumulates in the integrator and eventually corrects itself. Figure 1.5 illustrates the response of the circuit to a ramp input; it shows how the quantized signal oscillates between two levels that are adjacent to the input value in such a manner that its local average equals the average input value [2].

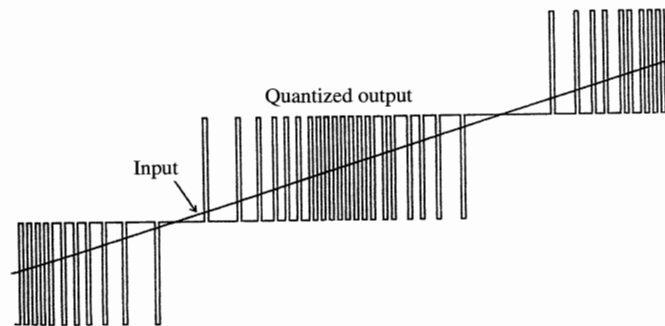


Figure 1.5 The response of a multilevel $\Delta\Sigma$ quantizer to a ramp input. A two-level response is obtained by curtailing input amplitude to a range of values that lies between two adjacent quantization levels.

1.2.2.2 Modulation Noise in Busy Signals. We analyze the modulator by means of the equivalent circuit shown in Figure 1.4(b). Here an added signal e represents the quantization error in accordance with Eq. (1.1) and the quantization gain G set to unity. Because this is a sampled-data circuit, we represent the integration by accumulation, also with unity gain. It can easily be shown that the output of the accumulator is

$$w_i = x_{i-1} - e_{i-1} \quad (1.6)$$

and the quantized signal is

$$y_i = x_{i-1} + (e_i - e_{i-1}) \quad (1.7)$$

Thus this circuit differentiates the quantization error, making the modulation error the first difference of the quantization error while leaving the signal unchanged, except for a delay.

To calculate the effective resolution of the $\Delta\Sigma$ modulator, we now assume that the input signal is sufficiently busy that the error e behaves as white noise that is uncorrelated with the signal. The spectral density of the modulation noise

$$n_i = e_i - e_{i-1} \quad (1.8)$$

may then be expressed as

$$N(f) = E(f) |1 - e^{-j\omega T}| = 2e_{\text{rms}} \sqrt{2T} \sin\left(\frac{\omega T}{2}\right) \quad (1.9)$$

where $\omega = 2\pi f$.

Figure 1.6 compares this spectral density with that of the quantization noise when the oversampling ratio is 16. Clearly, feedback around the quantizer reduces the noise at low

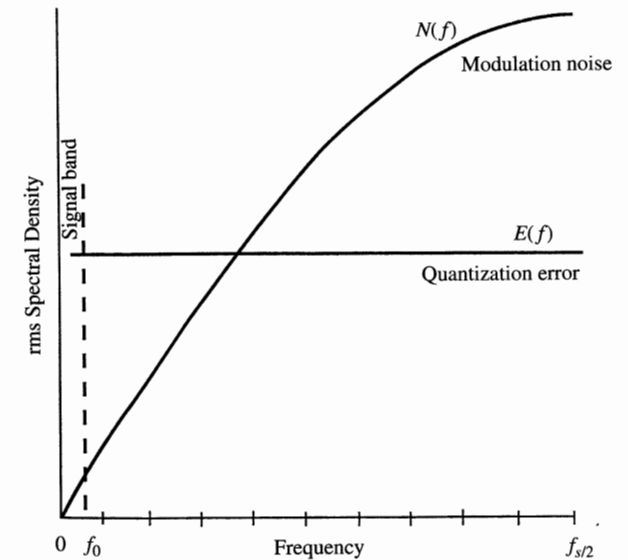


Figure 1.6 The spectral density of the noise $N(f)$ from $\Delta\Sigma$ quantization compared with that of ordinary quantization $E(f)$.

frequencies but increases it at high frequencies. The total noise power in the signal band is

$$n_0^2 = \int_0^{f_0} |N(f)|^2 df \approx e_{\text{rms}}^2 \frac{\pi^2}{3} (2f_0 T)^3 \quad f_s^2 \gg f_0^2 \quad (1.10)$$

and its rms value is

$$n_0 \approx e_{\text{rms}} \frac{\pi}{\sqrt{3}} (2f_0 T)^{3/2} = e_{\text{rms}} \frac{\pi}{\sqrt{3}} (\text{OSR})^{-3/2} \quad (1.11)$$

Each doubling of the oversampling ratio of this circuit reduces the noise by 9 dB and provides 1.5 bits of extra resolution. The improvement in resolution requires that the modulated signal be decimated to the Nyquist rate with a sharply selective digital filter. Otherwise, the high-frequency components of the noise will spoil the resolution when it is sampled at the Nyquist rate. Some early oversampling converters employed primitive decimation. One merely averaged the output samples of the modulator over each Nyquist interval to get a PCM signal. References [2] and [3] show that the rms noise in this PCM can be expressed as $\sqrt{2}e_{\text{rms}}(2f_0 T)$. They also show that taking a triangularly weighted sum over each Nyquist interval gives an rms noise $4e_{\text{rms}}(2f_0 T)^{1.5}$. An optimization of these techniques for attenuating the high-frequency noise is given in reference [4]. These decimators permit more noise to alias into the signal band than do the ones that employ filters having impulse responses that are longer than one Nyquist interval, but the techniques have been useful because their circuit implementation can be very simple.

This derivation of the average properties of modulation noise depends on representing the quantization error as white uncorrelated noise. But the analysis in Chapter 2, which does not depend on this assumption, shows that Eq. (1.11) may apply even when the error is not white. Moreover, it also shows that the quantization error is rarely truly white.

1.2.2.3 Pattern Noise from $\Delta\Sigma$ Modulation with dc Inputs. When the input to the modulator is a dc signal, the quantized signal bounces between two levels, keeping its mean value equal to the input. Figure 1.7 demonstrates that the oscillation may be repetitive; it returns to its starting condition after seven clock periods. The frequency of repetition depends on the input level; in this example the input is $3\Delta/7$ away from a level, and this results in a pattern that repeats every seven periods. When the repetition frequency lies in the signal band, the modulation is noisy, but when it does not, the modulation is quiet.

Figure 1.8 shows how the in-band rms modulation noise depends on the dc input level, for a $\Delta\Sigma$ modulator having quantization levels at ± 1 and an oversampling ratio of 16. The decimating filter that processes the modulation is the one described in Section 1.3. There are peaks of noise adjacent to integer divisions of the space between levels; elsewhere the noise is small. This structure of the quantization noise is called *pattern noise*. The largest peaks can exceed the expected noise level [Eq. (1.11)], which is at -41 dB in this example.

Surprisingly, it is also quite easy to get a mathematical expression [5, 6] for the noise from $\Sigma\Delta$ modulation with dc input. Let x be the input level to the modulator and Y' the

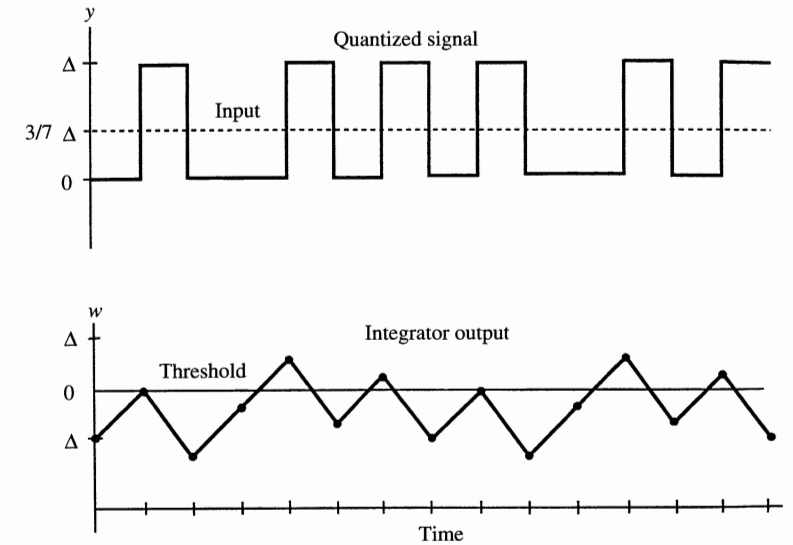


Figure 1.7 Waveforms in a $\Delta\Sigma$ circuit for a constant input situated $\frac{3}{7}\Delta$ above a quantization level.

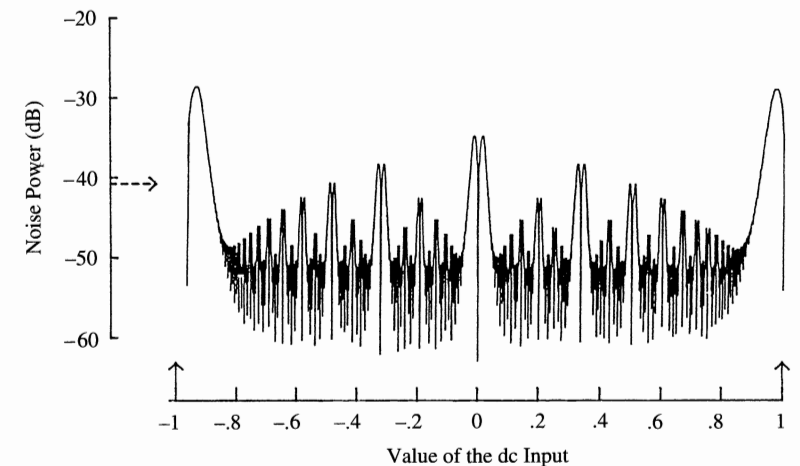


Figure 1.8 Noise from $\Delta\Sigma$ modulation for dc inputs. Quantization levels are at ± 1 , and the noise is plotted for dc inputs lying between these levels. Peaks of the noise occur adjacent to integer divisions of the level spacing.

adjacent quantization level. The modulator output can then be expressed as

$$y(t) = Y' + \sum_l \sum_k \frac{\sin(\pi l x')}{\pi l} \exp\left(j\pi + \frac{l x' + k}{T} t\right) \quad (1.12)$$

where $x' = (x - Y')/\Delta$. Thus, $y(t)$ has a dc component equal to the input x , accompanied by tones of frequency $(lx' + k)f_s$. The tones that lie in the signal band represent the inherent noise of the conversion. A sum of the powers of these tones, taking account of the response of the decimation filter, gives a good description of the pattern noise [5]. The following properties of pattern noise are noteworthy:

- The *height* of each peak is inversely proportional to the oversampling ratio.
- The *width* of each peak is inversely proportional to the oversampling ratio.
- The *power* in each peak is inversely proportional to the oversampling ratio cubed.
- The *height* and *width* of each peak are inversely proportional to the denominator of the reduced fraction that describes the position of the peak within the quantization interval relative to the level spacing. This fraction is $\frac{3}{7}$ in Figure 1.7.
- The average noise represented by the graph is given in Eq. (1.11).
- About half of the total power is in the end peaks and $\frac{1}{16}$ in the center ones.

The noise pattern in Figure 1.8 can be integrated against time as a function of a slowly changing input to get a measure of the noise introduced into a signal. Figure 1.9 shows the experimentally measured signal-to-noise ratio for two input sine waves, plotted against their amplitudes. Curve (a) corresponds to sine waves centered between two levels, and (b) corresponds to sine waves offset from center. A 0-dB input corresponds to a peak amplitude of $\Delta/2$. For comparison, the dotted line shows the values predicted from Eq. (1.11). The resolution is better than predicted when the larger peaks are not included, but it is sometimes worse when they are. The dependence of noise on signal values and the fact that the noise is composed of tones are reasons why this modulation is rarely used. When it is, dithering is usually applied to randomize the quantization noise and destroy tones that would be disturbing in audio applications, [7]. Dither will be described in Chapter 3.

1.2.2.4 Dead Zones in $\Delta\Sigma$ Modulation. The second graph in Figure 1.7 shows the output of the integrator for a steady input equal to $3\Delta/7$. Notice that this particular waveform may be raised as much as $\Delta/7$, with respect to the quantizer threshold level, without changing the sequence of decisions. Such a change of level at the output of the integrator corresponds to an impulse at its input; consequently, small fast changes of input may be ignored by the modulator, under certain conditions. It can be shown that the location and extent of the transient dead zones correspond in position and size with the peaks of noise in Figure 1.8. For most applications, the pattern noise is more noticeable than are the dead zones. However, when the integrator is a leaky one, having low dc gain, the dead zones can be significant.

We next describe some practical properties of the $\Delta\Sigma$ circuit because this simple modulator is useful for illustrating feedback quantization. Knowledge of its properties will help us to explain improved modulators.

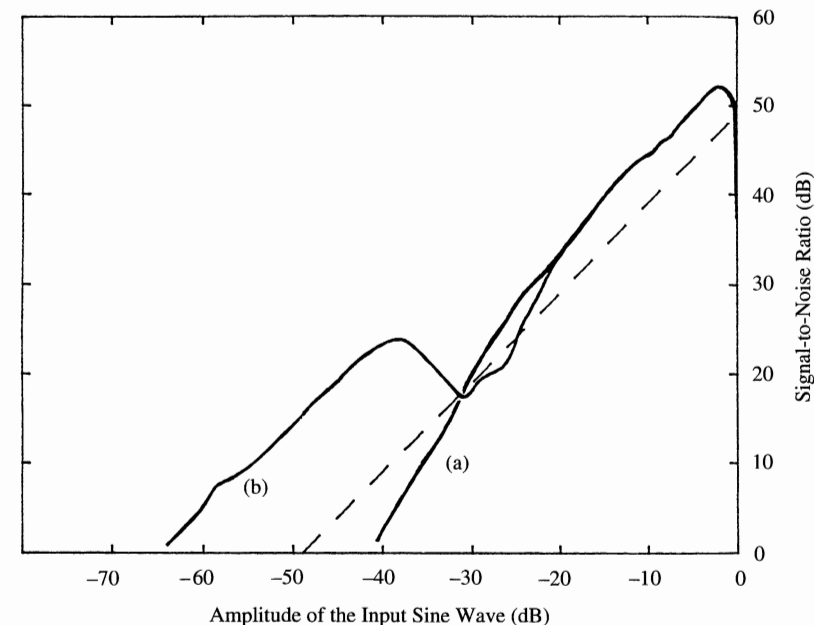


Figure 1.9 Graph of modulation noise plotted against the amplitude of applied sine waves; 0 dB corresponds to an amplitude of $\Delta/2$. Curve (a) is for sine waves centered midway between levels; curve (b) is for sine waves biased $\Delta/64$ away from center. The dashed line is the calculated noise.

1.2.2.5 Influence of Circuit Parameters on $\Delta\Sigma$ Modulation

NET GAIN IN THE FEEDBACK LOOP. Our discussion has so far assumed unity sample gain in every component of the modulator. Figure 1.10 shows a modulator that includes a constant gain G in the forward path of the feedback. Small deviations of this from unity have little effect on the overall properties, provided the net gain in the feedback loop is large. The gain of the accumulator is

$$H(f) = \frac{z^{-1}}{1 - z^{-1}} \approx (j\omega T)^{-1} = [j\pi(2fT)]^{-1} \quad fT \ll 1 \quad (1.13)$$

where $z = \exp(j\omega T)$. In the signal band this gain has a modulus greater than one quarter of the oversampling ratio [Eq. (1.4)], which is usually sufficiently large. Measurements on real modulators and simulations [2, 8, 9] have demonstrated that with small gains (i.e., $G < 0.7$), the circuit responds sluggishly to changing inputs. With gains greater than 1.3, the quantized signal bounces by more than two levels and eventually goes unstable when the gain exceeds 2, as can be predicted from the linearized model of Figure 1.4(b). For most applications 10% gain accuracy is tolerable for this circuit.

POSITIONING THE QUANTIZATION THRESHOLDS. Because of the need to have short delay, the quantizer in a multilevel $\Delta\Sigma$ modulator usually takes the form of a flash A/D.

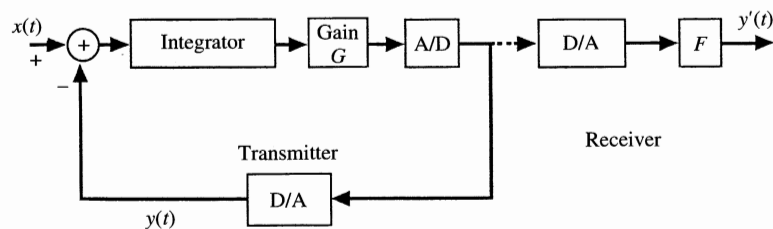


Figure 1.10 Block diagram of a $\Delta\Sigma$ modulator including a gain G in the feed-forward path and a nonlinearity F in the representative decoder.

The gain of the quantizer, defined in Eq. (1.1), is the level spacing divided by the threshold spacing, therefore misplaced thresholds may be regarded as a nonlinearity of gain. Such nonlinearity following the high gain of the integration in the forward path of the feedback loop has little effect on baseband properties of the overall modulator [2]. Misplacing the thresholds by as much as a quarter of the spacing is sometimes tolerable.

POSITIONING THE D/A QUANTIZATION LEVELS. Misplaced levels of the D/A in the feedback path of Figure 1.4(a) are more serious than misplaced thresholds because they introduce nonlinearity directly into the signal [2, 10, 11]. The feedback action forces the average value of the quantized amplitude, y , to track the input, even when levels are misplaced. If the effective D/A converter at the receiver in Figure 1.10 matches the one in the transmitter, the output y' will track y . When it does not, the mismatch can be represented as nonlinearity F at the receiver. Such nonlinearity usually must be very small, and this calls for highly accurate D/A converters [10].

TWO-LEVEL QUANTIZATION. Using two-level quantization avoids the need for matched level spacing. A misplacement of one level, as illustrated in Figure 1.11, introduces a change of quantization range and a dc offset, neither of which need be critical.

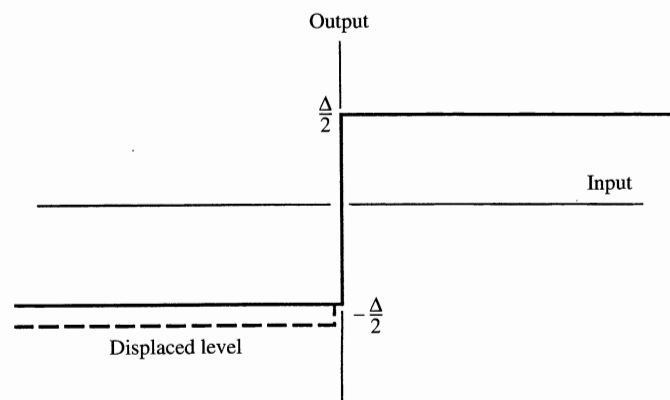


Figure 1.11 Diagram illustrating a misplaced level in two-level quantization.

Two-level quantization requires only one threshold so the concept of gain G in Eq. (1.1) is now arbitrary. Nevertheless, analysis in Chapter 2 shows that results (1.9) and (1.11) can apply.

Two-level modulators can have very robust circuits: The threshold need not be accurately positioned because it is preceded by the high dc gain of the integrator. The sample gain of the integrator is not critical because it drives a single threshold stage. The quantization levels need be positioned only to accommodate the range of input signals.

LEAKAGE IN THE INTEGRATOR. When the integrator in Figure 1.4(a) includes leakage (α), its transfer function is given by

$$H(z) = \frac{z^{-1}}{1 - \alpha z^{-1}} \quad (1.14)$$

and its dc gain by

$$H_0 = H(1) = \frac{1}{1 - \alpha} \quad (1.15)$$

The output of the modulator can be expressed as

$$Y = \frac{z^{-1}X}{1 + (1 - \alpha)z^{-1}} + \frac{(1 - \alpha z^{-1})E}{1 + (1 - \alpha)z^{-1}} \quad (1.16)$$

There is increased noise at low frequency, as illustrated schematically in Figure 1.12. If the dc gain of the integrator is at least equal to the oversampling ratio, the increase in baseband noise is less than 0.3 dB. This condition also ensures that the dead zone described in Section 1.2.2.4 is not troublesome; it could be with smaller gains.

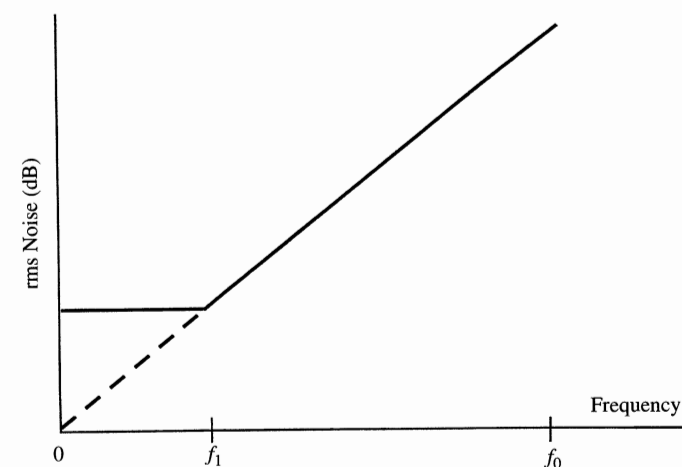


Figure 1.12 Illustration of the effects of leakage in the integrator on the spectral density of modulation noise.

1.2.3 High-Order Modulation

1.2.3.1 Predicting In-Band Values of Quantization Error. In a $\Delta\Sigma$ circuit, feedback via an integrator shapes the spectrum of the modulation noise, placing most of its energy outside the signal band. In general, the characteristics of the filter included in the feedback loop determine the shape of the noise spectrum [12]. In this section we discuss a number of filters and circuit structures that are improvements on ordinary $\Delta\Sigma$ circuits.

The objective of using improved noise shaping filters is to reduce the net noise in the signal band. To do this well, we need to subtract from the quantization error a quantity whose in-band component is a good prediction of the in-band error. Ordinary $\Delta\Sigma$ modulation subtracts the previous error [Eq. (1.7)]. Higher order prediction should give better results than this first-order prediction.

1.2.3.2 Noise in High-Order $\Delta\Sigma$ Modulation. We will see that there are several circuit arrangements that give second-order predictions of the quantization error and that the one shown in Figure 1.13 is easy to build and is tolerant of circuit imperfection. It is an iteration of $\Delta\Sigma$ feedback loops. The output of this modulator can be expressed as

$$y_i = x_{i-1} + (e_i - 2e_{i-1} + e_{i-2}) \quad (1.17)$$

so that the modulation noise is now the second difference of the quantization error. The spectral density of this noise is

$$N(f) = E(f) \left(1 - \varepsilon^{-j\omega T}\right)^2 \quad (1.18)$$

For busy signals

$$|N(f)| = 4e_{\text{rms}}^2 \sqrt{2T} \sin^2\left(\frac{\omega T}{2}\right) \quad (1.19)$$

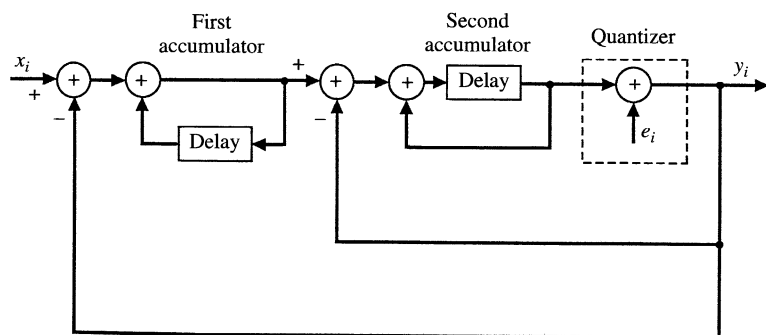


Figure 1.13 Second-order $\Delta\Sigma$ quantizer.

and the rms noise in the signal band is given by

$$n_0 \approx e_{\text{rms}} \frac{\pi^2}{\sqrt{5}} (2f_0 T)^{5/2} = e_{\text{rms}} \frac{\pi^2}{\sqrt{5}} \text{OSR}^{-5/2} \quad f_s^2 \gg f_0^2 \quad (1.20)$$

This noise falls by 15 dB for every doubling of the sampling frequency, providing 2.5 extra bits of resolution [8, 13].

The technique can be extended to provide higher order predictions by adding more feedback loops to the circuit [8]. In general, when a modulator has L loops and is not over-loaded, it can be shown that the spectral density of the modulation noise is

$$|N_L(f)| = e_{\text{rms}} \sqrt{2T} \left[2 \sin\left(\frac{\omega T}{2}\right) \right]^L \quad (1.21)$$

For oversampling ratios greater than 2, the rms noise in the signal band is given approximately by

$$n_0 = e_{\text{rms}} \frac{\pi^L}{\sqrt{2L+1}} (2f_0 T)^{L+1/2} \quad (1.22)$$

This noise falls $3(2L-1)$ decibels for every doubling of the sampling rate, providing $(L - \frac{1}{2})$ extra bits of resolution, but we shall see that there are difficulties in implementing circuits containing more than two integrators. Figure 1.14 plots the in-band noise against

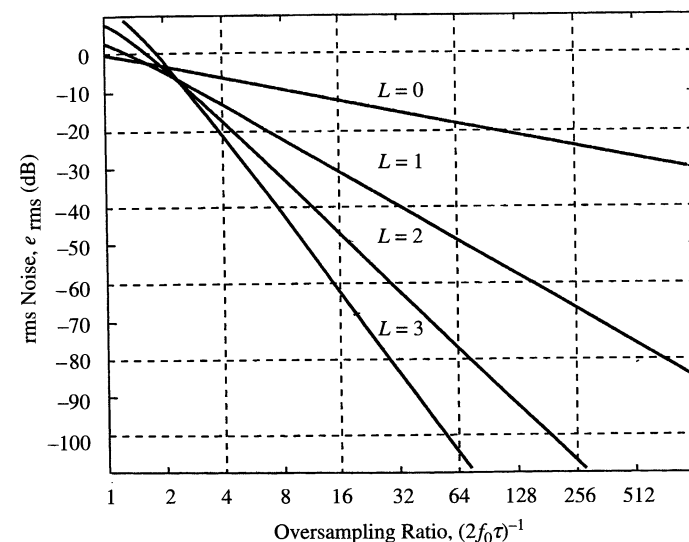


Figure 1.14 The rms noise that enters the signal band for oversampling ratios in the range 1 through 512, assuming busy input signals. Graphs are plotted for ordinary quantization without feedback $L = 0$, and first-, second-, and third-order $\Delta\Sigma$ quantization. 0 dB of noise corresponds to that of PCM sampled at the Nyquist rate. A common level spacing is used in all the quantizers.

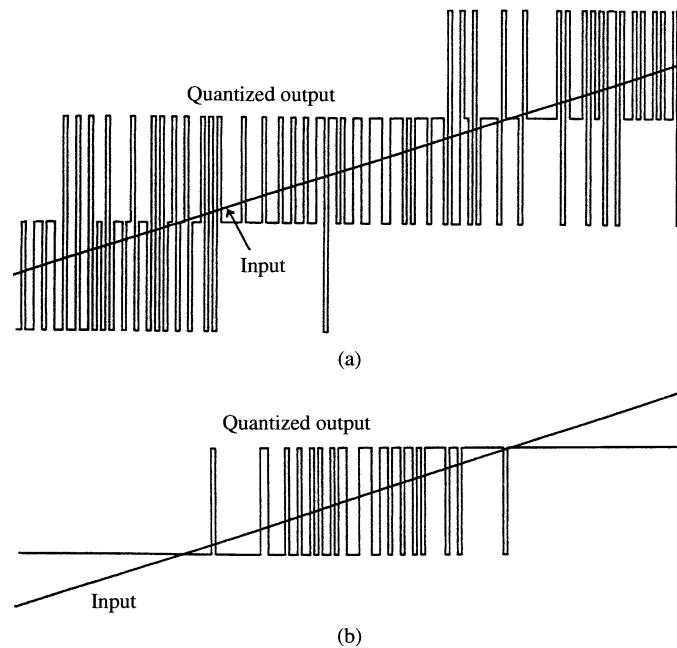


Figure 1.15 Response of a second-order $\Delta\Sigma$ quantizer to a ramp input for both multilevel and two-level quantization.

the oversampling ratio for examples of PCM, and modulators with one, two, and three feedback loops. These graphs are derived from result (1.21), which assumes white, uncorrelated quantization error, and this may not be valid unless the signal is sufficiently busy to randomize the error or unless sufficient dithering is included. It is fortunate that the noise in second- and higher order circuits is more random than it is in the first-order ones [8]. The randomizing influence is provided by the retention of noise in the integrators and depends on the circuits having long-term memory.

Figure 1.5 illustrates the output of the first-order modulator oscillating between two levels adjacent to the input value. The noise ranges in amplitude between $\pm\Delta$, which is consistent with Eq. (1.8). The output of the second-order modulator oscillates predominantly between three levels, but occasionally reaches a fourth, which is consistent with the expression for the noise in Eq. (1.17). Figure 1.15(a) shows the output of a second-order modulator having quantization levels at integer values, when responding to a ramp, and Table 1.1 lists its output for a steady input of 1.3. The output includes levels 0, 1, 2, and sometimes 3 in seemingly random order, but keeping its average close to the input value. These measurements commenced with arbitrary initial values in the integrators. Starting with integer values results in repetitive patterns in the output.

1.2.3.3 Dynamic Range of the Modulators. The oscillation of the signal uses up some of the dynamic range of the circuit, and if the quantizer is not to overload the

TABLE 1.1 A STRING OF 200 OUTPUT SAMPLES FROM A SECOND-ORDER DELTA-SIGMA MODULATOR THAT QUANTIZES ANALOG VALUES INTO INTEGERS*

0	2	1	2	1	1	2	1	1	2	0	2	2	0	2	2	0	2	1	2	1	1	1	2	1	1	2
1	1	2	1	1	1	2	1	2	0	2	2	1	2	0	3	0	2	1	1	2	2	0	2	1	1	2
1	1	2	1	2	0	2	1	2	1	1	2	0	2	2	0	2	2	1	1	2	1	1	1	1	2	1
1	2	1	1	2	1	1	2	1	2	0	2	2	1	2	1	2	1	1	2	1	1	2	2	1	1	1
2	1	1	1	2	1	2	0	2	2	1	2	0	2	2	1	2	2	1	1	2	1	1	2	1	1	1
2	1	1	2	1	1	2	1	1	2	1	2	0	2	2	1	2	1	1	1	2	1	1	2	1	1	1
2	1	1	2	1	1	2	1	1	2	1	2	0	2	2	1	2	1	1	1	2	1	1	2	0	3	0

*The input to the modulator is a constant 1.3 value, and the measurement commences with arbitrary initial conditions.

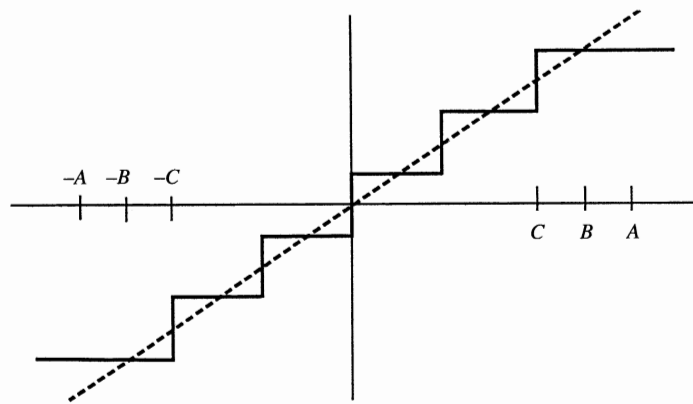


Figure 1.16 Range of amplitudes that can be accommodated by multilevel quantizers. Ordinary quantization accommodates input in the range $\pm A$, and first-order $\Delta\Sigma$ quantization accommodates $\pm B$. Second-order $\Delta\Sigma$ accommodates $\pm C$ with small probability of overloading.

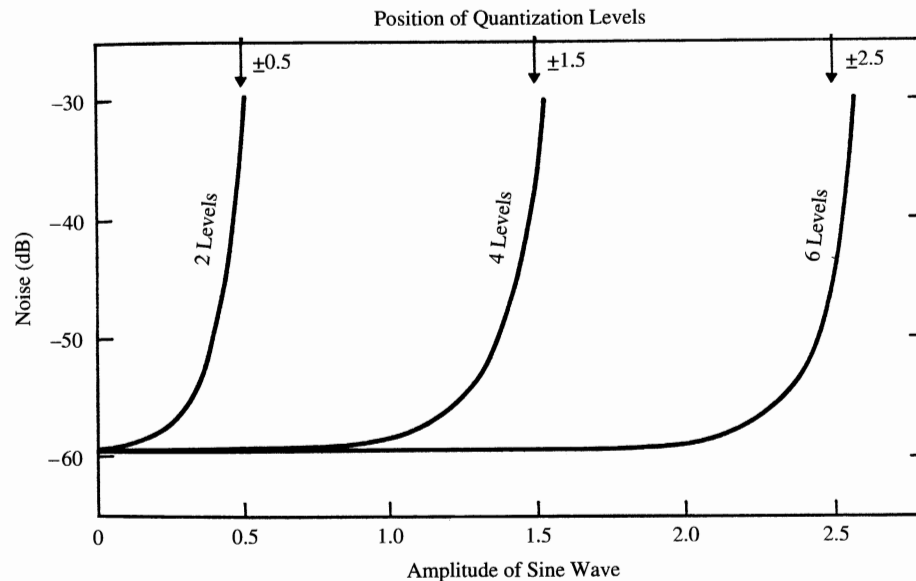


Figure 1.17 Noise introduced into sine waves of various amplitudes by second-order $\Delta\Sigma$ quantization with either 2, 4, or 6 quantization.

input amplitude to the modulator needs to be limited. Figure 1.16 shows the range of inputs that can be accommodated in several modulators, each employing six-level quantization. Ordinary PCM requires that its input be restricted to $\pm A$ in order that the quantization error lie in the range $\pm\Delta/2$. Inputs to a corresponding first-order $\Delta\Sigma$ modulator need be restricted to $\pm B$ for them to be interpolated by oscillation between two levels. Inputs to a second-order modulator need be restricted to $\pm C$ to prevent frequent overloading of its quantizer. This permits the output to oscillate between three levels, but overload can occur occasionally when the oscillation attempts to step outside this three-level range.

When large inputs cause the quantizer to overload, the modulation noise increases as illustrated in Figure 1.17. This plots the noise introduced into sine waves of various amplitude, during second-order modulation. Graphs are drawn for three cases: two-, four-, and six-level quantization. The levels are positioned at ± 0.5 , ± 1.5 , and ± 2.5 . The results demonstrate that the *excess noise* due to overloading the quantizer increases quite slowly with increasing amplitude. Even two-level quantization has a useful range, despite the fact that theoretically it is overloaded for all conditions, except zero input with zero initial conditions. The response of a second-order modulator having two-level quantization to a ramp input is shown in Figure 1.15(b). Examples of such modulators are described in later chapters.

For small input amplitudes, the noise in these modulators agrees with Eq. (1.20). Simulations show that the excess noise introduced into larger inputs appears as odd-order harmonic distortion of centrally biased sine waves and includes a minute increase in the gain of the fundamental. The excess noise decreases with increased oversampling ratio but increases with the frequency of the applied sine wave. A complete characterization of the excess noise is not yet available, but attempts have been made to analyze it [14].

1.2.3.4 Influence of Circuit Parameters on Second-Order Modulators

CIRCUIT TOLERANCES. The second-order modulator in Figure 1.13, like the first-order one in Figure 1.4, is very tolerant of circuit imperfections, especially when two-level quantization is employed. Compared with first-order systems, the second-order system has one more design parameter available; it is the ratio of the gains of the two feedback paths. The outer path dominates in determining the low-frequency properties of the circuit, while the inner path serves to stabilize the system, and determines high-frequency properties. Matching their relative gains to within $\pm 5\%$ is usually satisfactory. Sometimes, the gain of the inner loop is deliberately increased to compensate for delay in the outer loop [15].

RANGE OF INTEGRATION. An important parameter is the range of signal amplitudes that must be accommodated at the outputs of the integrators. Simple theory gives an adequate description of these signals for multilevel quantization that does not saturate. The output of the first integrator is given by $x_i - e_i + e_{i-1}$, the second by $x_{i-1} - 2e_{i-1} + e_{i-2}$, with e and x bounded by $\pm\Delta/2$. These results are inadequate for describing two-level modulators, and we resort to simulations. Figure 1.18 plots the maximum signal level at the output of the integrators as a function of a dc input level. The signal in the first integrator remains well bounded, but the second one becomes very large as the input exceeds 0.4Δ . Clipping this signal at about three times the range of the input signal has little effect on overall performance of the modulator [8, 15].

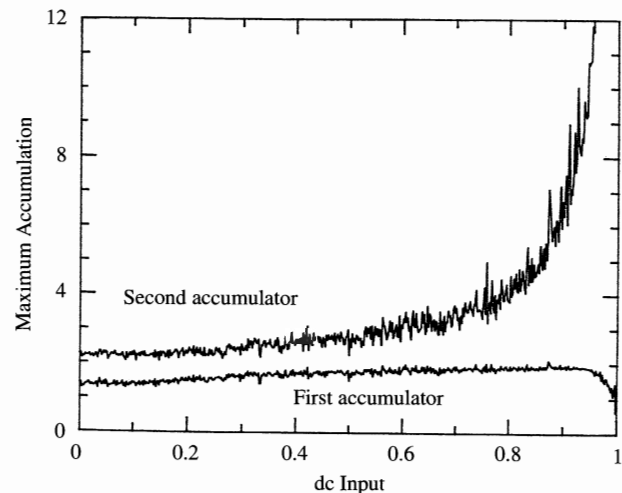


Figure 1.18 Maximum amplitude of signals in the accumulators of a second-order $\Delta\Sigma$ modulator. The two quantization levels are at ± 1 . In practice the first accumulation is often clipped at ± 2 and the second effectively at ± 4 .

LEAKAGE IN THE INTEGRATORS. First-order modulators need integrators with dc gains H_0 that are greater than the oversampling ratio, in order to have low noise. Calculations of noise in second-order modulators indicate that somewhat lower gains could be tolerated because the gains of two integrator amplifiers are cascaded in the outer loop. But there is another consideration: Leakage can permit the oscillation of the quantized signal to settle into regular patterns when there is insufficient long-term memory to randomize it. This is most noticeable at the center of the range where the output can settle into a $+1, -1, +1, -1$ pattern. The effect is illustrated by Figure 1.19, which shows the filtered output of a modulator responding to a very slowly changing ramp. The full range of the output signal is ± 1 V; Figure 1.19 has such an expanded scale that the noise is apparent. At the center of the range the output locks into the pattern and the input is ignored in the range ± 0.2 mV. It may be shown that the width of the dead zone is given approximately by $1.5\Delta H_0^{-2}$, and for this to be less than twice the rms noise requires that the dc gain of each integrator satisfy

$$H_0 \geq (2f_0 T)^{-5/4} \quad (1.23)$$

The dead zone is seldom noticeable because it is present only in very slowly changing signals: It takes time for the oscillations to settle into a pattern. Such a dead zone could actually be useful for audio applications, which need a very quiet idle state.

1.2.3.5 Limit Cycles in Third-Order $\Delta\Sigma$ Modulators. Simple linear theory predicts that the third-order modulator shown in Figure 1.20 has an rms noise given by Eq. (1.22) with $L = 3$. This can be realized in practice with a multilevel quantizer that does not overload [8]; but the circuit is much more sensitive to circuit values than the first-

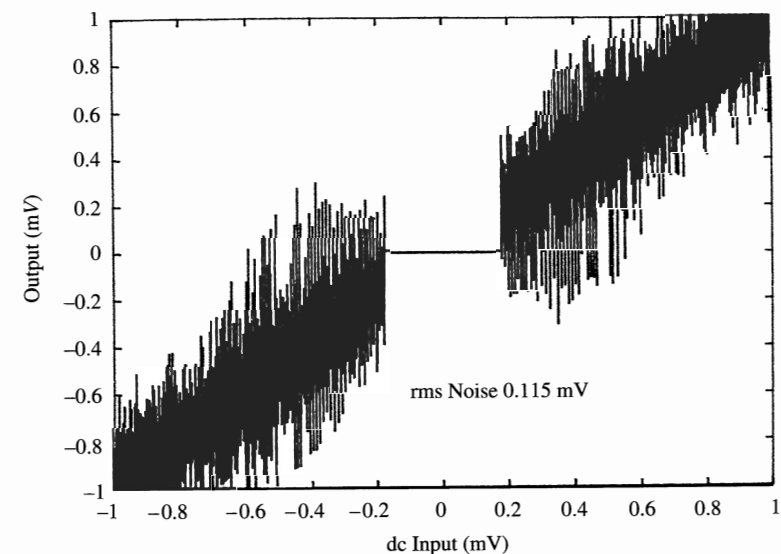


Figure 1.19 Illustration of the dead zone caused by leakage in the accumulators of second-order $\Delta\Sigma$ quantization. The dc gain of each accumulator is 64, and the oversampling ratio is also 64. The range of input and output amplitudes that can be accommodated is ± 1 V, and the noise is less than that of 12-bit PCM.

and second-order ones. For example, the equivalent linear circuit of this modulator becomes unstable with quantizer gains G in excess of 1.15 compared with 2.0 and 1.33 for first- and second-order modulators.

More seriously, the third-order circuit is also unstable when its quantizer gain falls below 0.3. When the quantizer saturates, its effective gain falls, and this usually results in

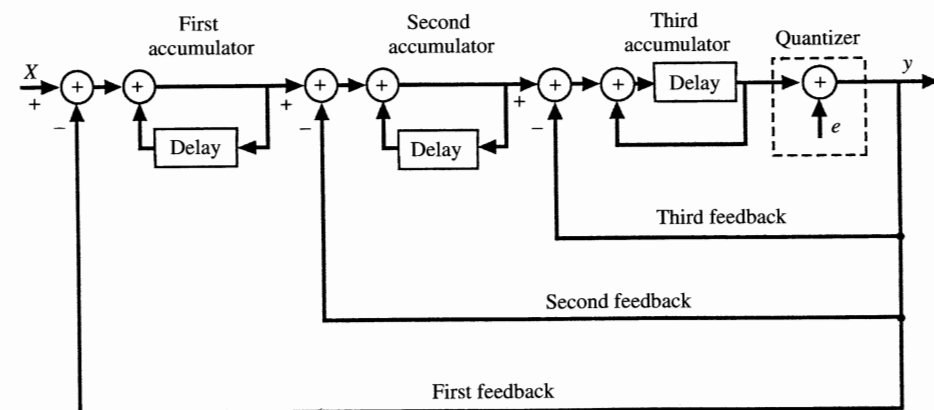


Figure 1.20 Third-order $\Delta\Sigma$ quantizer.

an instability in which the circuit settles into a large-amplitude low-frequency limit cycle. In this state the clipped signals fed back via the two inner feedback paths are small compared with the signals emerging from the integrators. Properties of the outer loop dominate; it contains three integrators and a delay, a strong basis for instability. Unembellished, two-level, third-order $\Delta\Sigma$ modulators cannot escape from this condition. Their circuits can be made stable by clipping the outputs of the integrators or including other nonlinearities that make the inner feedback effective when the quantizer saturates. The noise performance of these modified modulators is considerably worse than Eq. (1.21) predicts. Better performance is obtained by redesigning the filter used in the feedback loop.

1.2.3.6 Noise Shaping Using Filters with Nonmonotonic Transfer Functions. The $\Delta\Sigma$ modulators described so far contain filters in their feedback loop that have multiple poles at dc and zeros at high frequency to stabilize the circuits. The frequency responses of these filters fall monotonically through the range 0 to $f_s/2$.

The noise at the output of the modulator is shaped approximately as the inverse of the filter characteristic. Later chapters describe modulators that replace these filters with a more rectangular high-pass filter. The poles are distributed through the signal band in order to lower the in-band noise. The zeros are chosen to flatten the filter response at high frequency in order to reduce the high-frequency noise and prevent it from using up dynamic range. A noise spectrum [16] obtained from such a modulator is given in Figure 1.21.

Modulators with two-level quantization and fourth- and fifth-order filters have been successfully built with this technique. Their circuits, illustrated in Figure 1.22, are based

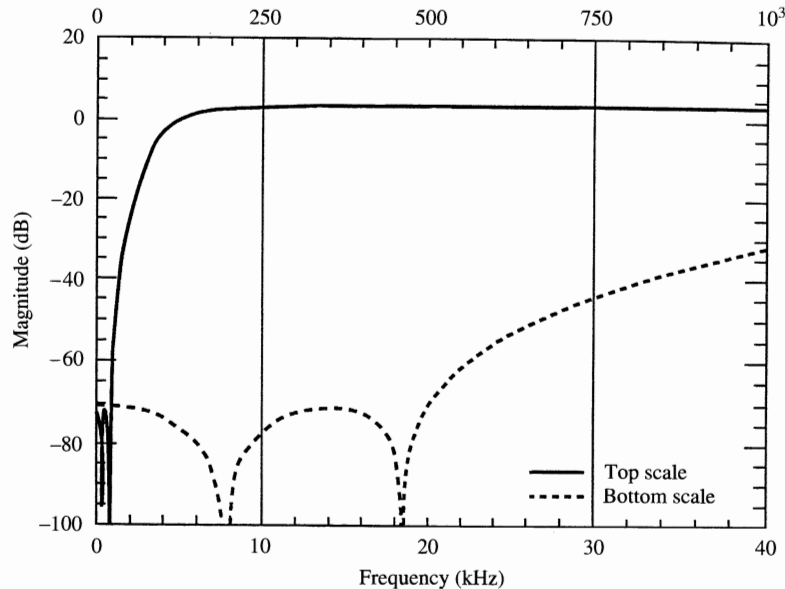


Figure 1.21 Spectral densities of the noise from a generalized feedback of the type shown in Figure 1.22.

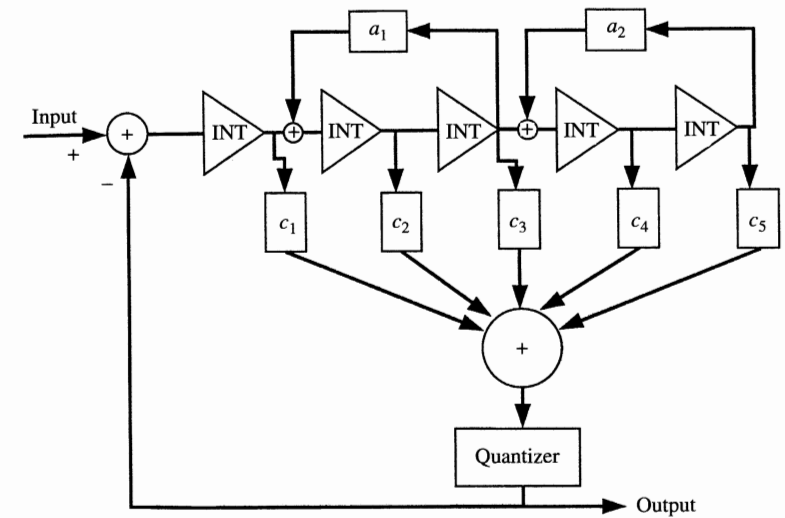


Figure 1.22 Fifth-order feedback quantizer. The loop gain includes two complex and one real pole with zeros positioned to ensure stability. The quantization is two level.

on cascaded integrators [16] with feedback branches dimensioned to position the poles and feedforward ones dimensioned to position the zeros. The danger of these circuits locking into high-frequency limit cycles is avoided by allowing the integrators to clip at quite small amplitudes. The input amplitudes are limited to less than $\pm\Delta/4$ to avoid distorting the signal. Alternative structures are described in other chapters of this book.

The noise performances of these modulators are poorer than anticipated by Eq. (1.21), but better than that obtained from second-order $\Delta\Sigma$ modulators. For example, 16-bit encoding of 20-kHz signals has been obtained by using a fourth-order modulation at 3 MHz.

1.2.4 Some Alternative Modulator Structures

1.2.4.1 Error Feedback. Noise-shaping quantization was first introduced using the structure shown in Figure 1.23. In this circuit, the difference between the input

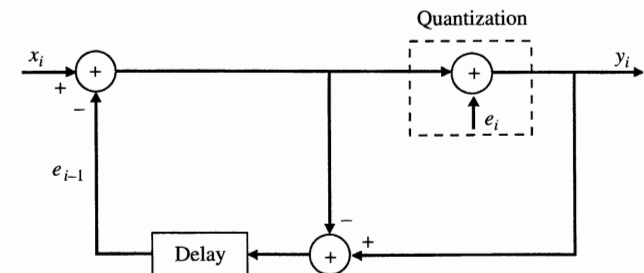


Figure 1.23 Quantizer with error feedback.

and the output of the quantizer is a measure of the quantization error, which is fed back and subtracted from the next input sample. The circuit is algebraically equivalent to the $\Delta\Sigma$ circuit in Figure 1.4, but it has the serious practical disadvantage that inaccuracies in the analog subtractors have a strong impact on the modulator's properties. We shall see, however, that the circuit can be used as a demodulator because there the processing is performed digitally. The circuit can be generalized by replacing the delay with a prediction filter, design methods for which are given in [12].

1.2.4.2 Cascaded Modulators. The performance of a modulator can be improved by taking a measure of its noise, digitizing that measure in a second modulator, and combining the output of the two modulators in a way that cancels the noise of the first modulator. This technique was proposed for use with two $\Delta\Sigma$ modulators and has since been widely applied to $\Delta\Sigma$ modulators [17]. Figure 1.24 shows a method for cascading two first-order $\Delta\Sigma$ modulators. The output of the integrator in the first modulator is fed to the second modulator. Its output is digitally differentiated and subtracted from the output of the first modulator to provide the net output of the circuit. We have used e' to denote the quantization error in the first modulator and e that of the second one. When scaling factors are ignored, it can be shown that the net output of the circuit may be expressed in the form

$$y_i = x_{i-2} + (1-g)(e'_{i-2} - e'_{i-3}) + (e_i - 2e_{i-1} + e_{i-2}) \quad (1.24)$$

where g is a measure of the accuracy of the error cancellation. It depends on a number of parameters, including the precision of component values and the low-frequency gain of the first integrator. Ideally, g is unity; then the noise of the first modulator does not contribute to the output. The remaining noise is the second difference of the quantization error

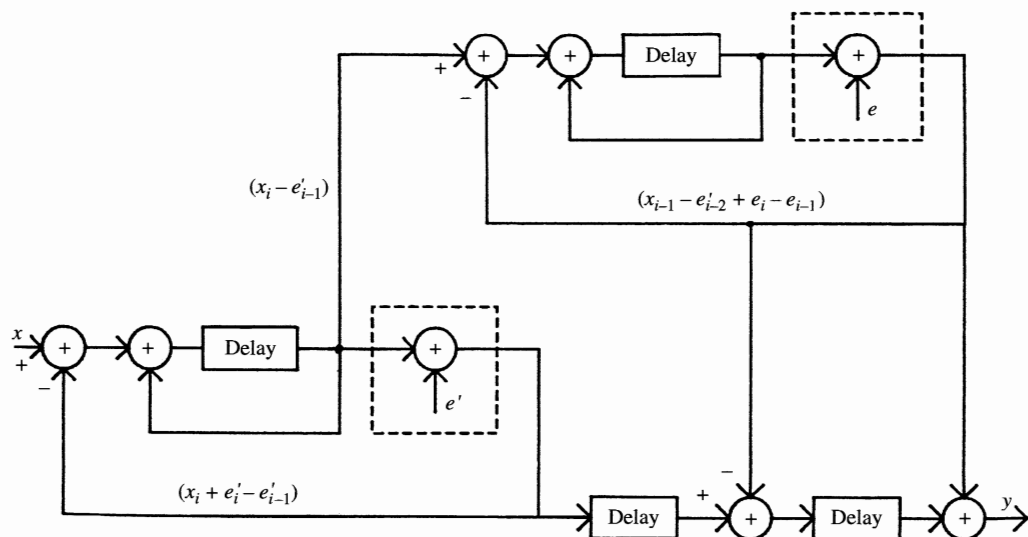


Figure 1.24 Cascade of two first-order $\Delta\Sigma$ modulators. The second modulator serves to digitally encode the quantization error e' of the first modulator so that e' may be cancelled from the net output.

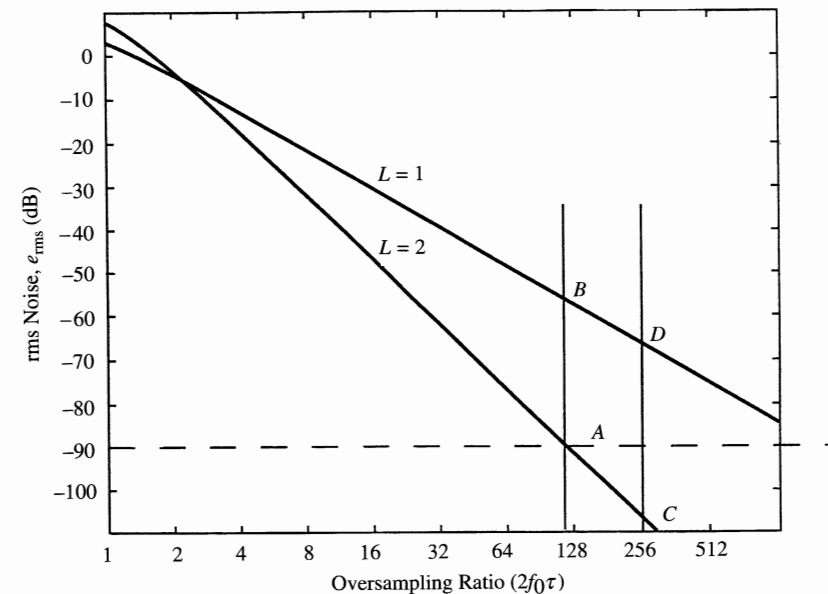


Figure 1.25 A graphical comparison of noise sources in the cascaded modulators illustrated by Figure 1.24.

from the second modulator: It is in the same form as the noise of a second-order $\Delta\Sigma$ modulator given in Eq. (1.17).

The following question is a first consideration in designing cascaded modulators: How close to unity must the factor g be? This can be determined from Figure 1.25, which was derived from Figure 1.14. As an example, suppose we are designing a modulator similar to the one in Figure 1.24 to provide resolution equivalent to 15-bit PCM, which corresponds to a noise of -90 dB on the ordinate. An ideal second-order oversampling by a factor of 120, represented by point A on the graph would meet the requirement. At this sampling rate the noise from the first-order modulation is given by the ordinate of point B. It is at -57 dB. We therefore need sufficient precision to make the term $(1-g)$ much less than $57 - 90 = -33$ dB; that is, g needs to be well within 2% of unity. In practice this requirement is tightened by needs to scaling signal amplitudes in practical circuits. We can be looking for component tolerances below 0.1% and amplifier gains in excess of 10,000.

The need for such precision is alleviated by raising the sampling rate: for example, at an oversampling ratio of 256, corresponding to points C and D on the graph. The second-order noise is now at -108 dB, well below the requirement. The first-order noise is -68 dB, and we need to reduce this by only 22 dB (i.e., keep g within 8% of unity) to achieve a net noise of -90 dB.

Because of the difficulty in obtaining adequate precision, the noise from these cascaded circuits is often dominated by the noise from the first stage, which in our example is first order and will include peaks of pattern noise. This difficulty is avoided by using a second-order modulator for the first stage [18, 19]. When the performance is constrained

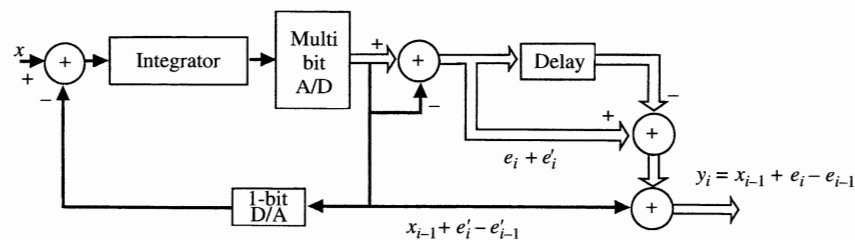


Figure 1.26 A delta modulator and its sampled-data equivalent circuit.

by circuit imperfection, there is little advantage in using higher than first-order modulation as the second stage or adding a third stage.

When circuits have sufficient precision to eliminate the noise of the first stage from the output, the cascade modulator has several attractive features. For example, when $(1 - g) = 0$, the circuit in Figure 1.24 provides second-order modulation yet the feedback loops are first order with two-level quantization. The output can oscillate between four levels, illustrated in Figure 1.15(a), and there need be no excess noise caused by quantizer overload provided the first-order input stage does not overload. Adding a third stage to the cascade can provide third-order modulation without incurring the dangers of instability associated with third-order feedback quantization.

An ingenious circuit [20] that can be interpreted as a cascade of a two-level first-order $\Sigma\Delta$ modulator with multilevel PCM is shown in Figure 1.26. Only the sign bit of the A/D drives the feedback. The complete digital word from the A/D is reduced by subtracting the value of its sign bit (this is equivalent to inverting the sign bit of a 2's complement code). The first difference of the resulting code adds to the original sign bit to provide the output. This technique can obviously be extended to higher order modulators.

1.2.4.3 Delta Modulation. Most early work on oversampling was concerned with Δ modulation. Later work turned its attention to $\Delta\Sigma$ modulation because its circuits are more robust. The main difference between the techniques is that $\Delta\Sigma$ modulators, and other noise-shaping modulators, change the spectrum of the noise but leave the signal unchanged. By contrast, Δ modulation and other signal-predicting modulators shape the spectrum of the modulated signal but leave the quantization noise unchanged at the receiver. It is the need for a filter at the receiver to restore the signal that makes signal-predicting modulators vulnerable to analog circuit inaccuracy. These filters usually have high gain in the signal band and thus magnify distortion introduced in the channel or the D/A.

Figure 1.27 is a diagram of a Δ modulator and demodulator. It transmits the first difference of the signal; consequently an integrator is needed at the receiver. The output is contaminated with the quantization error itself, and it saturates by clipping the derivative of the signal (slope overloading). In contrast the output of a $\Delta\Sigma$ modulator is contaminated by the first difference of the quantization error and saturates by clipping amplitudes. To compare these modulators, we now calculate their signal-to-noise ratios for sinusoidal inputs. The largest sine wave that the $\Delta\Sigma$ modulator can accommodate without saturating has peak value $\Delta/2$. Its rms noise is given by Eq. (1.11), and therefore the maximum rms

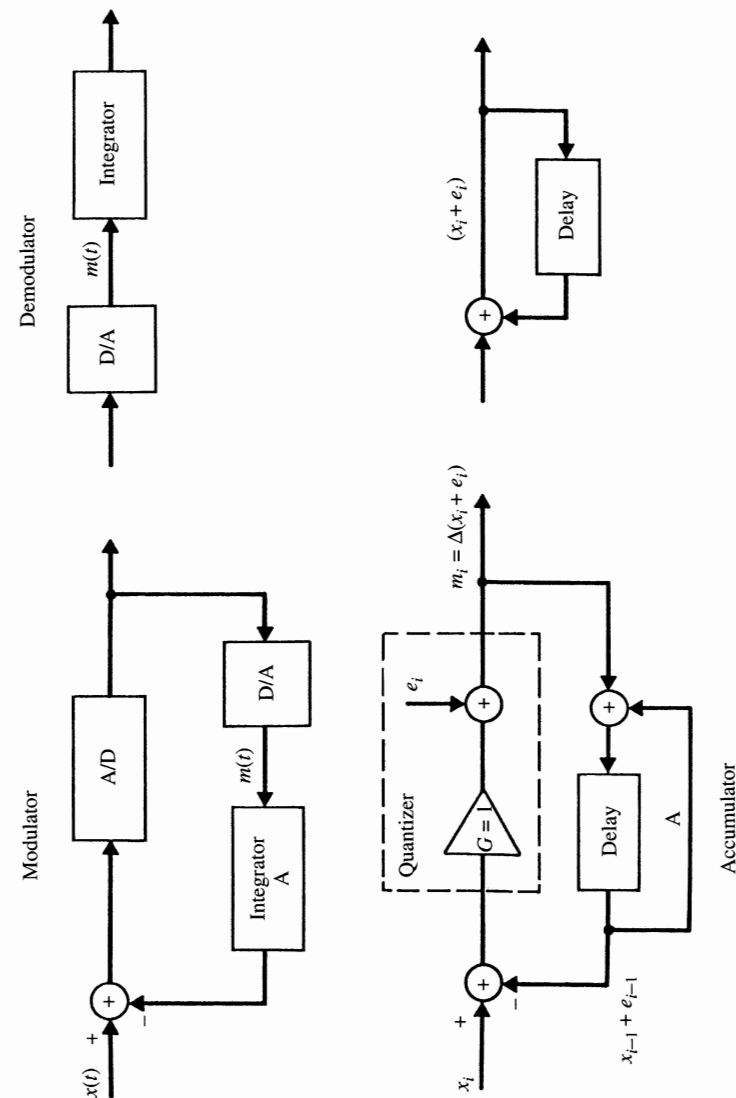


Figure 1.27 A delta modulator and demodulator with its sampled-data equivalent circuit.

signal-to-noise ratio (SNR) can be expressed as

$$\text{SNR}_{\text{dsm}} = \sqrt{\frac{3}{8}} \frac{\Delta}{\pi e_{\text{rms}}} (2f_0 T)^{-3/2} = \frac{\sqrt{4.5}}{\pi} (2f_0 T)^{-3/2} \quad (1.25)$$

The peak value of the largest sine wave that a delta modulator will accommodate is slope limited and is given by $\Delta/\omega T$, where ω is the angular frequency of the sinusoid signal. If f' is the frequency at which the signal source delivers its steepest slope, the amplitude of sinusoidal inputs need be constrained to be less than $\Delta/(2\pi f' T)$. The rms inband noise introduced into the signal is given by (3). Therefore the signal-to-noise ratio is given by

$$\text{SNR}_{\text{dm}} = \frac{\sqrt{6}}{\pi} \left(\frac{f_0}{f'} \right) (2f_0 T)^{-3/2} \quad (1.26)$$

For general signals $f' = f_0$ then Δ modulation has only a 1-dB advantage over $\Delta\Sigma$ modulation. For some speech signals $f' = f_0/3$. Then Δ modulation has a 10-dB advantage. The Δ modulation and signal predictive modulators can be useful for applications where f' is much less than f_0 and when clipping slopes is more tolerable than clipping amplitudes, as in some audio and video application.

1.3 DECIMATING THE MODULATED SIGNAL

1.3.1 Multistage Decimation

The output of the modulator represents the input signal together with its out-of-band components, modulation noise, circuit noise, and interference. The digital filter shown in the encoder of Figure 1.2 serves to attenuate all of the out-of-band energy of this signal so that it may be resampled at the Nyquist rate without incurring significant noise penalty because of aliasing.

A fairly simple filter would suffice to remove the modulation noise alone because its spectrum rises slowly; for example, the noise increases 12 dB per octave for second-order $\Delta\Sigma$ modulation. However, abrupt low-pass filters are often needed to remove out-of-band components of the signal. And such filters are expensive to build at the elevated sampling rates of the modulator. In practice, it nearly always pays to perform the decimation in more than one stage [21]. This is illustrated in Figure 1.28, using the example of 4-kHz

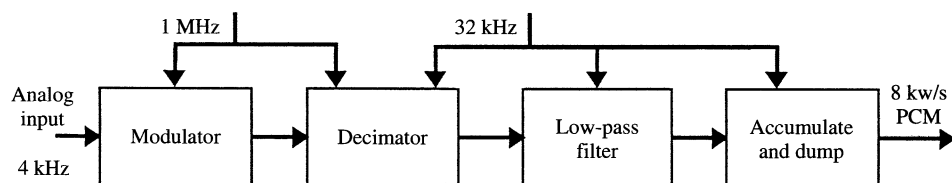


Figure 1.28 Decimating the output of the modulator in two stages, from 1 MHz to 32 kHz and then to 8 kHz.

telephone signals that have been modulated at 1 MHz. The first stage of decimation lowers the word rate from 1 MHz to 32kHz, an intermediate decimation frequency which is four times the Nyquist rate. The filter in this stage is designed primarily to remove modulation noise, because that noise dominate at high frequency. Out-of-band components of the signal dominate at lower frequency, and these are attenuated by the abrupt low-pass filter in the final stage of decimation. As the signal propagates through the filters and resampling stages, the word length increases from 1 to 16 bits in order to preserve the resolution as the word rate decreases. We will describe the design of these decimating circuits individually and explain why an intermediate frequency of four times the Nyquist rate was selected.

1.3.2 Design of the First-Stage Decimator

Figure 1.29 illustrates the action of the decimator. Frequencies below f_0 form the signal band, f_D is the intermediate decimation frequency, and f_s is the modulation frequency. The raised-cosine curve represents the spectral density of the quantization noise arising from second-order modulation. When this noise is sampled at f_D , its components in the vicinity of f_D and harmonics of f_D fold into the signal band. Consequently, it is sensible to place zeros of the decimation filter at these frequencies. There is no need for an abrupt cutoff at f_0 because noise in the range f_0 to $f_D - f_0$ folds on itself without entering the signal band. A small droop in the response over the signal band can easily be compensated in the filters of the next stage.

A convenient filter for this decimation has a frequency response based on sampled $\text{sinc}(\pi f/f_D)$ functions. An example is shown in Figure 1.29. The simplest of these decimators is the accumulate-and-dump circuit. If its input samples are x_i occurring at rate

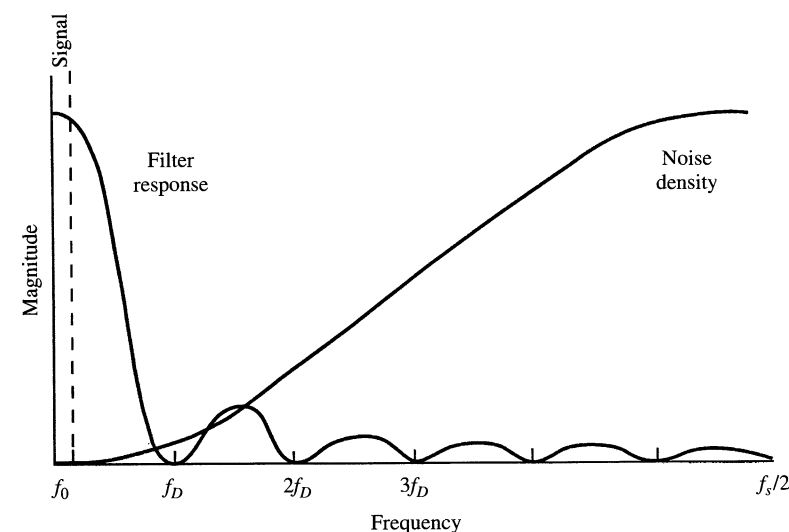


Figure 1.29 Decimating with a filter having sinc^2 frequency response; f_s is the modulation rate, f_D the intermediate decimation frequency, and $0 \leq f < f_0$ the signal band.

f_s and output samples are y_k occurring at f_D , then

$$y_k = \frac{1}{N} \sum_{i=N(k-1)}^{Nk-1} x_i \quad (1.27)$$

where the decimation ratio N is the integer ratio of the input frequency to the output frequency,

$$N = \frac{f_s}{f_D} \quad (1.28)$$

The transfer function of the filter is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{N} \sum_{i=0}^{N-1} z^{-i} = \frac{1}{N} \frac{1-z^{-N}}{1-z^{-1}} \quad (1.29)$$

and its frequency response is given for $z = e^{j\omega T}$ by

$$H(f) = \frac{\text{sinc}(\pi f N T)}{\text{sinc}(\pi f T)} \quad (1.30)$$

This has zeros at f_D and all of its harmonics in the range $f_0 \leq f < f_s$. This simple filter was used for decimation in oversampling A/D converters [2] at a time when it was important to have simple digital processing circuits. Much better performance can now be obtained by using a filter that is represented by a product of sinc functions [22, 23]. It has been shown

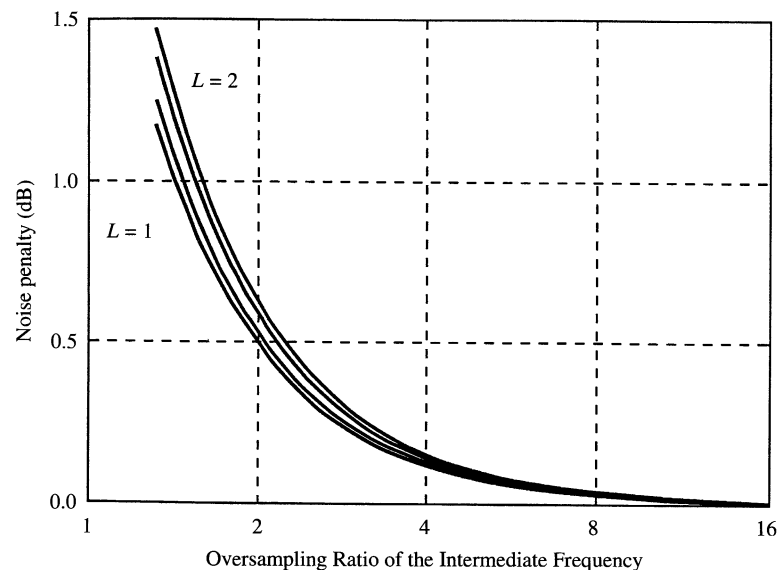


Figure 1.30 Increase in noise caused by decimation with sinc^{L+1} filters. $N = f_s/f_D$ is the decimation ratio. Results are plotted for first-order $\Delta\Sigma$ modulation $L = 1$ and second order $L = 2$.

that a filter function $[\text{sinc}(\pi f N T)/\text{sinc}(\pi f T)]^{(L+1)}$ is close to being optimum for decimating the signal from $\Delta\Sigma$ modulation of order L , which have noise spectral densities given by Eq. (1.12). The order of the filter should be one more than the order of the modulation. The penalty for using this class of decimation is typically less than a 0.5-dB increase in noise. Figure 1.30 plots the penalty against the oversampling ratio $(2f_0NT)^{-1}$ of the intermediate frequency signal.

When the intermediate frequency is four times the Nyquist rate, the penalty is about 0.14 dB, but it increases as the intermediate frequency is lowered. Another factor that influences the choice of intermediate frequency is the droop in the frequency response of the filter at the edge of the signal band f_0 . This is plotted in Figure 1.31 against the intermediate oversampling ratio for filters of various orders K and decimation ratios N . With third-order decimation and an intermediate oversampling ratio of 4, the droop is about 2.75 dB, but it increases rapidly if the intermediate oversampling ratio is lowered. It usually is inconvenient to compensate for more than 3 dB of droop.

Besides attenuating the modulation noise, the filter must also provide sufficient attenuation of the high-frequency components of the signal that alias into the signal when re-sampled at the intermediate frequency. We can see in Figure 1.29 that this attenuation is least at the frequency $f_D - f_0$. The attenuation at this frequency is plotted in Figure 1.32 for various conditions; it is about 50 dB for an intermediate oversampling ratio of 4, third-order decimation, and decimation ratios N greater than 15.

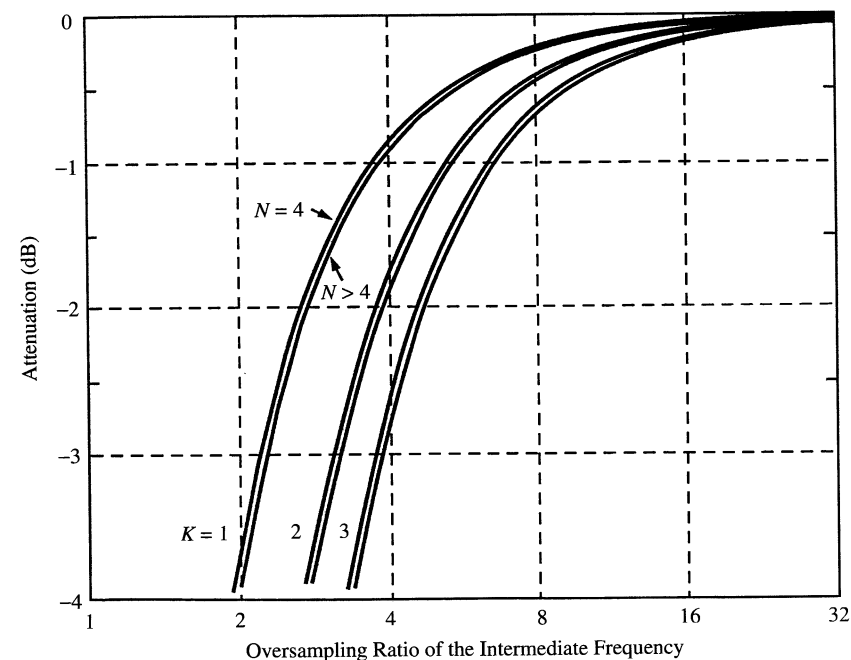


Figure 1.31 Attenuation of sinc^k decimation filters at the edge of the signal band, f_0 . This amount of droop in the signal needs to be compensated.

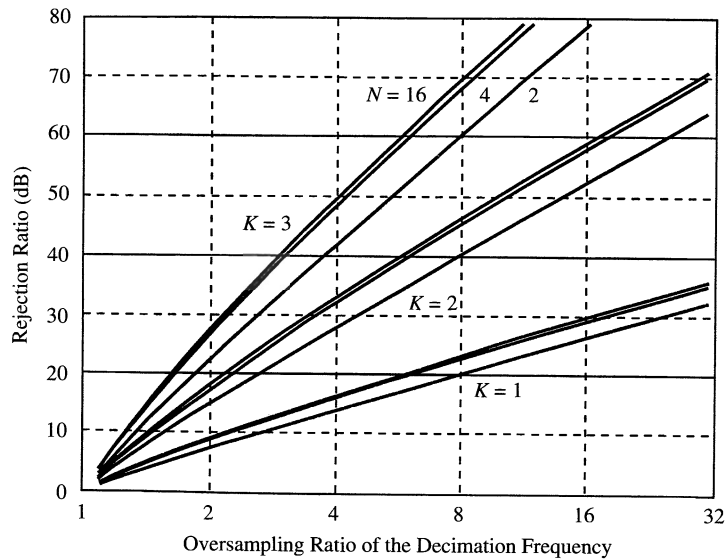


Figure 1.32 A graph of

$$\frac{\text{sinc}^k\{\pi(f_D - f_0)NT\}}{\text{sinc}^k\{\pi(f_D - f_0)T\}}$$

It is attenuation of out-of-band components of the signal at frequency $f_D - f_0$ for sinc^k decimation; N is the decimation ratio $N = f_s/f_D$. This attenuation should meet the antialiasing requirement of the application. In Section 1.4.2 we show how this graph can be used to measure the attenuation of an interpolator.

An intermediate oversampling ratio of about 4 and sinc^k decimation has favorable characteristic for use with $\Delta\Sigma$ modulation in many applications. Using ratios less than 4 results in rapidly deteriorating characteristics, higher ratios give less favorable design requirements for the low-pass filter in the next decimating stage. The results in Figure 1.30 do not apply to modulators that have sharply rising noise spectral densities, such as described in Section 1.2.3.6. One of the penalties of using these modulators is the fact that they need more complex decimation filters than do the ordinary $\Delta\Sigma$ modulators that have spectral densities [Eq. (1.21)] that rise more slowly with frequency. The next section shows that sinc^k decimators can have very simple implementations.

1.3.3 Implementing sinc Decimators

Good designs of decimators place the resampling within the filter so that the sample values that are not needed for the output are not calculated [21]. The input section of the filter processes the bits of short words in parallel at the high word rate. After resampling, the later stages of the filter can process the bits of the longer word serially at the lower word rate.

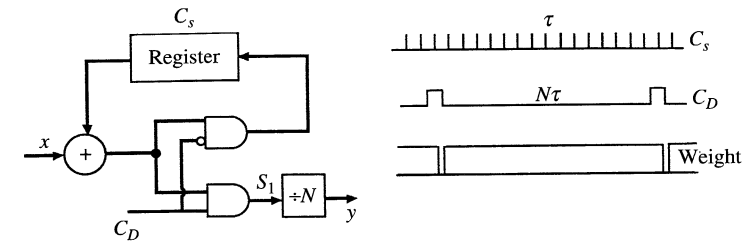


Figure 1.33 Accumulate-and-dump circuit. C_s is the input clock and C_D the output clock.

Figure 1.33 shows an implementation of the accumulate-and-dump function given by Eq. (1.29). Input words are added to the contents of the register. The sum is placed back in the register, except at the time of every N th input word, when the sum S_1 goes to the output, while the register is cleared.

Figure 1.34 shows a related implementation of the sinc^2 decimation [22]. It comprises a cascade of two accumulate-and-dump circuits. Their outputs S_1 and S_2 are separately converted from parallel to serial words and combined to give the next output. The output of the first accumulator, S_1 , is $NH(z)X(z)$, where $H(z)$ is given by Eq. (1.29), and the output of the second is given by

$$\frac{S_2}{X} = \sum_{i=0}^{N-1} \sum_{k=0}^i z^{-k} = \sum_{i=0}^{N-1} (i+1)z^{-i} = \frac{1-z^{-N}}{(1-z^{-1})^2} - \frac{Nz^{-N}}{1-z^{-1}} \quad (1.31)$$

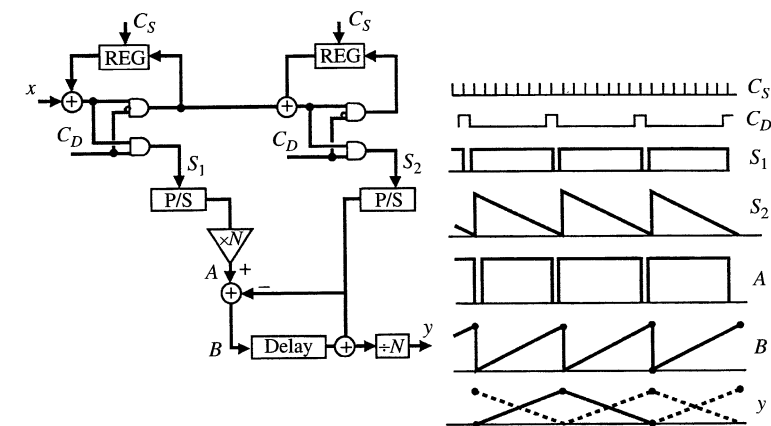


Figure 1.34 A sinc^2 decimating circuit, with a diagram of weighting factors in the summations of input samples at various points in the circuit; P/S denotes a change from parallel to serial format.

An expression for the signal filtered by the second-order sinc function can then be constructed as

$$\frac{1}{N^2} \left(\frac{1-z^{-N}}{1-z^{-1}} \right)^2 X = \frac{1}{N^2} \left[(NS_1 - S_2)z^{-N} + S_2 \right] \quad (1.32)$$

and this is implemented at the low word rate because the expression does not include the short delay z^{-1} , associated with the fast clock.

These decimating circuits are relatively simple to implement because the first accumulator needs to hold no more than $\log(N) + b$ bits and the second one $\log_2[N(N+1)] + b$ bits, b being the number of bits in the input word. When N is a power of 2, multiplication and division by N in Figures 1.33 and 1.34 are mere changes in the significance of the bits.

A third-order decimator can be designed by appending a third accumulator generating a sum S_3 . This is combined with the other accumulations according to the relationship

$$\left(\frac{1-z^{-N}}{1-z^{-1}} \right)^3 X = (1-z^{-N})^2 S_3 + Nz^{-N} (1-z^{-N}) \left(S_2 + \frac{S_1}{2} \right) + \frac{N^2 z^{-N}}{2} (1-z^{-N}) S_1 \quad (1.33)$$

An alternative method for designing decimators that gives simpler circuits for third- and higher-order filters [24] is illustrated by Figure 1.35. The input signal feeds to a cascade of K accumulators, which in normal operation are not reset. This provides the filter action $(1-z^{-1})^{-K}$. The signal is then resampled at rate f_D and feeds to a cascade of K differentiators to generate the decimating function $[(1-z^{-N})/(1-z^{-1})]^K$ of the input.

An apparent objection to this method is the fact that the accumulators need to be very large if sufficient space is provided to prevent their overflowing. The difficulty is avoided by employing modulo arithmetic [24]. Each accumulator and differentiator stage holds only sufficient bits to accommodate the length of the output word; no more than

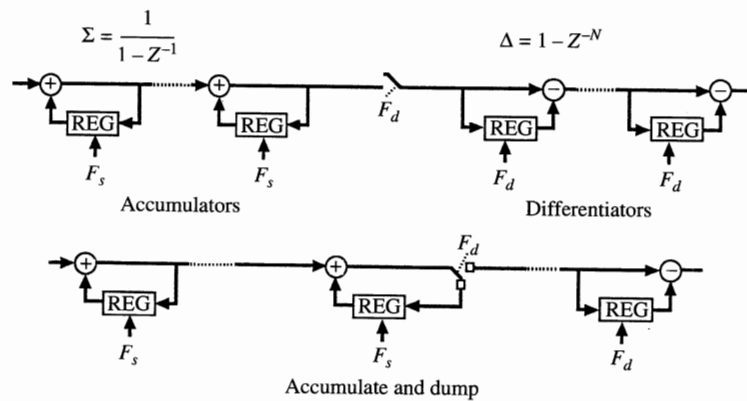


Figure 1.35 A sinc^K decimating circuit that comprises K accumulators, followed by resampling and K differentiators. All additions are performed modulo 2^b , b being the number of bits required in the output word. The second circuit resamples using an accumulate-and-dump circuit.

$K \log_2(N) + b$ bits. The circuits are allowed to overflow naturally. It can be shown that this overflow does not affect the net output of the decimator. The circuit can be simplified by replacing one accumulator stage and one differentiating stage, together with the resampling switch by an accumulate-and-dump stage, as shown in the second circuit in Figure 1.35.

1.3.4 The Low-Pass Filter

The low-pass filter in the final stage of the decimator in Figure 1.28 is designed to meet the antialiasing requirements of the input signal. Its circuit can usually be very simple [22] and because the word rate f_D is low, the circuit may process bits in serial rather than in parallel. Moreover, the word length of the coefficients can be so short that dedicated multiplier circuits can be used rather than shared ones. The accumulate-and-dump stage performs the final resampling to the Nyquist rate. Its frequency response [Eq. (1.30)] contributes useful zeros to the low-pass filter function. The positioning of these zeros depends on the intermediate oversampling ratio; there sometimes is advantage in using a nonbinary ratio such as 5 in order to place zeros off the axes.

Figure 1.36 shows the frequency response of a low-pass filter that is intended for decimating 4 kHz telephone signals from an intermediate sampling frequency of 40 kHz to the 8-kHz Nyquist rate. Zeros in the response near 4.5 and 6 kHz are included in two recursive sections; zeros at 10 and 20 kHz lie on the imaginary and negative axis of the z

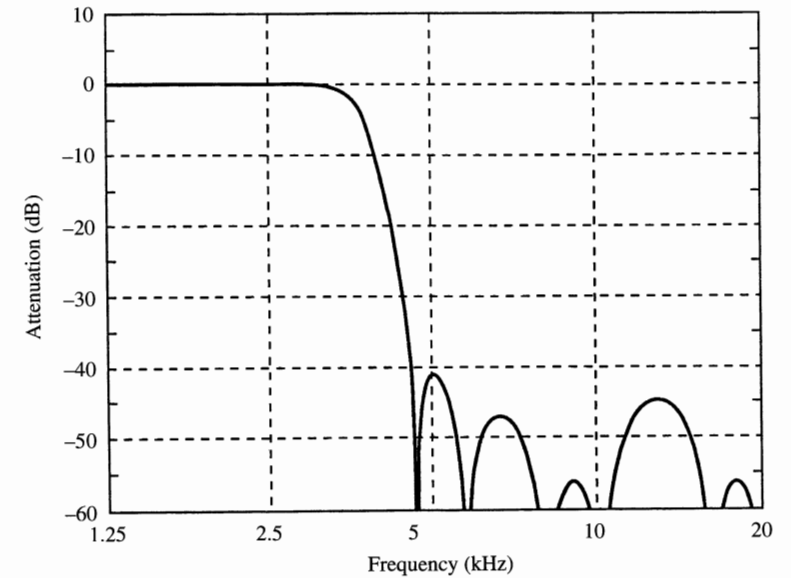


Figure 1.36 Frequency response of a low-pass filter used for decimating 4 kHz telephone signals from 40 to 8 kilowords per second (kw/s). Its z transform is given by

$$\left(\frac{1 - \frac{3}{2}z^{-1} + z^{-2}}{1 - \frac{11}{8}z^{-1} + \frac{5}{8}z^{-2}} \right) \left(\frac{1 - \frac{5}{4}z^{-1} + z^{-2}}{1 - \frac{101}{64}z^{-1} + \frac{7}{8}z^{-2}} \right) (1+z^{-1})(1+z^{-2}) \left(\frac{1-z^{-5}}{1-z^{-1}} \right)$$

plane, and zeros at 8 and 16 kHz are provided by the accumulate-and-dump stage. The poles of the recursive stages are chosen to flatten the in-band response. These circuits are easy to build because the word lengths of the coefficients are short [22].

1.4 OVERSAMPLING D/A CONVERTERS

1.4.1 Demodulating Signals at Elevated Word Rates

The lower part of Figure 1.2 shows an outline of an oversampling D/A converter. In this circuit a digital filter interpolates sample values of the input signal in order to raise the word rate well above the Nyquist rate [21]. A demodulator then truncates the words and converts them to analog form at the high sample rate. In most applications it is advantageous to raise the word rate of the signal in stages, in much the same way as it was decimated in stages at the encoder. We illustrate the details of the oversampling method for D/A conversion by an example that processes 4-kHz telephone signals encoded into 16-bit words at 8 kHz. Figure 1.37 shows an outline of this oversampling D/A converter [22]. The input words enter a register from which they feed into a low-pass filter at 32 kHz: Each word repeats four times. The output of the filter resembles a PCM encoding of the signal at 32 kHz. The next stage is a linear interpolation that inserts three new values between each adjacent pair of 32-kHz samples, raising the word rate to 128 kHz. The words enter a register from which they feed the demodulator at 1 MHz; each word repeats eight times. The demodulator rounds off the code to single-bit words, converts them to analog levels, and smooths these with an analog filter. The single-bit quantization occurs in a feedback circuit that shapes the spectrum of the quantization noise, moving most of the power far above the signal band. The 1 MHz demodulation rate is sufficiently high that a very simple analog filter will smooth the noise.

The filtering actions [21] that are inherent in the interpolation that raises the word rate from 8 kHz to 1 MHz smooth out sampling images of the signal, leaving only those adjacent to the new sampling rate, 1 MHz, and its harmonics. Figure 1.38 illustrates this action: (a) represents the spectral density of the baseband signal, (b) is spectral density when sampled at the Nyquist rate, (c) is the frequency response of the low-pass filter including the sinc response of the holding register, both (d) and (e) represent the output spectrum of the low-pass filter, (f) is the sinc^2 response of linear interpolation, (g) is the result of this interpolation, (h) is the frequency response of the final holding register, and (i) is the spectral density of its output.

The filter requirements for attenuating sampling images of the signal at the decoder are usually less stringent than are the requirements for preventing aliasing at the encoder.

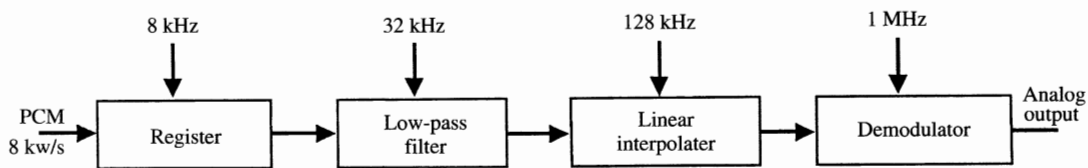


Figure 1.37 Oversampling decoder for 4 kHz signals.

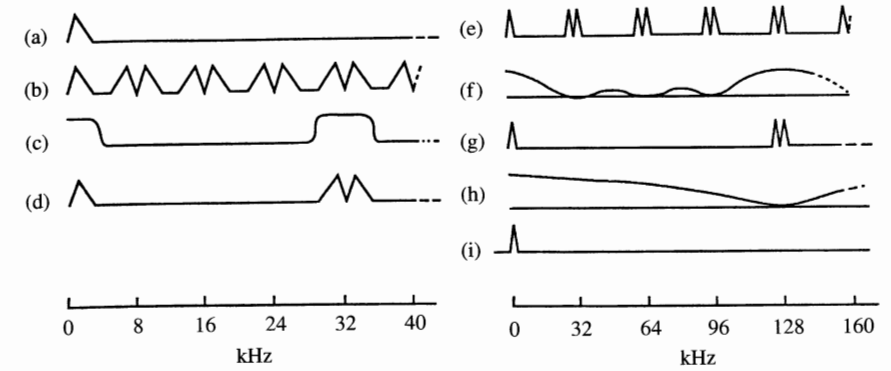


Figure 1.38 Spectral densities of signals, and the frequency response of filters used for interpolating sample values: (a) spectral density of the signal; (b) spectral density of the sampled signal; (c) low-pass filter characteristic; (d, e) spectral density of the filter output on different frequency scales; (f) frequency response of linear interpolation; (g) spectral density of the interpolated signal; (h) frequency response of the holding register; (i) spectral density of the held signal.

Consequently, a copy of the low-pass filter structure used in the encoder of Figure 1.28 is used as the low-pass filter in the decoder of Figure 1.27. The zeros in the filter response, which were provided by the accumulate-and-dump at the output of the A/D converter, are now provided by the holding register at the input of the D/A converter.

1.4.2 Interpolating with sinc^K -Shaped Filter Functions

The frequency responses associated with the linear interpolation circuit and the holding register are sampled sinc^K functions. The amount by which they attenuate sampling images of the signal needs to be calculated in order to determine good values for the intermediate sampling frequencies used between stages of the circuit. For this purpose, let f_I be the rate at which digital words are applied to an interpolating stage, and Nf_I the rate at which words emerge at its output. Here, N is defined as the interpolation ratio. The frequency response of this class of interpolation stage can be expressed as

$$I(f) = \frac{\text{sinc}^K(\pi f / f_I)}{\text{sinc}^K(\pi f / Nf_I)} \quad (1.34)$$

where K is the order of the interpolation. Images of the signal will be situated adjacent to f_I and all of its harmonics (i.e., in the frequency range $kf_I \pm f_0$), as illustrated in Figure 1.38. The attenuation of the unwanted images is least at the frequency $f_I - f_0$. The attenuation at this frequency may be read directly from the graphs in Figure 1.32, which was provided originally for evaluating decimators. The relevant value of the abscissa is now the oversampling ratio of f_I at the input to the interpolator (i.e., $f_I/2f_0$).

As an example, some telephone applications require that sampling images be attenuated by at least 28 dB. For interpolation ratios N greater than 3 this is achieved by using a holding register ($K = 1$) and an oversampling ratio of at least 16 at the input to the stage. Linear interpolation ($K = 2$) is satisfactory for oversampling ratios down to 4. These results are somewhat conservative because the digital low-pass filter attenuates the signal at the upper edge of its passband, and the analog filter at the output contributes to the smoothing of images.

1.4.3 Demodulator Stage

1.4.3.1 Quantizing the Digital Signal. The final stage of demodulation rounds off the long digital words to short ones, preferably all the way to single-bit words that will conveniently convert to analog form. Circuit structures of these quantizers resemble those of the modulators described in Section 1.2. The main difference is that the signals processed in the demodulator are digital instead of analog; hence, there is little trouble in achieving high precision. Properties of quantizing noise derived in Section 1.2 apply the noise of demodulation and may be used to determine the rates required to ensure adequately low demodulation noise. The need for this quantization distinguishes oversampling D/A converters from their conventional counterparts. Conventional converters contain no such intentional source of error, but they are much more sensitive to analog circuit imprecision.

Just as there are several forms of oversampling modulators, so also there are corresponding forms of demodulators. In general, there is no clear general advantage of one demodulator structure over others; the best choice depends on requirements of the application and on properties of the technology. Because much of the signal processing is digital in demodulators and analog in modulators, the trade-offs in their designs differ. We next discuss the design of several demodulator structures.

1.4.3.2 Quantization with Error Feedback. The error feedback circuit in Figure 1.23 is unsuitable for use as a modulator because of its sensitivity to inaccuracy in the analog subtractors, but this sensitivity is not a major concern in design of a demodulator. Figure 1.39 shows a digital implementation of an error feedback quantizer [25]. It uses digital codes that represent only positive values. The sum generated by the adder is quantized by using only its most significant bits as output. The remaining bits then represent the negative of the quantization error. These are delayed and added to the next input sample for error correction. In the extreme case of single-bit quantization, only the carry bit from the adder constitutes the output, and all the sum bits feed back.

The noise introduced into the signal by this quantization is the same form as first-order modulation noise [Eq. (1.8)]. For busy signals its spectral density is given by Eq. (1.9), and the noise power in the signal band by Eq. (1.11). For slowly changing inputs, the noise has a pattern structure as illustrated by Figure 1.8. For this and other reasons, first-order demodulation does not find wide application today; high-order demodulators are usually preferred. The circuit was useful at times when it was important to have very simple circuits.

Higher order noise shaping is achieved by replacing the delay in the feedback path by a prediction filter [12]. Figure 1.40 shows a noise-shaping demodulator of this kind. The

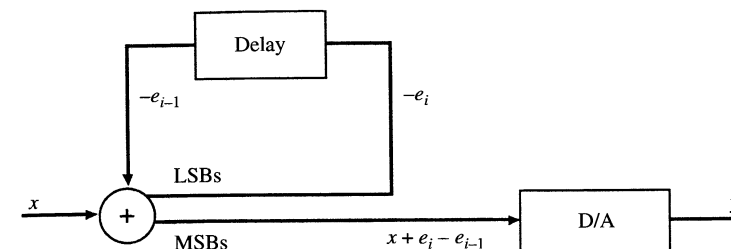


Figure 1.39 Digital quantization with error feedback (LSB, least significant bit; MSB, most significant bit).

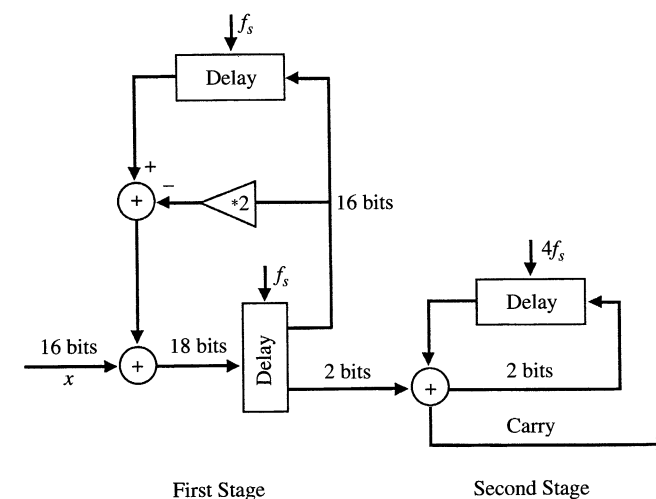


Figure 1.40 Second-order error-feedback quantizer with an auxiliary first-order quantization to 1-bit code.

first stage of the circuit is a second-order quantizer. This circuit requires that its output words comprise of at least two bits if clipping of signal amplitudes in the feedback loop is to be avoided. Including a clipping circuit in order to get single-bit outputs causes significant increase in noise, no matter where the clipping occurs. To avoid this penalty, the second stage of the circuit in Figure 1.40 is included to perform a first-order quantization of the two-bit words to single-bit ones [26]. This second stage is clocked at least four times faster than the first stage. The technique is possible when this higher frequency is achievable, but then, one also has the option of avoiding the need for the second stage altogether, by clocking the first stage fast enough to allow for the increased noise caused by clipping the feedback signal.

1.4.3.3 Cascaded Demodulators. Demodulator circuits can be cascaded [27] in the same way that modulators are cascaded in Figure 1.24. Figure 1.41 shows a cascade of two first-order quantizers in which the derivative of the second-stage output is added to the output of the first stage. This results in a cancellation of the first-stage noise from the net output. In contrast to cascaded analog quantizers, these digital quantizers can provide perfect cancellation because there is no inherent reason for error in the digital circuits besides the two quantization errors.

Even when the outputs of the individual stages are single-bit words, the net output contains two-bit words. This is an advantage of cascaded modulators because it allows the output to oscillate between four levels and avoids introducing excess noise into large signals. However, for demodulation this advantage is outweighed by the need for high precision in the four-level D/A at the output. This difficulty has been overcome to some extent by providing a separate D/A converter at the output of each stage. Then the signal passes only through the first-stage D/A, which should be two level to avoid distorting the signal. Only noise passes through the second D/A, and its imperfections result in imperfect noise cancellation, not signal distortion. Reference [27] describes a demodulator where the differentiation of the second-stage output is performed in the analog circuit; then both D/A converters can be single-bit ones. The precision required in matching the analog circuits is similar to the accuracy requirement of parameter g in Eq. (1.24).

1.4.3.4 Circuit Design for $\Delta\Sigma$ Demodulation. A digital implementation of an ordinary $\Delta\Sigma$ demodulator can serve to quantize the signal in a demodulator. There are many ways of designing these circuits, and Figure 1.42 shows one example of a second-order quantizer. This circuit introduces noise that is given by Eqs. (1.19) and (1.20). Although it is not troubled by leakage in the accumulators, tones may be present in the

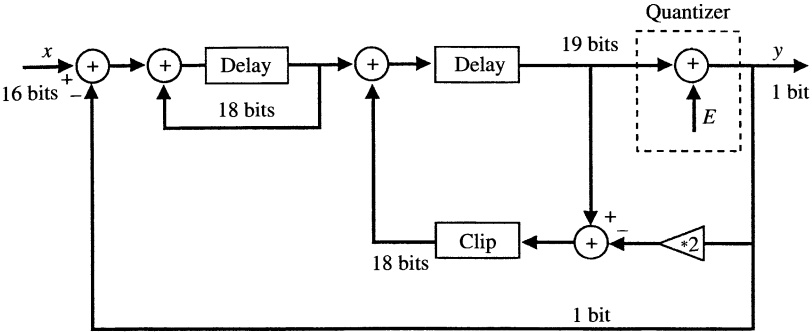


Figure 1.42 Equivalent circuit of a digital second-order $\Delta\Sigma$ quantizer.

noise. The randomness in the oscillation pattern of second-order modulators depends on avoiding signals that are rational multiples of the level spacing in the accumulators. This is not possible in digital implementations, where the signal values are always rational. One method of ensuring randomness is to inject a relatively large dither signal. Another is to add a random bit pattern to the least significant bit of the input, to mimic the effect of having irrational values stored in the first accumulator.

Structures used for higher order feedback quantizers of the type described in Section 1.2.3.6 may also be used in demodulators. It may also be possible to use higher order digital $\Sigma\Delta$ quantizers with digital control circuits that prevent them from going into saturating limit cycles.

1.5 CONCLUSION

Oversampling methods can provide very high resolutions even when relatively inaccurate analog components are used. For example, 20-bit resolution has been reported for 20-kHz audio applications. The ever-increasing speed capabilities of new VLSI technology will allow larger oversampling ratios and possibly higher resolutions, but this will soon be limited by circuit noise.

Designers of oversampling converters can select from a wide variety of architectures for modulators and demodulators each with its own advantages and disadvantages. They can make trade-offs between oversampling ratios, resolution, circuit complexity, and circuit tolerances and choose from numerous designs of digital decimation and interpolation filters. The later chapters of this book will describe many of these options in more detail than has been attempted here. The second chapter will lay a rigorous foundation for the design equations that have been introduced here.

REFERENCES

[1] J. C. Candy and G. C. Temes, "Oversampling methods for A/D and D/A conversion," *Oversampling Delta-Sigma Data Converters*, IEEE Press, New York, 1992, pp. 1–275.
[2] J. C. Candy, "A use of limit cycle oscillations to obtain robust analog-to-digital converters," *IEEE Trans. Commun.*, vol. COM-22, pp. 298–305, March 1974.

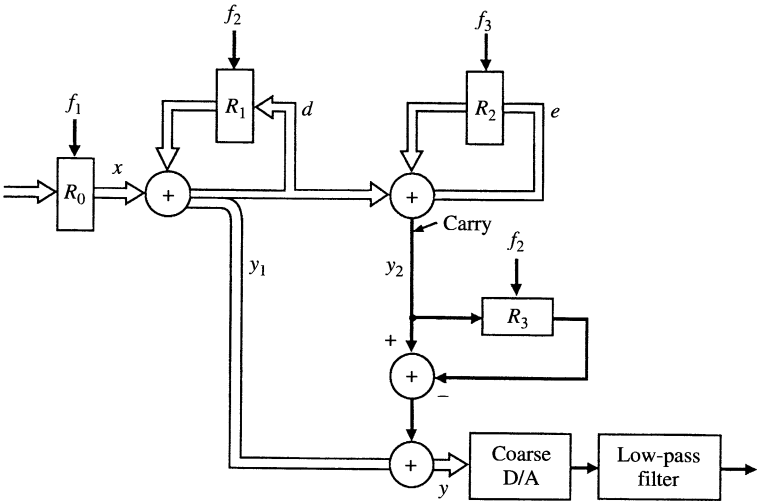


Figure 1.41 Cascade of two first-order digital quantizers with digital combining of their outputs. Single-bit words travel on the bold single-line paths. Multibit words travel on the wide paths.

- [3] J. C. Candy, Y. C. Ching, and D. S. Alexander, "Using triangularly weighted interpolation to get 13-bit PCM from a sigma-delta modulator," *IEEE Trans. Commun.*, vol. COM-29, pp. 815–830, June 1981.
- [4] A. N. Netravali, "Optimum filters for interpolative A/D converters," *Bell Sys. Tech. J.*, vol. 56, pp. 1629–1641, Nov. 1977.
- [5] J. C. Candy and O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-29, pp. 1316–1323, Sept. 1981.
- [6] R. M. Gray, "Quantization noise spectra," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 1220–1244, Nov. 1990.
- [7] B. H. Leung, R. Neff, P. R. Gray, and R. W. Brodersen, "Area-efficient multichannel oversampled PCM voice-band coder," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1351–1357, Dec. 1988.
- [8] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, March 1985.
- [9] M. W. Hauser and R. W. Brodersen, "Circuit and technology considerations for MOS delta-sigma A/D converters," *IEEE Proc. ISCAS '86*, pp. 1310–1315, May 1986.
- [10] T. Cataltepe, G. C. Temes, and L. E. Larson, "Digitally corrected multi-bit Σ - Δ data converters," *IEEE Proc. ISCAS '89*, pp. 647–650, May 1989.
- [11] R. W. Adams, "Companded predictive delta modulation; a low cost conversion technique for digital recording," *J. Audio Eng. Soc.*, vol. 32, pp. 659–672, Sept. 1984.
- [12] H. A. Spang III and P. M. Schultheiss, "Reduction of quantizing noise by use of feedback," *IRE Trans. Commun. Sys.*, pp. 373–380, Dec. 1962.
- [13] B. E. Brandt, D. E. Wingard, and B. A. Wooley, "Second-order sigma-delta signal acquisition," *IEEE J. Solid-State Circuits*, vol. SC-26, pp. 618–627, April 1991.
- [14] S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Trans. Circuits Sys.*, vol. CAS-34, pp. 593–603, June 1987.
- [15] B. E. Boser and B. A. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1298–1308, Dec. 1988.
- [16] K. C. H. Chao, S. Nedeem, W. L. Lee, and C. G. Sodini, "A higher order topology for interpolative modulators for oversampling A/D conversion," *IEEE Trans. Circuits Sys.*, vol. CAS-37, pp. 309–318, March 1990.
- [17] Y. Matsuya, K. Uchimura, A. Iwata, et al., "A 16-bit oversampling A-to-D conversion technology using triple-integration noise shaping," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 921–929, Dec. 1987.
- [18] L. Logo and M. Copeland, "A 13 bit ISDN-band oversampled ADC using two-stage third order noise shaping," *IEEE Proc. Custom IC Conf.*, pp. 21.2.1–21.2.4, Jan. 1988.
- [19] L. A. Williams III and B. A. Wooley, "Third-order cascade sigma-delta modulators," *IEEE Trans. Circuits Sys.*, vol. CAS-38, pp. 489–498, May 1991.
- [20] T. C. Leslie and B. Singh, "An improved sigma-delta modulator architecture," *IEEE Proc. ISCAS '90*, pp. 372–375, May 1990.
- [21] R. E. Crochiere and L. R. Rabiner, "Interpolation and decimation of digital signals—a tutorial review," *Proc. IEEE*, vol. 69, pp. 300–331, March 1981.
- [22] J. C. Candy, B. A. Wooley, and O. J. Benjamin, "A voiceband codec with digital filtering," *IEEE Trans. Commun.*, vol. COM-29, pp. 815–830, June 1981.

- [23] J. C. Candy, "Decimation for sigma delta modulation," *IEEE Trans. Commun.*, vol. CAS-31, pp. 913–924, Nov. 1984.
- [24] E. B. Hogenaur, "An economical class of digital filters for decimation and interpolation," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. ASSP-29, April 1981.
- [25] G. R. Ritchie, J. C. Candy, and W. H. Ninke, "Interpolative digital to analog converters," *IEEE Trans. Commun.*, vol. COM-22, pp. 1797–1806, Nov. 1974.
- [26] H. G. Musmann and W. Korte, "Generalized interpolative method for digital/analog conversion of PCM signals," U.S. Patent, No. 4,467,316, 1984 (filed 1981).
- [27] J. C. Candy and An-Ni Huynh, "Double interpolation for digital-to-analog conversions," *IEEE Trans. Commun.*, vol. COM-34, pp. 77–81, Jan. 1986.

Robert M. Gray

Chapter 2

Quantization Noise in $\Delta\Sigma$ A/D Converters

2.1 INTRODUCTION

The heart of a $\Delta\Sigma$ modulator and any other analog-to-digital converter (ADC) is a quantizer, a device that maps real numbers into a finite set of possible representative values, often as few as two. Any analysis of the behavior of a $\Delta\Sigma$ modulator must include consideration of the behavior of the quantizer. The quantization operation is inherently nonlinear and hence rigorous analysis is complicated even in the simplest of systems. When quantizers are incorporated into linear systems with feedback such as $\Delta\Sigma$ modulators and bang-bang control systems, the analysis becomes even more difficult. Simulations cannot capture all aspects of possible system behavior and are not always reproducible as different random number generators are used and care is not always taken to ensure that sample functions are long enough for sample averages to be close to expectations with high probability. As a result, various methods based on approximations have been widely used, even in some applications where they were known to give misleading or outright incorrect results. Often, however, approximate methods have quite successfully predicted some aspects of system behavior, as many of the other chapters of this book will attest. These approximations are usually implicitly or explicitly based on either the asymptotic results of Bennett [1] or on the exact results of Widrow [2] as extended by Sripad and Snyder [3]. We shall see, however, that the underlying conditions assumed by these results can be and usually are invalid in typical $\Delta\Sigma$ modulators, and there are not any good guidelines for determining when the approximations might nonetheless yield good results. A common reason for using the approximation methods in spite of their shortcomings is that simulations are inadequate and that exact analysis is thought to be either impossible or prohibitively difficult. Successful applications of the approximations do not lessen the puzzlement of engineers who use such

methods to quantify the behavior of a system and then find that the system exhibits bizarre artifacts not suggested by the theory. They often suspect the system rather than the theory of being flawed. A classical example is the ability of approximate methods to correctly predict the signal-to-noise ratio of a single-loop $\Delta\Sigma$ modulator while producing a completely incorrect prediction of the quantization error spectrum that fails to include audible objectionable tones in the final output.

The goal of this chapter is to discuss in some detail the most common approximations, their underlying justification in quantization systems, and the common errors made in their application. The hope is to give practicing engineers a degree of skepticism in their use of these methods and to prepare them for unpleasant surprises. We also consider a variety of examples where the approximations are not needed and exact descriptions of system behavior can be found by combining linear systems methods with a few nonlinear system techniques. The mathematical methods are not particularly deep; most come from Fourier analysis and probability theory, but much of the algebra and calculus is not particularly pretty and is left to the references. By “exact” it is meant that exact solutions are found to the nonlinear difference equations modeling $\Delta\Sigma$ modulators. Certainly real systems will have many variations that are not yet included in the basic equations, but the given nonlinear equations will be solved without recourse to linearizing approximations.

Unfortunately, the exact solutions do not extend to all $\Delta\Sigma$ architectures of practical interest, but they do provide a rich collection of types of behavior and of important attributes of systems that determine that behavior. These examples can provide useful insight to more complex systems. The theory also provides some surprising results, including the fact that some of the common approximations hold exactly in some systems even though the underlying conditions usually assumed for those approximations are violated.

Because the quantizer plays the key role in a $\Delta\Sigma$ modulation, we begin by looking at quantization error in a simple quantizer that is used without feedback or linear filtering. This leads in several steps to the more complicated example of $\Delta\Sigma$ modulation.

2.2 UNIFORM QUANTIZATION

The basic common component to most analog-to-digital converters is a uniform quantizer. It is assumed that the quantizer has an even number, say M , of levels and that the distance between the output levels (the *bin width*) is Δ . The special case of $M = 2$ is common in $\Delta\Sigma$ modulators, but the theory here and later holds for any even M .

The M quantization levels are equally interspersed in the interval $B = [a, a + M\Delta]$, where often the region is chosen to be symmetric around the origin, that is, $a = -M\Delta/2$, and each level y_k is the center of its quantization cell $R_k = [a + k\Delta, a + (k + 1)\Delta]$; $k = 0, 1, \dots, M - 1$. Any input u in this range will map into a quantized value $q(u) = y_k$ if $u \in R_k$; that is, the quantizer mapping is a minimum distance (nearest neighbor) mapping. For a given number of levels M , there are only two free parameters in a uniform quantizer: the offset a and the bin width Δ .

The quantizer error is defined as $\varepsilon = q(u) - u$. If u is in the region B , then the maximum error resulting is $\Delta/2$. Outside this region, the input is mapped into the nearest quantization level, but the error is greater than $\Delta/2$ and the quantizer is said to *overload* or

saturate. We will refer to the interval B as the *no-overload region* of the quantizer. Some of the methods described hold only when u is in the no-overload region with probability 1.

If the input is described by a probability distribution, then the performance of a quantizer is often measured by the mean-square error, the expected error energy $E(\epsilon^2)$. A uniform quantizer is said to be *optimal* for an input probability distribution if, for the given M , the parameters a and Δ are chosen to minimize the mean-square error.

If the input is a sequence of samples u_n , then we will be interested in the error sequence $\epsilon_n = q(u_n) - u_n$, and we will wish to see if it resembles random noise in some way. A trivial rewriting of this formula yields the so-called additive noise model of quantization

$$q_n = q(u_n) = u_n + \epsilon_n$$

expressing the output of the quantizer as its input plus a “noise” term. There is no genuine modeling here; this is simply a convenient definition of the quantization error such that the output can be written as the sum of the input and a noise term. The modeling enters when assumptions are made about the statistical behavior of ϵ and its dependence on the input signal, as will be seen.

2.3 ADDITIVE WHITE-NOISE APPROXIMATION

Many of the original results and insights into the behavior of quantization error are due to Bennett [1], and much of the work since then has its origins in that classic study. Bennett first developed conditions under which quantization noise could be reasonably modeled as additive white noise. Unfortunately, much of the literature assumes more than was proved by Bennett and often uses Bennett’s approximations when his results do not apply. Subsequently Sripad and Snyder [3] extended Widrow’s approach [2] and found necessary and sufficient conditions for certain aspects of the approximation to hold exactly. We explore here the approximations and these underlying conditions in order to consider their suitability for use in analyzing $\Delta\Sigma$ modulators.

A common statement of the approximation is that the quantization error ϵ_n has the following properties, which we refer to collectively as the “input-independent additive white-noise approximation”:

Property 1. ϵ_n is statistically independent of the input signal u_k for all n, k (strong version) or ϵ_n is uncorrelated with the input signal u_n (weak version).

Property 2. ϵ_n is uniformly distributed in $[-\Delta/2, \Delta/2]$.

Property 3. ϵ_n is an independent identically distributed (i.i.d.) sequence (strong version) or ϵ_n has a flat power spectral density (it is “white”) (weak version).

These approximations enormously simplify system analysis because they replace a deterministic nonlinearity by a stochastic linear system, thereby permitting the use of linear systems methods to analyze a nonlinear system containing a quantizer. These properties have been used in the wide majority of published analyses of systems containing quantizers in the communications, control, and signal processing literature. Most often the

approximation is made without reference and with only small (if any) mention of its possible limitations. The natural questions that arise are

Question 1. Are the approximations good under ordinary conditions; that is, do they accurately model the true behavior of quantization error?

Question 2. If the approximations are not good, is it still possible for them to yield good predictions of actual system behavior?

It is easy to demonstrate a negative answer to question 1 if the strong form of the approximation is considered: The quantizer error is a deterministic function of the input and hence cannot be statistically independent of the input. There is hope, however, that a weaker form of independence of uncorrelation (or linear independence) holds in the sense that $E[u_n \epsilon_k] = E[u_n]E[\epsilon_k]$ for all n, k , where E denotes expectation or probabilistic average. This property is sufficient to ensure that second-order analysis involving output correlations and spectra can be carried out without the complexity of cross terms. This allows the common “noise-shaping” interpretation of $\Delta\Sigma$ modulators because it implies that the quantization noise can be filtered without thereby also changing the input signal. This approximation is almost universally made in the analysis of oversampled ADCs, yet, as we shall see, it can be incorrect.

Question 2 is not so easily answered. Engineering mathematics often uses ideas such as impulses and flicker ($1/f$) noise that are physically impossible, yet yield perfectly good predictions of real system behavior when carefully used and suitably interpreted. The answer to this question will vary depending on the system, and a goal of this chapter is to provide a feel for examples where answers based on the white-noise approximation can be trusted and where they cannot be.

If we back off on the complete input-independent additive noise model by eliminating the first property, Bennett’s theory provides a motivation for approximating quantization noise by properties 2 and 3, which will be seen to hold under specific conditions (first proved by Bennett). When these properties hold approximately, we shall refer to the approximation as the “additive white noise approximation,” dropping the “input-independent” modifier. We shall also discuss the nonasymptotic results of Sripad and Snyder, which provide conditions under which several of the common approximations hold in an exact sense. Unfortunately, we shall see that the conditions of both Bennett and of Sripad and Snyder do not hold in typical $\Delta\Sigma$ modulators, and hence the additive white-noise approximation is not justified mathematically. Perhaps surprisingly, we shall later find that in spite of this fact, the additive white noise approximation in fact holds exactly for ideal multistage and higher order $\Delta\Sigma$ architectures provided the quantizers do not overload.

Property 2 might be true if the quantizer cannot overload, but it is clearly false if overload occurs with nonzero probability. Property 3 is also plausible, but must be demonstrated for a particular system. A simple but important example where both properties 2 and 3 hold exactly is in the case where the input signal is itself i.i.d. and is uniformly distributed over the no-overload range. In this example it is easy to see that the quantization error is uniformly distributed over $(-\Delta/2, \Delta/2)$ and has zero mean and variance $\Delta^2/12$, the ubiquitous result for the mean-square error of a uniform quantizer with bin width Δ . It should be pointed out that even in the case of a uniformly distributed input signal, property

1 is not true even in the weak sense; that is, the input signal and quantization error are not uncorrelated. It is straightforward to show that $E(u_n \varepsilon_k) = -\sigma_{\varepsilon_n}^2 \delta_{n-k}$, where δ_l is the Kronecker delta function, 1 if $l = 0$ and 0 otherwise. In words, the correlation between input and error at a common sample time is as large as the error variance, it is not 0! In contrast, if the input probability density is a triangle on the no-overload range (the convolution of two uniform densities covering half the no-overload range), then again the error is uniformly distributed and white, but now $E(u_n \varepsilon_k) = 0$ for all n and k ; that is, the input and the quantizer are indeed uncorrelated. The point is that the weak version of property 1 might or might not hold, depending on the input density and the particular quantizer.

Bennett argued more generally that properties 2 and 3 are approximately true (along with some other properties) if certain underlying conditions hold. The uniform white quantization noise assumption subsequently gained a wide popularity, largely due to the work of Widrow [2] who provided a sufficient condition for the quantizer error to be uniformly distributed. For completeness we quote the basic results of Bennett and sketch their proof.

Bennett's Theorem

Suppose that the following conditions (Bennett's conditions) hold:

1. The input is in the no-overload region.
2. The term M is asymptotically large.
3. The term Δ is asymptotically small.
4. The joint probability density function (pdf) of the input signal at different sample times is smooth.

Then the error sequence has the following properties:

1. The sequence $\{\varepsilon_n\}$ is approximately uniformly distributed; that is, it has marginal pdf

$$f_{\varepsilon}(\alpha) \approx \begin{cases} \frac{1}{\Delta} & \text{if } \alpha \in \left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right) \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

This in turn implies that $E[\varepsilon_n] \approx 0$,

$$\sigma_{\varepsilon_n}^2 \approx \frac{\Delta^2}{12} \quad (2.2)$$

and

$$E[q_n] \approx E[u_n] \quad (2.3)$$

that is, the expectation of the quantized output is approximately the same as that of the input. In statistical terms, the quantized value is an unbiased estimator of the input.

2. The sequence $\{\varepsilon_n\}$ is approximately an i.i.d. random process.

Sketch of Proof of Bennett's Theorem. First consider marginal distribution of the error. Define as usual the cumulative distribution function (cdf) $F_{\varepsilon_n}(\alpha) = \Pr(\varepsilon_n \leq \alpha)$; $\alpha \in (-\Delta/2, \Delta/2)$ and the pdf $f_{\varepsilon_n}(\alpha) = dF_{\varepsilon_n}(\alpha)/d\alpha$. Referring to the definitions of q and ε we can write

$$F_{\varepsilon_n}(\alpha) = \sum_{k=0}^{M-1} \Pr(\varepsilon_n \leq \alpha \text{ and } u_n \in R_k)$$

Since the pdf is assumed to be smooth, the mean value theorem of calculus implies that

$$\Pr(\varepsilon_n \leq \alpha \text{ and } u_n \in R_k) = \int_{-(M/2+k)\Delta}^{-(M/2+k)\Delta + \alpha} f_{u_n}(\beta) d\beta \approx f_{u_n}(y_k) \alpha$$

Using the Riemann sum approximation to an integral yields

$$F_{\varepsilon_n}(\alpha) \approx \frac{\alpha}{\Delta} \sum_{k=0}^{M-1} f_{u_n}(y_k) \Delta \approx \frac{\alpha}{\Delta} \int_{-M/2}^{M/2} f_{u_n}(u) du \approx \frac{\alpha}{\Delta}$$

and hence,

$$f_{\varepsilon_n}(\alpha) \approx \frac{1}{\Delta} \text{ for } \alpha \in \left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$$

To prove that the error sequence is approximately memoryless, a similar idea is applied to vectors of error samples:

$$\Pr(\varepsilon_l \leq \alpha_l; l = n, \dots, n+k-1) = \sum_{i_1, \dots, i_k} \Pr(\varepsilon_l \leq \alpha_l \text{ and } u_l \in R_{i_l}; l = n, \dots, n+k-1)$$

For $\alpha_l \in (-\Delta/2, \Delta/2)$, $l = 1, \dots, k$

$\Pr(\varepsilon_l \leq \alpha_l \text{ and } u_l \in R_{i_l}; l = n, \dots, n+k-1)$

$$\begin{aligned} &= \int_{-(M/2+k)\Delta}^{-(M/2+k)\Delta + \alpha_1} \dots \int_{-(M/2+k)\Delta}^{-(M/2+k)\Delta + \alpha_k} f_{u_n, \dots, u_{n+k-1}}(\beta_1, \dots, \beta_k) d\beta_1 \dots d\beta_k \\ &\approx f_{u_n, \dots, u_{n+k-1}}(y_{i_1}, \dots, y_{i_k}) \alpha_1 \dots \alpha_k \end{aligned}$$

where

$$f_{\varepsilon_n, \dots, \varepsilon_{n+k-1}}(\alpha_1, \dots, \alpha_k) \approx \frac{1}{\Delta^k}$$

This immediately implies that

$$R_{\varepsilon}(n, k) = E[\varepsilon_n \varepsilon_k] \approx \sigma_{\varepsilon}^2 \delta_{n-k}$$

Bennett did not explicitly treat the issue of the correlation of quantizer error and input, but his basic method of calculus approximations can be applied to the task. The analysis is somewhat more delicate, however, since higher order terms can make a difference as pointed out by Kollár [4]. It is easy to argue that $R_{q,\varepsilon}(n, k) \approx 0$ when $n \neq k$, so we focus on the case of $n = k$. Following [4], we can approximate the input density in the quantization bin R_l for output y_l by a Taylor series expansion as $f_u(y_l + \delta) \approx f_u(y_l) + f'_u(y_l)\delta$, where the higher order terms can be shown to be negligible in comparison. This leads to the approximation that

$$\begin{aligned} R_{q,\varepsilon}(n, k) &= E[q(U_n)\varepsilon_k] = \sum_{l=0}^{M-1} y_l E[\varepsilon_k | q(u_n) = y_k] \Pr[q(u_n) = y_k] \\ &= \sum_{l=0}^{M-1} y_l \int_{-\Delta/2}^{\Delta/2} (-\delta) [f_u(y_l) + f'_u(y_l)\delta] d\delta \\ &= - \sum_{l=0}^{M-1} \frac{\Delta^2}{12} y_l f'_u(y_l) \\ &\approx \frac{\Delta^2}{12} \int_a^{a+M\Delta} y f'_u(y) dy \end{aligned} \quad (2.4)$$

Integrating by parts then yields

$$R_{q,\varepsilon}(n, n) \approx \frac{\Delta^2}{12} [1 - (a + M\Delta)f_u(a + M\Delta) - af_u(a)] \quad (2.5)$$

The behavior thus depends on the behavior of the input density near the borders of the no-overload range $[a, a + M\Delta]$. If the density is zero at the edge of the no-overload range, then $R_{q,\varepsilon}(n, n) \approx \Delta^2/12 = R_\varepsilon(n, n)$, which implies in turn that

$$\begin{aligned} R_{u,\varepsilon}(n, k) &= E[u_n \varepsilon_k] = E[(q(u_n) - \varepsilon_n) \varepsilon_k] \\ &= E[q(U_n) \varepsilon_k] - R_\varepsilon(n, k) = 0 \end{aligned} \quad (2.6)$$

that is, $R_{u,\varepsilon}(n, k) \approx 0$, and the input and quantization error are uncorrelated. If, however, the density is not 0 at the borders of the no-overload zone (as is the case with a uniform density on the no-overload region), then the signal and quantizer error are not uncorrelated. Bucklew and Gallagher [5] have shown that if the uniform quantizer is optimal—that is, if a and Δ are chosen to minimize mean-square quantization error—then one will have exactly $R_{q,\varepsilon}(n, k) = 0$ and hence $R_{u,\varepsilon}(n, k) = -\sigma_\varepsilon^2 \delta_{n-k}$, and the input and quantizer error have correlation equal to minus the quantizer error energy. Optimal choice of Δ will involve a shrinking of the bin width and a resulting overload of the quantizer, but the Bennett approximations still hold [5]. We note that this same result holds if the Lloyd-Max optimal nonuniform quantizer is used (see, e.g., [6–7] and pp. 180–181 of [8]).

An alternative to the asymptotic (large number of quantizer levels) analysis of Bennett is the exact approach of Sripad and Snyder [3], which is a variation of the characteristic function method that will be used here. Sripad and Snyder demonstrated necessary and sufficient conditions for the various properties to hold. The conditions are stated in terms of the characteristic function

$$\Phi_u(v) = E[e^{jv u_n}] \quad (2.7)$$

of the input random variable and the joint characteristic function $\Phi_{u_n, u_k}(v, \mu) = E[e^{j(v u_n + \mu u_k)}]$. The input process $\{u_n\}$ is assumed to be stationary so that the characteristic function does not depend on n .

- A necessary and sufficient condition for the quantizer error to be uniformly distributed on $[-\Delta/2, \Delta/2]$ is that

$$\Phi_u\left(\frac{2\pi k}{\Delta}\right) = 0 \quad k = \pm 1, \pm 2, \dots \quad (2.8)$$

- A necessary and sufficient condition for ε_n and ε_k to be independent and uniformly distributed on $[-\Delta/2, \Delta/2]$ is that $\Phi_{u_n, u_k}(2\pi l/\Delta, 2\pi m/\Delta) = 0$ for all l, m for which $(l, m) \neq (0, 0)$.
- A sufficient condition for ε_n and u_n to be uncorrelated is that

$$\Phi_u\left(\frac{2\pi k}{\Delta}\right) = \dot{\Phi}_u\left(\frac{2\pi k}{\Delta}\right) \quad k = \pm 1, \pm 2, \dots \quad (2.9)$$

where $\dot{\Phi}$ is the derivative of Φ_u with respect to its argument.

The condition that Φ_u have zero value for all integral multiples of $2\pi/\Delta$ is satisfied by a random variable u having a uniform density over $[-\Delta/2, \Delta/2]$ as well as for any density formed by adding an independent random variable x to u to form $x + u$, since the resulting product of characteristic functions will inherit the zeros of Φ_u . It is also satisfied for a uniform density on $[-A, A]$ if $2A/\Delta$ is an integer M . The sufficient condition for uncorrelated signal and quantization error is satisfied, for example, by a triangular density on $[-A, A]$ if $2A/\Delta$ is an integer M . It should be pointed out that in this case the uniform quantizer is not optimal so that this result is consistent with Kollár's development, but differs from Bucklew and Gallagher.

It should be noted that the Sripad and Snyder conditions yield exact results rather than approximations, but the bin width Δ is essentially assumed to be fixed. Furthermore, because the derivation involves a Fourier series expansion of the quantizer error probability density function on $[-\Delta/2, \Delta/2]$, the derivation implicitly assumes that the quantizer does not overload, that is, that all of the nonzero probability density resides in the no-overload region.

We have seen a variety of conditions under which various aspects of the white-noise approximation are true approximately or exactly. The question now is whether these conditions are relevant to $\Delta\Sigma$. First consider the Bennett conditions.

1. Is the input in the no-overload region?

Often this is not known, a problem especially true if the quantizer is inside a feedback loop. This must be verified for a particular $\Delta\Sigma$ modulator architecture and is in fact known only for a few.

2. Is M asymptotically large?

This is almost never true in $\Delta\Sigma$, where typically $M = 2$, which is not large by any stretch of the imagination.

3. Is Δ asymptotically small?

This is almost never true in $\Delta\Sigma$, where typically Δ is as large as the allowed input signal range.

4. Is the joint density of the input signal at different sample times smooth?

This is never true in $\Delta\Sigma$ where the input to the quantizer includes a discrete component due to the feedback from the quantizer. Thus the pdf of the quantizer input has a continuous component and an impulsive component, violating the smoothness condition used to prove the Bennett theorem.

If the Bennett theorem cannot be made to apply, the next alternative is to test the Sripad and Snyder conditions. This is not easily done, however, because in a $\Delta\Sigma$ modulator the input signal to the quantizer includes both an original signal and a fed back output of the quantizer, as well as linear filtering. Hence the Sripad and Snyder conditions cannot be tested without solving for the quantizer input density. This is in fact close to the method that will be used to approach the problem.

Given the above observations, the white-noise approximation is at best suspicious and at worst simply wrong in $\Delta\Sigma$ analysis. Not surprisingly, it was found early in some oversampled ADCs (such as the simple single-loop $\Delta\Sigma$ modulator) that the quantizer noise was not at all white and that the noise contained discrete spikes whose amplitude and frequency depended on the input [9, 10]. Perhaps surprisingly, however, simulations and actual circuits of higher order $\Delta\Sigma$ modulators and of interpolative coders often exhibited quantizer noise that appeared to be approximately white. Unfortunately, these systems also often exhibited unstable behavior not predicted by the white-noise analysis.

The approximation can be modified to attempt to better approximate quantization error. Common approaches involve replacing the quantizer by a linear gain using describing function analysis. (See, e.g., [11–15].) The remaining error can then be approximated by input independent white noise, an approach introduced by Booton [16, 17] and applied to $\Delta\Sigma$ modulators (with additional linear filtering permitted in the loops) as in Ardalan and Paulos [18].

One can improve the approximations by using higher order terms in various expansions of the quantizer nonlinearity, but this approach has not been noticeably successful in the ADC application, primarily because of its difficulty. In addition, traditional power series expansions are not well suited to the discontinuous nonlinearities of quantizers. (See Arnstein [19] and Slepian [20] for series expansion solutions for quantization noise in delta modulators and differential pulse code modulation [DPCM].)

Another approach to the analysis of quantization error is to modify the system by adding a small random signal or dither at the input of the quantizer. By suitably choosing the dither signal, one can in some cases force the quantization error to satisfy all aspects of the white-noise approximation, but at the possible cost of corrupting the signal and reducing the allowed no-overload range [21–24].

The approach taken here is a variation of the classical characteristic function method of Rice [25] and the transform method of Davenport and Root [26], who represented memoryless nonlinearities using Fourier or Laplace transforms. A similar application of Fourier analysis was made to quantization noise analysis by Clavier et al. [27, 28], whose work was contemporary to Bennett's but did not have the impact of the latter. They provided an exact analysis of the quantizer noise resulting when a uniform quantizer is driven by one or two

sinusoids and thereby demonstrated both that quantization noise could behave quite unlike the predictions of the Bennett theory and that in some cases the behavior of such noise could be exactly quantified.

Subsequently the characteristic function method was applied to the study of quantization noise by Widrow [29], and his formulation has been used in the subsequent development of conditions under which the quantization noise is white, in particular by Sripad and Snyder [3] and the classic work of Iwerson on delta modulation [30].

Combining the characteristic function method with solutions to nonlinear difference equations and some basic results from nonlinear dynamical systems theory, we can obtain an exact analysis of several interesting $\Delta\Sigma$ modulators.

2.4 CHARACTERISTIC FUNCTION METHOD

Consider now a quantizer input u , quantizer output $q(u)$, and quantizer error $\epsilon = q(u) - u$. It is convenient to normalize the quantizer output and input by the bin width Δ (or equivalently to assume $\Delta = 1$) and write after some algebra

$$e = \frac{\epsilon}{\Delta} = \frac{1}{2} - \left\langle \frac{u}{\Delta} \right\rangle \quad (2.10)$$

where $\langle r \rangle$ denotes the fractional part of r (or $r \bmod 1$). The function $e(u)$ can be expanded as a Fourier series for u in the no-overload region as

$$e = e(u) = \sum_{l \neq 0} \frac{1}{2\pi j l} e^{2\pi j l \frac{u}{\Delta}} = \sum_{l=1}^{\infty} \frac{1}{\pi l} \sin\left(2\pi l \frac{u}{\Delta}\right) \quad (2.11)$$

This series will hold for almost all values of u in the no-overload region, and it is a key formula for all of the subsequent analysis. One can similarly write a Fourier series for e^2 as

$$e(u)^2 = \frac{1}{12} + \sum_{l \neq 0} \frac{1}{2(\pi l)^2} e^{2\pi j l \frac{u}{\Delta}} = \frac{1}{12} + \sum_{l=1}^{\infty} \frac{1}{2(\pi l)^2} \cos\left(2\pi l \frac{u}{\Delta}\right) \quad (2.12)$$

Suppose now that a sequence u_n is put into the quantizer, where we require that $|u_n| \leq M\Delta/2$. We wish to study the behavior of the normalized error sequence $e_n = \epsilon_n/\Delta$. From Eqs. (2.10) and (2.11) it is immediate that

$$e_n = \frac{1}{2} - \langle u_n \rangle = \sum_{l \neq 0} \frac{1}{2\pi j l} e^{2\pi j l \frac{u_n}{\Delta}} = \sum_{l=1}^{\infty} \frac{1}{\pi l} \sin\left(2\pi l \frac{u_n}{\Delta}\right) \quad (2.13)$$

For some specific examples of sequences u_n , Eq. (2.13) can be used to obtain a form of Fourier series representation directly for the sequence e_n . We take a more direct route and focus on second-order properties (mean, correlation, spectra) rather than a complete characterization of the sequence. Here the primary interest is the long-term average behavior

of the error sequence e_n . In particular we look at the average mean, second moment, and autocorrelation function. As we also wish to consider probabilistic expectations when dealing with random inputs such as dithered inputs, it is useful to consider averages that include both time and probabilistic averages. A useful formalism for simultaneously considering such averages is the class of *quasi-stationary processes* considered by Ljung [31, 32]. A discrete-time process e_n is said to be *quasi-stationary* if there is a finite constant C such that $E(e_n) \leq C$ for all n ; and $|R_e(n, k)| \leq C$ for all n, k , where $R_e(n, k) = E(e_n e_k)$; and if for each k the limit

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N R_e(n, n+k) \quad (2.14)$$

exists, in which case the limit is defined as $R_e(k)$. Following Ljung we introduce some notation: Given a process x_n , define

$$\bar{E}\{x_n\} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E(x_n) \quad (2.15)$$

if the limit exists. Thus for a quasi-stationary process $\{e_n\}$ the autocorrelation is given by

$$R_e(k) = \bar{E}\{e_n e_{n+k}\} \quad (2.16)$$

the mean is defined by

$$m_e = \bar{E}\{e_n\} \quad (2.17)$$

and the average power is given by

$$R_e(0) = \bar{E}\{e_n^2\} \quad (2.18)$$

Other moments are similarly defined. These moments reduce to the corresponding time averages or probabilistic averages in the special cases of deterministic or random processes, respectively.

The *power spectrum* of the process is defined in the general case as the discrete-time Fourier transform of the autocorrelation:

$$S_e(f) = \sum_{n=-\infty}^{\infty} R_e(n) e^{-2\pi jfn} \quad (2.19)$$

where the frequency f is normalized to lie in $[0, 1]$. The usual linear system input/output relations hold for this general definition of spectrum (see Chapter 2 of Ljung [31]). In fact, the class of quasi-stationary processes can be viewed as the most general class for which the familiar formulas of ordinary linear system second-order correlation and spectral analysis remain valid.

We now proceed to apply the basic formulas (2.13) and (2.12) to find an expression for the moments. Plugging Eq. (2.13) into (2.17) and Eq. (2.12) into (2.18) and assuming that the limits can be interchanged results in

$$\begin{aligned} \bar{E}\{e_n\} &= \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \sum_{l \neq 0} \frac{1}{2\pi j l} e^{2\pi j l \frac{u_n}{\Delta}} = \sum_{l \neq 0} \frac{1}{2\pi j l} \bar{E}\left\{e^{2\pi j l \frac{u_n}{\Delta}}\right\} \\ \bar{E}\{e_n^2\} &= \frac{1}{12} + \sum_{l \neq 0} \frac{1}{2(\pi l)^2} \bar{E}\left\{e^{2\pi j l \frac{u_n}{\Delta}}\right\} \end{aligned}$$

and for $k \neq 0$

$$R_e(k) = \sum_{i \neq 0} \sum_{l \neq 0} \frac{j}{2\pi i} \frac{j}{2\pi l} \bar{E}\left\{e^{2\pi j \left(i \frac{u_n}{\Delta} + l \frac{u_{n+k}}{\Delta}\right)}\right\}$$

These expressions can be most easily given in terms of the one-dimensional characteristic function

$$\bar{\Phi}_u(l) = \bar{E}\left\{e^{2\pi j l \frac{u_n}{\Delta}}\right\} \quad (2.20)$$

and a two-dimensional characteristic function

$$\bar{\Phi}_u^{(k)}(i, l) = \bar{E}\left\{e^{2\pi j \left(i \frac{u_n}{\Delta} + l \frac{u_{n+k}}{\Delta}\right)}\right\} \quad k \neq 0 \quad (2.21)$$

as

$$\bar{E}\{e_n\} = \sum_{l \neq 0} \frac{1}{2\pi j l} \bar{\Phi}_u(l) \quad (2.22)$$

$$\bar{E}\{e_n^2\} = \frac{1}{12} + \sum_{l \neq 0} \frac{1}{2(\pi l)^2} \bar{\Phi}_u(l) \quad (2.23)$$

and for $k \neq 0$

$$R_e(k) = - \sum_{i \neq 0} \sum_{l \neq 0} \frac{1}{2\pi i} \frac{1}{2\pi l} \bar{\Phi}_u^{(k)}(i, l) \quad (2.24)$$

The interchange of the limits is an important technical point that must be justified in any particular application.

If the characteristic functions of Eqs. (2.20) and (2.21) can be evaluated, then the moments and spectrum of the process can be computed from Eqs. (2.22)–(2.24).

2.5 PULSE CODE MODULATION QUANTIZATION NOISE

First consider a purely deterministic input to a simple quantizer with M levels, a simple pulse code modulation (PCM) system with no feedback. We will not consider in detail the

example of a dc input to an ordinary uniform quantizer in any detail because the results are trivial. We consider a more interesting (and active) input, a sinusoid $u_n = A \sin(n\omega_0 + \theta)$ with a fixed initial phase θ . We assume that $A \leq M/2$ so that the quantizer is not overloaded. Define also $f_0 = \omega_0/2\pi$.

For the given purely deterministic example, the one-dimensional characteristic function can be expressed as

$$\bar{\Phi}_u(l) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N e^{j2\pi l \gamma \sin(n\omega_0 + \theta)} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N e^{j2\pi l \gamma \sin\left(2\pi\left(nf_0 + \frac{\theta}{2\pi}\right)\right)} \quad (2.25)$$

where the fractional part can be inserted since $\sin(2\pi u)$ is a periodic function in u with period 1.

This limit can be evaluated, but the result depends strongly on what is assumed about the input signal. As this is a delicate and often misunderstood issue that will arise often in the analysis of quantization systems, it merits some discussion. If we choose f_0 to be a rational number, say K/N in lowest terms, then clearly $\sin(n2\pi f_0 + \theta)$ is periodic with period N since $\sin[(n+N)2\pi f_0 + \theta] = \sin(n2\pi f_0 + K2\pi + \theta) = \sin(n2\pi f_0 + \theta)$. In this case it is also true that $e^{j2\pi l \gamma \sin(n\omega_0 + \theta)}$ is periodic, and hence the limiting sum becomes a finite sum over a single period. It is also immediately true in this case that the quantizer output and the quantizer error signal are also periodic. If, however, the frequency is chosen at random according to a probability density function, then the probability of it being a rational number will be 0 and the probability of it being an irrational number will be 1. In this case the discrete-time input signal is not periodic (it is an example of what is called an “almost periodic function” [33]). We henceforth consider only this case in detail because it is typical when the frequency is selected at random from a continuous distribution.

When the frequency is irrational, the above limit can be evaluated using a classical result in ergodic theory of Hermann Weyl (see, e.g., Petersen [34]): If g is an integrable function, a is an irrational number, and b is any real number, then

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N g(\langle an + b \rangle) = \int_0^1 g(u) du \quad (2.26)$$

This remarkable result follows since the sequence of numbers $\langle an + b \rangle$ uniformly fills the unit interval and hence the sums approach an integral in the limit. Applying Eq. (2.26) to (2.25) yields

$$\bar{\Phi}_u(l) = \int_0^1 e^{j2\pi l \gamma \sin(2\pi u)} du = J_0(2\pi l \gamma) \quad (2.27)$$

where $\gamma = A/\Delta$ and J_m is the ordinary Bessel function of order m .

The mean and second moment of the quantizer noise can then be found using the fact that $J_0(r) = J_0(-r)$:

$$\bar{E}\{e_n\} = \sum_{l \neq 0} \frac{1}{2\pi j l} J_0(2\pi l \gamma) = 0 \quad (2.28)$$

$$\bar{E}\{e_n^2\} = \frac{1}{12} + \frac{1}{\pi^2} \sum_{l=1}^{\infty} \frac{1}{l^2} J_0^2(2\pi l \gamma) \quad (2.29)$$

Note that the result does not depend on the frequency of the input sinusoid (provided the frequency is an irrational number) and that the time average mean is 0, which agrees with that predicted by the assumption that e_n is uniformly distributed on $[-\Delta/2, \Delta/2]$. The second moment, however, differs from the value of $\frac{1}{12}$ predicted by the uniform assumption by the right-hand sum of weighted Bessel functions. Note that if $\gamma = A/\Delta$ becomes large (which with A held fixed and the no-overload assumption means that the number of quantization levels is becoming large), then $J_0(2\pi l \gamma) \rightarrow 0$ and hence the second moment converges to $\frac{1}{12}$ in the limit.

To compute the autocorrelation of the quantization noise, we use similar steps to find the joint characteristic function $\bar{\Phi}_u^{(k)}(i, l)$ as

$$R_e(k) = \sum_{n=-\infty}^{\infty} S_n e^{2\pi j k \lambda_n} \quad (2.30)$$

where $\lambda_n = \langle (2n-1)\omega_0/2\pi \rangle$ are normalized frequencies in $[0, 1)$ and

$$S_n = \left[\frac{1}{\pi} \sum_{l=1}^{\infty} \frac{J_{2n-1}(2\pi l \gamma)}{l} \right]^2 \quad (2.31)$$

are the spectral components at the frequency λ_n . Thus

$$S_e(f) = \sum_n S_n \delta(f - \lambda_n) \quad (2.32)$$

where $\delta(f)$ denotes a Dirac delta function.

The spectrum of the quantizer error therefore is purely discrete and consists of all odd harmonics of the fundamental frequency of the input sinusoid. The energy at each harmonic depends in a very complicated way on the amplitude of the input sinusoid. In particular, the quantizer noise is decidedly not white since it has a discrete spectrum and since the spectral energies are not flat. Thus here the white-noise approximation of Bennett and of the describing function approach is invalid, even if M is large. Claasen and Jongepier [35] argued that if the spectrum analyzer has limited resolution, then one can make assumptions about the statistical behavior of the coefficients in the spectrum which leads to an approximately white spectrum. Indeed a short-term fast Fourier transform (FFT) will look somewhat white, but higher resolution will clearly show the discrete nature of the error. In the $\Delta\Sigma$ case to be considered, even short-term FFTs clearly show the spikes and the corresponding tones can be heard in audio reconstructions.

The cross correlation is handled in a similar fashion to obtain

$$R_{ue}(k) = \sum_{l \neq 0} \bar{E}\left\{u_n e^{j2\pi l \frac{u_{n+k}}{\Delta}}\right\} = A \cos(k\omega_0) \sum_{l=1}^{\infty} \frac{J_1(2\pi l \gamma)}{l} \quad (2.33)$$

which is not equal to the product of the means since the error mean is 0. Thus the error and the input are not asymptotically uncorrelated.

The basic procedure used above of computing characteristic functions, which in turn yield the quantization error moments and spectra, can be used with more complicated input signals to obtain exact formulas that can be evaluated numerically.

2.6 DITHERED PCM

We next consider a quantizer input process of the form $u_n = x_n + w_n$, where x_n is the possibly nonstationary original system input (such as the deterministic sinusoid previously considered) and w_n is an i.i.d. random process called a *dither* process. A key attribute of the dither process is that it is independent of the x_n process, that is, x_n is independent of w_k for all times n and k . We still require that the quantizer input u_n be in the no-overload region. This has the effect of reducing the allowed input dynamic range and hence limiting the overall signal-to-quantization noise ratio (SQNR). Dithering has long been used as a means of improving the subjective quality of quantized speech and images (see Jayant and Noll, Section 2.8, and the references therein [36]). The principal theoretical property of dithering was developed by Schuchman [22], who proved that if the quantizer does not overload and the characteristic function of the marginal probability density function of the dither signal is 0 at integral multiples of $(2\pi)/\Delta$, then the quantizer error $\varepsilon_n = q(x_n + w_n) - (x_n + w_n)$ is independent of the original input signal x_n . It follows from Sripad and Snyder [3] that under these conditions the quantization error is also white. See, for example, [24, 35]. It is not true, however, that the quantization noise $q(x_n + w_n) - x_n$ is independent of signal or white (a common misconception that is still found in some texts, see [37, 24] for a discussion).

Given a stationary random process w_n , recall the definition of Eq. (2.7) of $\Phi_w(\alpha) = E(e^{j\alpha w_n})$, the ordinary characteristic function of w_n . Because of the independence of the processes, the one-dimensional characteristic function of Eq. (2.20) becomes

$$\bar{\Phi}_u(l) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E \left(e^{2\pi j l \frac{1}{\Delta} (x_n + w_n)} \right) = \Phi_w \left(2\pi \frac{l}{\Delta} \right) \bar{\Phi}_x(l) \quad (2.34)$$

The two-dimensional characteristic function of Eq. (2.21) is

$$\begin{aligned} \bar{\Phi}_u^{(k)}(i, l) &= \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E \left(e^{2\pi j \frac{1}{\Delta} [i(x_n + w_n) + l(x_{n+k} + w_{n+k})]} \right) \\ &= \Phi_w \left(2\pi \frac{1}{\Delta} i \right) \Phi_w \left(2\pi \frac{1}{\Delta} l \right) \bar{\Phi}_x^{(k)}(i, l) \quad k \neq 0 \end{aligned} \quad (2.35)$$

Now suppose that the marginal distribution of w_n is such that Schuchman's conditions are satisfied, that is, the quantizer is not overloaded and

$$\Phi_w \left(2\pi \frac{1}{\Delta} l \right) = 0 \quad l = \pm 1, \pm 2, \pm 3, \dots \quad (2.36)$$

[Recall that $\Phi_w(0) = 1$ for any distribution.] This is the condition shown by Schuchman to be necessary and sufficient for the quantization error to be independent of the original input x_n . The principal example is a dither signal with a uniform marginal on $[-\Delta/2, \Delta/2]$ (and an input amplitude constrained to avoid overload when added to the dither) or sums of independent uniform variates. For this case we have

$$\bar{\Phi}_u(l) = \begin{cases} 1 & l = 0 \\ 0 & \pm 1, \pm 2, \dots \end{cases} \quad (2.37)$$

and for $k \neq 0$

$$\bar{\Phi}_u^{(k)}(i, l) = \begin{cases} \bar{\Phi}_x^{(k)}(0, 0) = 1 & i = l = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.38)$$

Thus in this example we have from Eq. (2.22) to (2.24) that e_n has zero mean, a second moment of $\frac{1}{12}$, and an autocorrelation function $R_e(k) = 0$ when $k \neq 0$, that is, the quantization error is indeed white when Schuchman's condition is satisfied. This is true for a general quasi-stationary input, including the sinusoid previously considered. Observe that Eq. (2.38) is in fact a Sripad and Snyder condition for the generalized characteristic function $\bar{\Phi}$ for the quantizer input process. Since it is obtained by multiplying the characteristic function of the input by those for i.i.d. uniform random variables, the resulting dithered signal results in uniform white quantization error. If the input sequence is $A \sin(n\omega_0)$ as before and the dither sequence is an i.i.d. sequence of uniform random variables on $[-\Delta/2, \Delta/2]$, then the overload condition becomes $A/\Delta \leq M - \frac{1}{2}$, which has effectively reduced the allowable γ .

A similar exercise shows that the error and input are uncorrelated. Additional effort is needed to prove independence (see, e.g., [24]).

Although dithering yields a quantizer error with nice statistical properties, it corrupts the signal (unless subtractive dither is used) and reduces the SQNR achievable with a given quantizer since the input amplitude must be reduced enough so that the original signal plus the dither stays within the no-overload region. This loss may be acceptable (and small) when the number of quantization levels is large. It is significant if there are only a few quantization levels. For example, if $M = 2$, then a uniform dither on $[-\Delta/2, \Delta/2]$ can only avoid overload if the signal is confined to have magnitude less than $\Delta/2$.

2.7 SINGLE-LOOP $\Delta\Sigma$ MODULATION

The basic $\Delta\Sigma$ modulator can be motivated by an intuitive argument based on the dithering idea. Suppose that instead of adding an i.i.d. random process to the signal before quantization, the quantization noise itself is used as a dither signal; that is, i.i.d. signal-independent noise is replaced by deterministic signal-dependent noise that (hopefully) approximates a white signal-independent process. Reversing the noise sign for convenience and inserting a delay in the forward path of the feedback loop (to reflect the physical delay inherent in a

quantizer) yields the system described by the nonlinear difference equation

$$u_n = x_{n-1} - \varepsilon_{n-1} = u_{n-1} + x_{n-1} - q(u_{n-1}) \quad n = 1, 2, \dots \quad (2.39)$$

This difference equation is equivalent to the traditional discrete-time form of a single-loop $\Delta\Sigma$ modulator, which can therefore be thought of as a deterministically dithered PCM system, an idea introduced in 1960 by Cutler in Figure 2 of [38], where he referred to the system as a quantizer “with a single step of error compensation.” The name delta-sigma modulator was introduced by Inose and Yasuda in 1963 [39], who provided the first published description of its basic properties. The name was intended to reflect the fact that the system first took a difference (Δ) and then integrated (Σ). The modern popularity of these systems, much of the original analysis, and the alternative name sigma-delta modulator are due to Candy and his colleagues [10, 40–43]. The name $\Sigma\Delta$ reflects the fact that the system can also be represented as the cascade of an integrator (Σ) and a Δ modulator. In the author’s opinion, this is a better name because the system does not really form a difference of successive samples of the input signal as suggested by the $\Delta\Sigma$ name; it forms the difference between the input and a digital approximation of the previous input that is fed back. The name $\Delta\Sigma$ does not incorporate the key attribute of quantization in the system; the reverse order does. The author bows to the majority of co-authors, however, and adopts the older name.

Given the interpretation of the system as a deterministically dithered quantizer, one might hope that the deterministic dither might indeed yield a white quantization noise process, but unfortunately this circular argument does not hold for the simple single-loop system, as will be seen.

Since $u_n = q(u_n) - \varepsilon_n$, Eq. (2.39) yields the difference equation

$$q(u_n) = x_{n-1} + \varepsilon_n - \varepsilon_{n-1} \quad (2.40)$$

which has the intuitive interpretation that the quantizer output can be written as the input signal (delayed) plus a difference (or discrete-time derivative) of an error signal. The hope is that this difference will be a high-frequency term that can be removed by low-pass filtering to obtain the original signal. For convenience we assume that $u_0 = 0$ and we normalize the above terms by Δ and use the definition of ε_n to write

$$e_n = \frac{\varepsilon_n}{\Delta} = \frac{q(u_n) - u_n}{\Delta} = \frac{q(x_{n-1} - \varepsilon_{n-1}) - x_{n-1}}{\Delta} - e_{n-1} \quad n = 1, 2, \dots \quad (2.41)$$

Since $u_0 = 0$, $\varepsilon_0 = \Delta/2$.

We shall assume that the input range is $[-b, b)$, that is, $-b \leq x_n < b$ for all n . Intuitively, we would like to make Δ small in order to keep the quantizer error small; but we dare not make it too small or the quantizer may overload (M is considered fixed). An easy induction argument in [44, 45] shows that the smallest value of Δ as a function of b for which overload never occurs is $\Delta = 2b/(M-1)$. In the most common case of a binary quantizer, $\Delta = 2b$. This implies that for Δ chosen in this manner, the Bennett condition of not overloading the quantizer is met for the simple single-loop $\Delta\Sigma$ modulator.

To find an explicit expression for e_n in terms of the x_n , sum Eq. (2.40) from $k = 1$ to n :

$$\sum_{k=1}^n \frac{q(u_k)}{\Delta} = \sum_{k=1}^n \frac{x_{k-1}}{\Delta} + \sum_{k=1}^n e_k - \sum_{k=1}^n e_{k-1} = \sum_{k=1}^n \frac{x_{k-1}}{\Delta} + e_n - \frac{1}{2}$$

Define $1(u) = 1$ if $u \geq 0$ and 0 otherwise so that $q(u)/\Delta = 1(u) - \frac{1}{2}$, and we have for $n = 1, 2, \dots$

$$y_n = \frac{1}{2} - e_n = \sum_{k=1}^n \left(\frac{x_{k-1}}{\Delta} + \frac{1}{2} \right) - \sum_{k=1}^n 1(u_{k-1})$$

(y_n is more convenient to deal with than e_n .)

Taking the fractional part of both sides yields

$$\langle y_n \rangle = \left\langle \sum_{k=1}^n \left(\frac{x_{k-1}}{\Delta} + \frac{1}{2} \right) \right\rangle$$

If $-b \leq x_n < b$ for all n , then $y_n \in [0, 1)$ and hence $\langle y_n \rangle = y_n$. Thus $y_0 = 0$ and

$$y_n = \left\langle \sum_{k=0}^{n-1} \left(\frac{1}{2} + \frac{x_k}{\Delta} \right) \right\rangle = \left\langle \frac{n}{2} + \sum_{k=0}^{n-1} \frac{x_k}{\Delta} \right\rangle \quad n = 1, 2, \dots$$

and hence

$$e_n = \frac{1}{2} - \left\langle \frac{n}{2} + \sum_{k=0}^{n-1} \frac{x_k}{\Delta} \right\rangle \quad (2.42)$$

Compare this with PCM case $e_n = \frac{1}{2} - \langle u_n/\Delta \rangle$.

When the quantizer is put into a feedback loop with an integrator, the overall effect is to integrate the input plus a constant bias before taking the fractional part. The overall nonlinear feedback loop therefore appears as an affine operation (linear plus a bias) on the input followed by a memoryless nonlinearity.

The techniques used to find the time average moments for e_n in the memoryless quantizer case can now be used by replacing u_n by the sum

$$s_n = \sum_{k=0}^{n-1} \left(\frac{1}{2} + \frac{x_k}{\Delta} \right) \quad (2.43)$$

evaluating the characteristic functions of Eqs. (2.20) and (2.21) and applying Eqs. (2.22) to (2.24) for Φ_s instead of Φ_u . Thus,

$$\bar{\Phi}_s(l) = \bar{E} \left\{ e^{\pi j l k} e^{2\pi j l \sum_{m=0}^{n-1} x_m} \right\} \quad (2.44)$$

$$\bar{\Phi}_s^{(k)}(i, l) = e^{\pi j l k} \bar{E} \left\{ e^{\pi j (i+l)n} e^{2\pi j (i+l) \sum_{m=0}^{n-1} x_m} e^{2\pi j l \sum_{m=n}^{n-1+k} x_m} \right\} \quad (2.45)$$

To evaluate these limits it is necessary as in the PCM case to assume a particular form for the input signal. A simple but important signal is a dc value: $x_k = x$ for all k , where $-b \leq x < b$ is fixed. Although clearly of limited practical application, it can be considered as an approximation to a very slowly varying input, that is, to the case where the $\Delta\Sigma$ modulator has a large oversampling ratio, as it might for sensor measurements.

Analogous to the analysis for PCM with a sinusoidal input, there are two possible assumptions on the dc value which lead to nice solutions, but which have fundamentally different behavior and interpretation. The assumption that we shall make is that $x/2b$ is an irrational number. This assumption is physically and intuitively correct for an ADC because any truly analog random signal will be describable by a probability density function and hence with probability 1 will produce an irrational number. As in the PCM case, choosing $x/2b$ irrational will permit the evaluation of the above limits using asymptotic results from ergodic theory such as Weyl's theorem [Eq. (2.26)]. If on the other hand it is assumed that $x/2b$ is a rational number, the limits become finite sums and the output and error signals become periodic; that is, "tones" or "limit cycles" are produced. Much of the analysis described here can be modified for rational inputs. For example, the analysis for single-loop $\Delta\Sigma$ for rational dc inputs may be found in [46, 47]. We do not pursue the analysis for rational inputs here because in the author's view it is of little interest in describing the behavior of ADC systems to analyze carefully behavior resulting from zero probability inputs. For further discussion on the issue of irrational vs. rational dc values see Iwersen [48]. This presents a potential cause for confusion, however, because simulations of ADC behavior on a digital computer will necessarily produce rational input signals and hence the resulting periodic behavior will appear to disagree with the theory. The reconciliation of this apparent paradox is to make sure that the simulations well approximate the assumptions required by the theory. If a rational dc is selected with a modest denominator (e.g., a few hundred), then the assumption of an irrational input is clearly violated and the resulting signals will indeed be periodic and the various statistics poor approximations to the theory. If one instead generates a random number using a uniform random number generator on a digital computer, the resulting number will still be rational, but it will be "approximately" irrational in that with high probability the fraction in lowest terms will have a denominator that is extremely large (hundreds of thousands or millions). This means that the resulting signals will be periodic, but with extremely long periods so that spectral analyzers will not see the periodicities. All statistics computed will well match the theory in this case.

One system where the assumption of a rational dc input signal is valid is a digital-to-analog converter (DAC) since by definition a digital input signal can take on only rational values. Hence the results developed here for irrational inputs do not apply to the analysis of $\Delta\Sigma$ modulators for DACs. The basic methods can still be used, but the asymptotic results must be replaced by appropriate finite sums, and the answers will be different from those of the irrational case.

Assuming an irrational dc input x , we can replace u_n by $s_n = n\beta$, where $\beta = (\frac{1}{2} + x/\Delta)$, in Eqs. (2.20) and (2.21) and evaluate the characteristic functions using Eq. (2.26):

$$\bar{\Phi}_s(l) = \int_0^1 du e^{j2\pi lu} = \begin{cases} 0 & l = \pm 1, \pm 2, \dots \\ 1 & l = 0 \end{cases} \quad (2.46)$$

$$\bar{\Phi}_s^{(k)}(i, l) = \begin{cases} e^{2\pi lk\beta} & i = -l \\ e^{2\pi lk\beta} \int_0^1 du e^{j2\pi u} = 0 & \text{otherwise} \end{cases} \quad (2.47)$$

Thus we have from Eqs. (2.22) and (2.23) that $\bar{E}\{e_n\} = 0$ and $\bar{E}\{e_n^2\} = \frac{1}{12}$, which agrees with the uniform noise approximation; that is, these are exactly the time average moments one would expect with a sequence of uniform random variables. The second-order properties, however, are quite different. From Eq. (2.24) and 1.443.3 of [49]

$$R_e(k) = \sum_{l \neq 0} \left(\frac{1}{2\pi l}\right)^2 e^{j2\pi lk\beta} = \frac{1}{2} \frac{1}{\pi^2} \sum_{l=1}^{\infty} \frac{\cos(2\pi lk\beta)}{l^2} = \frac{1}{12} - \frac{\langle k\beta \rangle}{2} (1 - \langle k\beta \rangle) \quad (2.48)$$

This does not correspond to a white process. The exponential expansion implies that the spectrum is purely discrete having amplitude

$$S_n = \begin{cases} 0 & \text{if } n = 0 \\ \frac{1}{(2\pi n)^2} & \text{if } n \neq 0 \end{cases} \quad (2.49)$$

at frequencies $\langle n\beta \rangle = \langle n(\frac{1}{2} + x/\Delta) \rangle$. Thus the locations and hence the amplitude of spikes of the quantizer error spectrum depend strongly on the value of the input signal. Thus as in the simple PCM case with a sinusoidal input, the Bennett and describing function white-noise approximations inaccurately predict the spectral nature of the quantizer noise process, which is neither continuous nor white.

Next consider a more "active" input $x_n = A \cos n\omega_0$, where $\omega_0/2\pi$ is assumed to be irrational and where we consider a full-scale sinusoid with $|A| = b$ as an example. Define $\alpha = \gamma/2 \sin(\omega_0/2)$. The same general procedure with a lot more algebra [50] now results in

$$m_e = 0 \quad (2.50)$$

$$R_e(0) = \frac{1}{12} - \sum_{l=1}^{\infty} \frac{1}{(\pi 2l)^2} (-1)^l J_0(4\pi l\alpha) \quad (2.51)$$

$$R_y(k) = \sum_{m=-\infty}^{\infty} S_m e^{j2\pi k\lambda_m} \quad (2.52)$$

where

$$S_m = \begin{cases} \frac{1}{2} & m = 0 \\ \left(\frac{1}{\pi} \sum_{l=1}^{\infty} \frac{J_m[2\pi\alpha(2l-1)]}{2l-1} (-1)^l \right)^2 & m \text{ even} \\ \left(\frac{1}{\pi} \sum_{l=1}^{\infty} \frac{J_m(4\pi\alpha l)}{2l} (-1)^l \right)^2 & m \text{ odd} \end{cases} \quad (2.53)$$

and

$$\lambda_m = \begin{cases} \left\langle m \frac{\omega_0}{2\pi} - \frac{1}{2} \right\rangle & m \text{ even} \\ \left\langle m \frac{\omega_0}{2\pi} \right\rangle & m \text{ odd} \end{cases} \quad (2.54)$$

With a sinusoidal input, the input and quantizer error are *not* uncorrelated.

As in the PCM case, the spectrum of y_n is purely discrete and has amplitude s_l at the frequency λ_l . This spectrum is extremely nonwhite since it is not continuous and not flat. The output frequencies depend on the input frequency ω_0 and comprise all harmonics of the input frequency ω_0 . It is interesting to observe that not only are all harmonics of the input frequency contained in the output signal but also all shifts of these harmonics by π (when computed in radians). These shifted harmonics are not present in the PCM case.

2.8 TWO-STAGE (CASCADE OR MASH) $\Delta\Sigma$ MODULATION

We now turn to the two-stage cascaded or MASH (multi-stage noise-shaping) $\Delta\Sigma$ modulator. Here two $\Delta\Sigma$ modulators are cascaded and the final output formed by a linear combination of the individual outputs. In particular, suppose that the first $\Delta\Sigma$ has input x_n , integrator state v_n , output $q(v_n)$, and quantizer error signal $\zeta_n = q(v_n) - v_n$. This first $\Delta\Sigma$ is defined by the difference equation

$$v_n = v_{n-1} + x_{n-1} - q(v_{n-1}) \quad n = 1, 2, \dots \quad (2.55)$$

The first-stage error sequence ζ_n is the input to the second loop, where the integrator state is denoted u_n and the quantizer error is denoted ϵ_n . The difference equation for the second stage is then

$$u_n = u_{n-1} + \zeta_{n-1} - q(u_{n-1}) \quad n = 1, 2, \dots \quad (2.56)$$

The quantizers are exactly as in Section 2.3, that is, uniform quantizers with M levels and bin width $\Delta = 2b/(M-1)$, where the input x_n is between $-b$ and b . Note that this means that the first quantizer does not overload and hence ζ_n is in the range $[-b, b)$ and hence also the second quantizer does not overload. The output of the two-stage $\Delta\Sigma$ modulator is defined by

$$\psi_n = q(v_{n-1}) - q(u_n) + q(u_{n-1}) \quad n = 1, 2, \dots \quad (2.57)$$

a linear combination of the two quantizer outputs. As in Eq. (2.40) we have that $q(v_n) = x_{n-1} + \zeta_n - \zeta_{n-1}$ and $q(u_n) = \zeta_{n-1} + \epsilon_n - \epsilon_{n-1}$ and hence Eq. (2.57) becomes

$$\psi_n = x_{n-2} + \epsilon_n - 2\epsilon_{n-1} + \epsilon_{n-2} \quad (2.58)$$

In contrast to Eq. (2.40), this has the interpretation of being the original signal plus a second-order difference instead of the first-order difference of the single-loop system. Note that although this signal depends on the outputs of both stages, the first-stage quantization noise cancels out and only the second-stage noise remains. This fact is basic to the

operation of the system. In particular, we need only know the behavior of the second-stage quantization noise in order to find the behavior of the output. We shall see that the second-stage noise is better behaved than the first-stage noise.

Equation (2.58) suggests that a natural means of producing the reproduction signal $\hat{x}$ is to pass ψ_n through a low-pass filter and downsample.

Assuming that initially the integrator states (quantizer inputs) are $u_0 = v_0 = 0$, then applying Eq. (2.7) to both stages gives, for $n = 1, 2, \dots$,

$$p_n = \frac{\psi_n}{\Delta} = \left\langle \frac{1}{2} - \frac{n}{2} + \sum_{k=0}^{n-1} \frac{x_k}{\Delta} \right\rangle$$

$$e_n = \frac{\epsilon_n}{\Delta} = \left\langle \frac{1}{2} - \frac{n}{2} + \sum_{k=0}^{n-1} p_k \right\rangle = -\frac{1}{2} + \left\langle \sum_{l=0}^{n-1} l \left(\frac{1}{2} + \frac{x_{n-l}}{\Delta} \right) \right\rangle \quad (2.59)$$

As in the ordinary $\Delta\Sigma$ modulator, we can modify Eqs. (2.20)–(2.24) by replacing the quantizer input u_n by a sum term

$$s_n = \sum_{i=0}^{n-1} \sum_{l=0}^{i-1} \left(\frac{1}{2} + \frac{x_l}{\Delta} \right) = \sum_{l=0}^{n-1} l \left(\frac{1}{2} + \frac{x_{n-l}}{\Delta} \right) \quad (2.60)$$

and then proceed exactly as before. We here illustrate the results only for the simple case of an irrational dc input $x_n = x$. Define $\beta = \frac{1}{2} + x/\Delta$ as before and we have that $s_n = \beta/2n^2 - \beta/2n$. The limits in the characteristic functions are evaluated using a form of Weyl's theorem to obtain

$$\bar{\Phi}_s(l) = \begin{cases} 0 & l \neq 0 \\ 1 & l = 0 \end{cases} \quad (2.61)$$

$$\bar{\Phi}_s^{(k)}(i, l) = \begin{cases} 1 & i = l = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.62)$$

These characteristic functions are identical to those of the dithered PCM case in Eqs. (2.37) and (2.38), and hence the conclusions are the same: The generalized Sripad and Snyder conditions are met and the quantization error in the second stage is indeed white and its marginal first and second moments agree with those of a uniform distribution! Since only the second-stage error appears in the final reconstruction in an ideal system, the white-noise approximation can safely be used for SQNR analysis. It is perhaps surprising that such a purely deterministic system with a fixed dc input can produce a sequence that appears to be uniformly distributed white noise when its first- and second-order moments are measured. A slight variation on the foregoing analysis can be used to prove that the second-stage quantizer noise and the original input are asymptotically uncorrelated.

The production of a deterministic signal that masquerades as white noise is reminiscent of the theory of "chaos," the branch of nonlinear dynamical systems theory that focuses on transformations on points that produce sequences that appear to be random. There is, however, nothing chaotic about the quantization noise sequence. Technically, its Lyapunov exponent is zero and hence it is not chaotic. Chaos can be made to occur in

variations on the basic $\Delta\Sigma$ architecture, for example, by using an integrator in the feedback loop with gain greater than 1. See, for example, Feely and Chua [51, 52] and Schreier [53].

It should be reemphasized that the above analysis depended critically on the underlying assumption of an irrational dc input. If the input were rational, then the asymptotic limits would be replaced by finite sums and the behavior would be different. In particular, the error and output sequences would be periodic and the system would exhibit limit cycle behavior. As previously discussed, the periodic behavior would be evident if a rational input signal with a modest denominator is chosen in a simulation. Choosing the input using a good random number generator yields results that well approximate the asymptotic theory. It is also important to note that the solutions hold for the idealized system represented by the nonlinear difference equations. Real circuits would not have perfectly matched nonleaky integrators, which would result in behavior departing from the theory. In particular, one would expect to see discrete frequency components in such systems even for irrational dc inputs (as one does for a single-stage idealized system). The accuracy of the theory at predicting actual behavior for simulated or physical circuits depends strongly on the degree to which the simulations reflect the assumptions of the theory and the physical circuits implement the commonly used nonlinear difference equations used to describe the systems.

The analysis can be extended to the case of a sinusoidal input, but the analysis is much more complicated, the noise is not white [54], and it is not asymptotically uncorrelated with the input.

2.9 SECOND-ORDER $\Delta\Sigma$ MODULATION

Another $\Delta\Sigma$ system is the second-order multiloop $\Delta\Sigma$ introduced by Candy [41] and first rigorously analyzed by He et al. [55–58]. Here a single-loop $\Delta\Sigma$ modulator is embedded in a second loop with an integrator in the feedforward path. It can be interpreted as a first-order loop with the original input replaced by the integrated error between the input and the quantizer output. From this viewpoint, the second-order $\Delta\Sigma$ is equivalent to Cutler's quantizer with "two steps of error compensation" of Figure 3 of his 1960 patent application [38] except for the location of the delay.

The basic nonlinear difference equation for the quantizer input process u_n is given by

$$u_n = x_{n-1} - 2\varepsilon_{n-1} + \varepsilon_{n-2} \quad (2.63)$$

where, as before, $\varepsilon_n = q(u_n) - u_n$ is the quantizer error. Observe that the output of the second-order $\Delta\Sigma$ modulator is

$$q(u_n) = \varepsilon_n + u_n = \varepsilon_n - 2\varepsilon_{n-1} + \varepsilon_{n-2} + x_{n-1} \quad (2.64)$$

a relation which bears a remarkable resemblance to Eq. (2.58) for the output of the two-stage $\Delta\Sigma$ modulator (and hence is capable of the same interpretation).

A well-known difficulty with the second- (and higher) order $\Delta\Sigma$ modulators is their potential for quantizer overload. In particular, if one uses a binary quantizer with levels $\pm b$ in a second-order system, then it is easy to find an input within the range $[-b, b]$, which will overload the quantizer and hence will be capable of producing large errors. The

potential overload also has the serious consequence for our purposes that it renders invalid a basic technique of the approach used here. No application of the techniques of this chapter seems possible for the case of a binary quantizer, but the techniques do apply if we permit a two-bit (or higher) quantizer. It can be shown that the smallest value of Δ for which no overload occurs is given by [57] $\Delta = 2b/(M-3)$. Clearly this result is useful only if $M \geq 4$, that is, if the quantizer has at least 2 bits. For the present we make this assumption and we can then proceed as earlier.

As with the first-order loop analysis we normalize the error and then sum the difference equation twice (since there is a second-order difference) to find

$$y_n = \frac{1}{2} - \frac{\varepsilon_n}{\Delta} = \sum_{l=1}^n l \left(\frac{1}{2} + \frac{x_{n-1}}{\Delta} \right) - \sum_{l=1}^n l l (u_{n+1-l}) \quad (2.65)$$

In the special case of a dc input, Eq. (2.65) can be written as

$$\begin{aligned} y_n &= \sum_{l=1}^n l\beta - \sum_{l=1}^n l l (u_{n+1-l}) \\ &= \sum_{l=1}^n l[\beta - 1(u_{n+1-l})] \\ &= \sum_{l=1}^n (n+1-l)[\beta - 1(u_l)] \end{aligned} \quad (2.66)$$

where as before $\beta = (\frac{1}{2} + x/\Delta)$. As in the first-order case,

$$\langle y_n \rangle = \left\langle \sum_{l=1}^n l \left(\frac{1}{2} + \frac{x_{n-1}}{\Delta} \right) \right\rangle \quad (2.67)$$

If there is no overload, then $y_n = \langle y_n \rangle$ and the solution is identical to that of He et al. (and to that for the two-stage $\Delta\Sigma$ modulator [59]). The problem is that if the quantizer has only 1 bit, then it is not in general true that $y_n = \langle y_n \rangle$ and hence Eq. (2.67) does not provide a solution for the error. The analysis does characterize $\langle y_n \rangle$ as being a uniform white-noise sequence, but this is the fractional part of the quantizer error and not in general the quantizer error itself.

There is no simple solution to this problem. The failure of the analysis to apply to the 1-bit second-order system does not detract from the usefulness or popularity of the system; it only leaves open the issues of finding the properties of the quantizer error and of comparing those properties to those predicted by the white-noise approximation.

As will be considered in Chapter 5, exact results for the 1-bit second-order case have been developed and reported by many researchers, including Wang [60], Hein and Zakhori [61], and by Pinault and Lopresti [62]. Both Wang and Pinault and Lopresti used ideas from dynamical systems techniques to show that the two integrator states must eventually lie in a compact set, demonstrating a form of stability for the system. The

quantizer overloads, but an absolute bound on the integrator states (and hence the quantizer error) can be found as a function of the dc input. In particular their results provide a bound of the form

$$\left| \frac{1}{n} \sum_{k=1}^n (q(u_k) - x) \right| \leq \frac{C}{n} \quad (2.68)$$

where $C = \frac{5}{4}$ using the normalizations adopted here (it can be tightened to 1). It can further be shown that

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n \epsilon_n = 0$$

but the evaluation of the variance, correlation, and spectra remains an open problem.

2.10 SOME EXTENSIONS

2.10.1 Dithered Single-Loop $\Delta\Sigma$ Modulation

The methods can be applied to a dithered $\Delta\Sigma$ modulator with an input of the form

$$x_n = v_n + w_n \quad (2.69)$$

where w_n is an i.i.d. process that is independent of the quasi-stationary process v_n , and it is assumed that the input and dither are constrained within $[-b, b]$ to avoid quantizer overload. Again the characteristic functions can be found and used to show that [33] the error sequence has 0 mean and variance $\frac{1}{12}$, but that the spectrum of the error is not flat. For a suitable dither sequence, however, it is smooth and tends to become increasingly white as the dither signal is increased (and the input is correspondingly decreased to avoid overload). In the limit of 0 signal the quantization error becomes white. As always, these results depend strongly on the underlying assumptions. Here, for example, the conclusions cannot be trusted if the sum of the dither and signal are allowed to cause quantizer overload.

2.10.2 Multistage and Higher Order $\Delta\Sigma$ Modulation

The two-stage $\Delta\Sigma$ results can be extended to multiple stages with binary quantizers, where it can be shown that dc inputs, sinusoidal inputs, and sums of sinusoidal inputs all yield white quantization noise if the integrators are assumed to be ideal and the dc inputs or frequencies are required to be irrational [63]. Similarly, dithering multistage (two stages or more) $\Delta\Sigma$ modulators with i.i.d. noise, which does not cause overload, also yields white quantization noise [33]. He et al. found the spectra for multibit higher order $\Delta\Sigma$ modulators [56, 58]. As with the second-order case considered here, the quantizer must have sufficient bits to avoid overload (specifically, if the number of loops is k , then k bits are needed). They consider both dc and sinusoidal inputs. As with the two-stage and

second-order systems considered here, the M -stage (one bit per stage) and M th order (single k -bit quantizer) $\Delta\Sigma$ modulators yield the same quantization noise spectra.

2.10.3 Leaky Integrating $\Delta\Sigma$ Modulation

All of the systems considered here so far had a key aspect in common that permitted solution: The linear filtering within the loop consisted only of ideal discrete-time integrators; more complicated filters such as leaky integrators or integrators with nonunity gain were not considered. While results for general filtering do not exist, the special case of leaky integration and nonunity gain has been considered. Kieffer [64–66] has extended the single-stage $\Delta\Sigma$ result to more general systems with dc inputs that include DPCM and leaky integrating $\Delta\Sigma$ s, but his techniques differ from those considered here and have not been fully exploited for the $\Delta\Sigma$ application. Feely and Chua [51] used ideas from dynamical systems theory to describe various properties of a leaky $\Delta\Sigma$ modulator, including the input/output relation. The methods described here can be applied to the leaky integrator and the integrator with nonunit gain [67]. The analysis is complicated, but the results can be easily summarized.

For the special case of a dc input, the following properties hold:

- The quantizer sequence and error sequence are periodic, even for irrational dc inputs (unlike the ideal integrator case).
- The error sequence is no longer uniform and the mean is not the input dc. This means traditional decoders (low-pass filters) give biased reproduction (unlike the ideal integrator case). In fact, it can be shown that for a dc input, the input/output relation resulting when a long comb filter is used for digital-to-analog conversion is a form of Cantor function or a “devil’s staircase,” and it is a complicated function to describe analytically. Even when an arbitrarily large number of bits are used to reconstruct the input, the output is biased.
- The quantizer errors are not white.
- The SQNR is reduced from the ideal integrator case.

2.10.4 Multibit Quantizer, Single-Bit Feedback

The methods described here do not work for all popular $\Delta\Sigma$ architectures, but they do work for a variety of systems. One such example of interest is the system introduced by Leslie and Singh [68, 69]. Although the theoretical treatment of $\Delta\Sigma$ modulation given here permits multibit quantization, such systems have the practical shortcoming of requiring extremely accurate digital-to-analog conversion in the feedback loop (a problem that vanishes in the binary quantizer case). Leslie and Singh proposed combining multibit in the forward loop with single bit in the feedback loop. Analysis shows that the performance of this system is identical to that of an ordinary multibit single stage $\Delta\Sigma$ having an additional bit in the quantizer (and feedback loop). Hence previous analysis applies [70].

2.10.5 Related Work

During recent years many efforts have been made to find and apply exact analysis methods to $\Delta\Sigma$ modulators for the purpose of describing their behavior, predicting their

performance, and developing improved systems. These works have in common with this chapter the goal of avoiding unjustified application of the white-noise approximation, but the detailed methods and applications are not constrained to those described here. Of particular relevance to the issues considered here are the work of Delchamps on the behavior of control systems containing quantizers inside of feedback loops [71, 72], the work of Galton et al. demonstrating the existence of stationary distributions for the error sequence for a general class of $\Delta\Sigma$ modulators with random inputs such as Gaussian processes [59, 73–75], the work of Kieffer [64–66, 76] and Kieffer and Dunham [77] on the stability and convergence of one-bit feedback quantizers, and the work of Thao and Vetterli [78, 79] and Hein and Zakhor [61, 80] on optimal nonlinear decoders for $\Delta\Sigma$ modulators.

2.11 CONCLUSION

It might be said that the theory has failed to keep up with the fast pace of practice, that the best $\Delta\Sigma$ modulators have been developed based on engineering insight and suspect approximations, and exact analysis has usually followed far behind, if at all. Nonetheless, this chapter argued for a proper appreciation of the common approximation techniques, their origins, and their limitations and to demonstrate several important examples where exact analysis is possible. Many open problems remain, including the evaluation of the second-order properties of the basic 1-bit second-order $\Delta\Sigma$ modulator as well as the general stability properties and the first- and second-order properties of the wide variety of hybrid cascade and higher order systems that have been proposed. Perhaps with time some of these difficult problems may yet yield to solution or interesting new systems may be found by modifying successful systems to make them more amenable to exact analysis.

ACKNOWLEDGMENTS

I would like to thank Ping-Wah Wong, Wu Chou, Sang Ju Park, and Rick VanderKam for their cooperation on much of the research reported herein. I would also like to thank A. H. Gray for first interesting me in quantization theory and pointing out the errors common in applications of Bennett's theory, Gabor Temes and Jim Candy for the pleasure of lecturing with them in several short courses on $\Delta\Sigma$ modulation, and István Kollár for helpful discussions on the differing properties of the approaches of Widrow, Sripad, and Snyder and of Bucklew and Gallagher. Much of the research leading to the results described here was funded by the National Science Foundation under Grant No. MIP-9014335 and by a Stanford University Center for Integrated Systems Seed Grant.

REFERENCES

- [1] W. R. Bennett, "Spectra of quantized signals," *Bell Syst. Tech. J.*, vol. 27, pp. 446–472, July 1948.
- [2] B. Widrow, "A study of rough amplitude quantization by means of Nyquist sampling theory," *IRE Trans. Circuit Theory*, vol. CT-3, pp. 266–276, 1956.

- [3] A. B. Sripad and D. L. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-25, pp. 442–448, Oct. 1977.
- [4] István Kollár, Associate Professor of Instrument and Measurement Engineering, Technical University of Budapest, Hungary, private communication, March 1994.
- [5] J. A. Bucklew and N. C. Gallagher, Jr., "Some properties of uniform step size quantizers," *IEEE Trans. Inform. Theory*, Vol. IT-26, pp. 610–613, 1980.
- [6] J. A. Bucklew and N. C. Gallagher, Jr., "A note on optimum quantization," *IEEE Trans. Inform. Theory*, Vol. IT-25, pp. 365–366, 1979.
- [7] J. A. Bucklew, "Two results on the asymptotic performance of quantizers," *IEEE Trans. Inform. Theory*, Vol. IT-30, pp. 341–348, 1984.
- [8] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston, 1992.
- [9] T. Mísawa, J. E. Iwersen, L. J. Loporcaro, and J. G. Rush, "A single-chip CODEC with filters utilizing $\Delta\Sigma$ modulation," *IEEE J. Solid State Circuits*, vol. SC-16, pp. 333–341, Aug. 1981.
- [10] J. C. Candy and O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-29, pp. 1316–1323, Sept. 1981.
- [11] M. Vidyasagar, *Nonlinear Systems Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1978.
- [12] A. Gelbe and W. E. V. Velde, *Multiple-Input Describing Functions and Nonlinear Systems Design*, McGraw-Hill, New York, 1968.
- [13] D. P. Atherton, *Stability of Nonlinear Systems*, Research Studies Press, Wiley, Chichester, 1981.
- [14] D. P. Atherton, *Nonlinear Control Engineering*, Van Nostrand Reinhold, New York, 1982.
- [15] A. R. Bergens and R. L. Franks, "Justification of the describing function method," *SIAM J. Control*, vol. 9, pp. 568–589, 1971.
- [16] R. C. Booton, Jr., "The analysis of nonlinear control systems with random inputs," in *Proceedings of the Symposium on Nonlinear Circuit Analysis*, Polytechnic Institute of Brooklyn, New York, April 1953.
- [17] R. C. Booton, Jr., "Nonlinear control systems with statistical inputs," Tech. Rep., Massachusetts Institute of Technology, Cambridge, MA, March 1952.
- [18] S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Trans. Circuits Systems*, vol. CAS-34, pp. 593–603, June 1987.
- [19] D. S. Arnstein, "Quantization error in predictive coders," *IEEE Trans. Commun.*, vol. COM-23, pp. 423–429, April 1975.
- [20] D. Slepian, "On delta modulation," *Bell Syst. Tech. J.*, vol. 51, pp. 2101–2136, 1972.
- [21] L. G. Roberts, "Picture coding using pseudo-random noise," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 145–154, Feb. 1962.
- [22] L. Schuchman, "Dither signals and their effects on quantization noise," *IEEE Trans. Commun. Technol.*, vol. COM-12, pp. 162–165, Dec. 1964.
- [23] J. Vanderkooy and S. P. Lipshitz, "Dither in digital audio," *J. Audio Eng. Soc.*, vol. 35, pp. 966–975, Dec. 1987.

- [24] R. M. Gray and T. J. Stockham, Jr., "Dithered quantizers," *IEEE Trans. Inform. Theory*, vol. 38, pp. 805–812, May 1993.
- [25] S. O. Rice, "Mathematical analysis of random noise," in N. Wax and N. Wax, eds., *Selected Papers on Noise and Stochastic Processes*, Dover, New York, 1954, pp. 133–294.
- [26] W. B. Davenport and W. L. Root, *An Introduction to the Theory of Random Signals and Noise*, McGraw-Hill, New York, 1958.
- [27] A. G. Clavier, P. F. Panter, and D. D. Grieg, "Distortion in a pulse count modulation system," *AIEE Trans.*, vol. 66, pp. 989–1005, 1947.
- [28] A. G. Clavier, P. F. Panter, and D. D. Grieg, "PCM distortion analysis," *Electric Eng.*, pp. 1110–1122, Nov. 1947.
- [29] B. Widrow, "Statistical analysis of amplitude quantized sampled data systems," *Trans. Am. Inst. Elec. Eng., Pt. II: Applications and Industry*, vol. 79, pp. 555–568, 1960.
- [30] J. E. Iwersen, "Calculated quantizing noise of single-integration delta-modulation coders," *Bell Syst. Tech. J.*, pp. 2359–2389, Sept. 1969.
- [31] L. Ljung, *System Identification*, Prentice-Hall, Englewood Cliffs, NJ, 1987.
- [32] W. Chou and R. M. Gray, "Dithering and its effects on sigma-delta and multistage sigma-delta modulation," *IEEE Trans. Inform. Theory*, vol. 37, pp. 500–513, May 1991.
- [33] H. Bohr, *Almost Periodic Functions*, Chelsea, New York, 1947.
- [34] K. Petersen, *Ergodic Theory*, Cambridge University Press, Cambridge, 1983.
- [35] T. A. C. M. Claasen and A. Jongepier, "Model for the power spectral density of quantization noise," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-29, pp. 914–917, Aug. 1981.
- [36] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
- [37] S. P. Lipshitz, R. A. Wannamaker, J. Vanderkooy, and J. N. Wright, "Non-subtractive dither," *IEEE Trans. Signal Proc.*, to appear.
- [38] C. C. Cutler, "Transmission systems employing quantization," U.S. Patent No. 2,927,962, 1960.
- [39] H. Inose and Y. Yasuda, "A unity bit coding method by negative feedback," *Proc. IEEE*, vol. 51, pp. 1524–1535, Nov. 1963.
- [40] J. C. Candy, "A use of limit cycle oscillations to obtain robust analog-to-digital converters," *IEEE Trans. Commun.*, vol. COM-22, pp. 298–305, March 1974.
- [41] J. C. Candy, "A use of double integration in sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, March 1985.
- [42] J. C. Candy, "Decimation for sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-34, pp. 72–76, Jan. 1986.
- [43] J. C. Candy, Y. C. Ching, and D. S. Alexander, "Using triangularly weighted interpolation to get 13-bit PCM from a sigma delta modulator," *IEEE Trans. Commun.*, pp. 1268–1275, Nov. 1976.
- [44] R. M. Gray, "Oversampled sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-35, pp. 481–489, April 1987.
- [45] R. M. Gray, "Quantization noise spectra," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 1220–1244, Nov. 1990.
- [46] V. Friedman, "Structure of the limit cycles in sigma delta modulation," *IEEE Trans. Commun.*, vol. 36, no. 8, pp. 972–979, Aug. 1988.
- [47] R. M. Gray, "Spectral analysis of quantization noise in single-loop sigma-delta modulation with dc inputs," *IEEE Trans. Commun.*, pp. 588–599, June 1989.
- [48] J. E. Iwersen, "Comments on 'The structure of the limit cycles in sigma delta modulation,'" *IEEE Trans. Commun.*, vol. 38, no. 8, p. 1117, Aug. 1990.
- [49] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*, Academic Press, New York, 1965.
- [50] R. M. Gray, W. Chou, and P. W. Wong, "Quantization noise in single-loop sigma-delta modulation with sinusoidal inputs," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 956–968, 1989.
- [51] O. Feely and L. Chua, "The effect of integrator leak in Σ - Δ modulation," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 1293–1305, Nov. 1991.
- [52] O. Feely and L. Chua, "Nonlinear dynamics of a class of analog-to-digital converters," *Int. J. Bifurc. Chaos*, vol. 2, pp. 325–340, June 1992.
- [53] R. Schreier, "Destabilizing limit cycles in delta-sigma modulators with chaos," *Proceedings of the 1993 International Symposium on Circuits and Systems*, Chicago, IL, May 1993, pp. 1369–1372.
- [54] P. W. Wong and R. M. Gray, "Two stage sigma-delta modulation," *IEEE Trans. Acoust. Speech Signal Proc.*, pp. 1937–1952, Nov. 1989.
- [55] N. He, A. Buzo, and F. Kuhlmann, "A frequency domain waveform speech compression system based on product vector quantizers," *Proc. IEEE ICASSP* Tokyo, Japan, vol. 4, pp. 3031–3034, April 1986.
- [56] N. He, A. Buzo, and F. Kuhlmann, "Multi-loop sigma-delta quantization: spectral analysis," *Proc. IEEE ICASSP* vol. 3, pp. 1870–1873, 1988.
- [57] N. He, F. Kuhlmann, and A. Buzo, "Double-loop sigma-delta modulation with dc input," *IEEE Trans. Commun.*, vol. COM-38, pp. 487–495, 1990.
- [58] N. He, F. Kuhlmann, and A. Buzo, "Multi-loop sigma-delta quantization with dc input," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1015–1028, 1993.
- [59] P. W. Wong and R. M. Gray, "Sigma-delta modulation with i.i.d. Gaussian inputs," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 784–778, July 1990.
- [60] H. Wang, "A geometric view of Σ - Δ modulation," *IEEE Trans. Circuits Syst.-II*, vol. 39, pp. 402–405, June 1992.
- [61] S. Hein and A. Zakhor, "Stability and scaling of double loop Σ - Δ modulators," in *Proceedings 1992 ISCAS*, San Diego, CA, pp. 1312–1315, IEEE, 1992.
- [62] S. C. Pinault and P. V. Lopresti, "On the behavior of the double loop sigma delta modulator," *IEEE Trans. Circuits Syst.*, vol. 40, pp. 467–479, Aug. 1993.
- [63] W. Chou, P. W. Wong, and R. M. Gray, "Multi-stage sigma-delta modulation," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 784–796, 1989.
- [64] J. C. Kieffer, "Sturmian minimal systems associated with the iterates of certain functions on an interval," in *Proceedings of the Special Year on Dynamical Systems*, Lecture Notes in Mathematics, Springer-Verlag, New York, 1988.
- [65] J. C. Kieffer, "Analysis of DC input response for a class of one-bit feedback encoders," *IEEE Trans. Commun.*, vol. COM-38, pp. 337–340, 1990.

- [66] J. C. Kieffer, "Note on 'Spectral analysis of quantization noise in a single-loop sigma-delta modulator with dc input,'" *IEEE Trans. Commun.*, vol. 38, pp. 337–340, March 1990.
- [67] S. J. Park and R. M. Gray, "Sigma-delta modulation with leaky integration and constant input," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1512–1533, Sept. 1992.
- [68] T. Leslie and B. Singh, "An improved sigma-delta modulator architecture," in *Proceedings 1990 IEEE International Symposium on Circuits and Systems*, vol. 1, New Orleans, LA, IEEE, May 1990, pp. 372–375.
- [69] T. Leslie and B. Singh, "Sigma-delta modulators with multibit quantising elements and single-bit feedback," *IEE Proceedings G (Circuits, Devices and Systems)*, vol. 139, pp. 356–362, June 1992.
- [70] S. J. Park and R. M. Gray, "Sigma-delta modulation with leaky integration and constant input," in *Abstracts of the 1991 IEEE International Symposium on Information Theory*, Budapest, Hungary, IEEE, June 1991, p. 119.
- [71] D. Delchamps, "Exact asymptotic statistics for sigma-delta quantization noise," in *Proceedings of the Twenty-Eighth Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, Oct. 1990, pp. 703–712.
- [72] D. Delchamps, "Quantizer dynamics and their effect on the performance of digital feedback control systems," in *Proceedings of the 1992 American Control Conference*, vol. 3, Chicago, IL, American Autom. Control Council, June 1992, pp. 2498–2503.
- [73] I. Galton, "Granular quantization noise in the first-order $\Delta\Sigma$ modulator," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1944–1956, Nov. 1993.
- [74] I. Galton, "Granular quantization noise in a class of delta-sigma modulators," *IEEE Trans. Inform. Theory*, vol. 40, pp. 848–859, May 1994.
- [75] T. Koski, "Statistics of the binary quantizer error in single-loop sigma-delta modulation with white Gaussian input," *IEEE Trans. Inform. Theory*, vol. 41, pp. 931–943, July 1995.
- [76] J. C. Kieffer, "Stochastic stability for feedback quantization schemes," *IEEE Trans. Inform. Theory*, vol. 28, pp. 248–254, March 1982.
- [77] J. C. Kieffer and J. G. Dunham, "On a type of stochastic stability for a class of encoding schemes," *IEEE Trans. Inform. Theory*, vol. 29, pp. 703–797, Nov. 1983.
- [78] N. Thao and M. Vetterli, "Optimal MSE signal reconstruction in oversampled A/D conversion using convexity," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, IEEE, March 1992.
- [79] N. Thao and M. Vetterli, "Oversampled A/D conversion using alternate projections," in *Proceedings of the Twenty-fifth Annual Conference on Information Sciences and Systems*, John Hopkins, Baltimore, MD, March 1991, pp. 241–248.
- [80] S. Hein and A. Zakhor, *Sigma Delta Modulators: Nonlinear Decoding Algorithms and Stability Analysis*, Kluwer Academic Publishers, Boston, 1993.

Quantization Errors and Dithering in $\Delta\Sigma$ Modulators

3.1 INTRODUCTION

The material presented in Chapter 2 lays a theoretical foundation for understanding the nature of quantization errors in $\Delta\Sigma$ modulation. An exact analytical solution exists only for a few simple $\Delta\Sigma$ modulator architectures. Many practical and commercially useful architectures do not yet have an exact analytical solution. It is therefore necessary to describe the behavior of these systems empirically. The purpose of this chapter is twofold: first, to present descriptions of the quantization errors of these architectures, which are derived from exhaustive empirical studies; then, to describe techniques for whitening these quantization errors through *noise-shaped dithering*. The methods and techniques presented in this chapter will cover issues pertaining to both A/D and D/A converters and their implementations based on $\Delta\Sigma$ modulation. However, the material is certainly not limited just to data converter applications. It also applies to digital requantization and more generally to any signal processing procedure that uses quantization, noise shaping, and feedback.

The quantization error from $\Delta\Sigma$ modulation is typically not white. For dc inputs, the quantization error is periodic, generating what is commonly referred to as *idle channel tones* or *pattern noise*. Its spectrum contains discrete tones whose frequencies and amplitudes are a function of the input level. For ac inputs, the quantization error is also periodic, containing components harmonically related to the input frequency and amplitude. The quantization error, more commonly referred to as quantization noise,¹ is manifested at the

1. While the term *quantization noise* is commonly used, it is somewhat imprecise and misleading because it is not necessarily a noiselike entity: In an all-digital implementation it is purely deterministic, while in a mixed analog/digital implementation it is at least partly deterministic. *Quantization error* is a more descriptive and precise term, and it specifically refers to *the output of the quantizer minus its input*. Nevertheless, the two terms will be used interchangeably in this book.

output of the $\Delta\Sigma$ modulator through the inherent high-pass noise transfer function of the modulator. This high-pass filter function is designed to suppress the baseband components of the quantization error, such that the design objectives for dynamic range and signal-to-noise ratio (SNR) can be met.

The periodic structure of the quantization error can cause problems when $\Delta\Sigma$ modulators are used in data converter applications, especially when the human ear is the end receiver. The ear is very sensitive to certain types of coloration and periodicities in sound. Many have tried to characterize the nature of the quantization error in the converter by simply observing the output spectrum. Simple spectral analysis *alone* is often insufficient and can even be misleading. It will be shown in Section 3.3 that important information about the true nature of the quantization error can be lost in an actual power spectral density (PSD). Short-term estimates of the autocorrelation or PSD are often more useful for analyzing the quantization error. The low-pass filtered output from a $\Delta\Sigma$ modulator can exhibit a periodic characteristic resembling an impulse train or sawtooth wave, having a peak-to-rms ratio as high as 20–30 dB, which helps explain why it can have a perceptually annoying quality. One can view this effect as a *time-domain distortion* and therefore argue that the converter actually has less resolution than what is typically claimed based just on rms measurements.

There are also other issues. Perhaps an even more serious problem occurs due to the fact that nearly all types of $\Delta\Sigma$ modulators can produce very high-powered tones near $f_s/2$. Even the slightest amount of clock noise near this frequency in physical proximity to the modulator can couple and demodulate these tones down into the baseband. Yet another type of problem occurs for ac inputs: Strong peaks and dips in the output noise power may be seen for certain input frequencies and amplitudes.

Until recently, many researchers and designers concluded that either higher order single-stage modulators or multistage modulators were the solution to making the quantization error more random and alleviating these problems. Unfortunately, some of these early conclusions were based on either incomplete empirical evidence or else required conditions in the implementation that were physically unrealizable.

A technique of directly dithering the quantizer of the $\Delta\Sigma$ modulator with a relatively large-magnitude pseudorandom dither signal will be described in Sections 3.8 and 3.9. It will be shown that such *noise-shaped dithering* eliminates the unwanted tones and whitens the noise floor of the $\Delta\Sigma$ modulator. In addition, this type of dithering may actually improve the resolution of the converter for lower level ac inputs. This technique has been successfully implemented in several commercial $\Delta\Sigma$ converter chips, and it can be applied to all types of $\Delta\Sigma$ modulators.

While such dithering is very practical and useful, it is not without its faults. It will diminish the usable dynamic range by several decibels for large-magnitude inputs near full scale. For higher order single-stage modulators, it will also cause the threshold of instability to occur a few decibels sooner. These problems are ameliorated with *dynamic dithering*. This entails modulating the dither with a nonlinear function of the input signal in a manner that produces no additional distortion and is perceptually inaudible. This enables virtually all of the potential dynamic range from the modulator to be realized while still whitening the quantization noise. Dynamic dithering will be discussed in Section 3.13.

The chapter also includes discussions on alternative methods to dithering, including the *chaos* technique.

3.1.1 Problems with Empirically Based Reports on $\Delta\Sigma$ Modulators

The literature on this subject—quantization errors, idle channel tones, and dithering in $\Delta\Sigma$ modulators—has produced considerably inconsistent results. For example, one report may suggest that a certain type of architecture is free of tones, while another publication shows the presence of tones in the same architecture. Other examples entail specific techniques for elimination of tones, wherein a specific technique is shown effective in one report while other evidence suggests that it is less effective.

There are two fundamental reasons that these types of inconsistencies exist. First of all, this subject lacks rigorous analytical solutions that apply to all useful modulator architectures and to nonidealities commonly encountered in the implementation of these architectures. To overcome this limitation, various simplifications in the analysis have been proposed that require some basic underlying assumptions. These assumptions may initially appear valid when checked with simulations but may be invalid in actual implementation because one or more dimensions to the problem were missing in the analysis. Another reason for inconsistency on this subject is due to the fundamental nature of empiricism. While empirical methods will typically vary among individual researchers, the biggest danger is that the results may not be complete enough to produce the type of general conclusions needed and useful for practice. Hence, there are a number of reports in the open literature that contain results valid only for a very narrow or specific set of conditions. Too often, this set of conditions is not clearly stated or qualified. More seriously, some empirical results are altogether erroneous due to a technical flaw in the simulation or testing method that may not have been discovered until much later.

3.1.2 Steps Taken to Ensure Accuracy of Results

Since this chapter relies largely on empirical support for its claims, the reader should be aware that the following precautionary steps were taken to ensure that the results were accurate:

1. Data were taken from rigorous simulation studies of various modulator architectures.
2. The actual behavior of commercially available $\Delta\Sigma$ conversion chips of various architectures was carefully analyzed, and measurement data from these chips is reported.
3. The author was involved in the design and implementation of $\Delta\Sigma$ conversion chips employing many of the techniques described in this chapter, and measurement data from these chips is also reported.
4. Data from simulations with actual measurement data were carefully correlated, and those differences are clearly articulated.

Care has been exercised throughout the chapter not to extend any claims beyond those that have been rigorously verified, nor to dismiss out-of-hand any alternative viewpoints just because they do not agree with the main points emphasized in the chapter.

The $\Delta\Sigma$ modulator is often analyzed by making a simple assumption that the quantization error is an additive, uniformly distributed noise independent of the signal. In this model, the nonlinear quantizer is replaced by a unity-gain summing element where the uniform noise is added. It was seen in Chapter 2 that this assumption has many limitations, but it is useful in at least two respects. First, it provides a general approximation of the noise-shaping properties of the system. Second, the white-noise model results in a near-exact calculation of the mean-square quantization error for a large variety of inputs and nonoverloading systems [1, 2]. In no way does the model predict the fundamental problem of idle channel tones and pattern noise [3], nor does it account for excess noise due to overload. This model was first used for PCM quantizers, but has limitations even in this context [3–5]. The application of the white-noise model to $\Delta\Sigma$ modulation predicts that the noise at the output of the modulator will also be uncorrelated from the input. Its transfer functions can be found from the block diagram as

1. The loop filters $G(z)$ and $F(z)$ are assumed to be sampled data filters, but they could also be continuous time filters without any loss of generality.

Delta-sigma modulator designs reduced to practice fall into two categories: single-stage and multistage modulators. The second-order single-stage modulator described by Candy and his colleagues [8, 16, 17] has a noise transfer function $H(z)$ of $(1 - z^{-1})^2$. Higher order single-stage modulators typically employ a noise-shaping filter having both

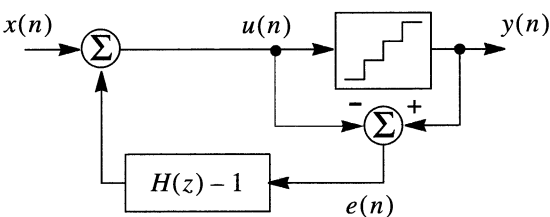


Figure 3.2 General noise-shaping coder.

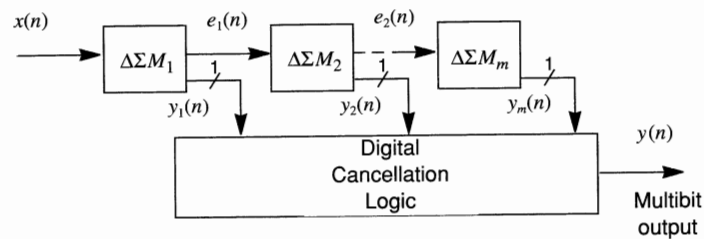


Figure 3.3 Multistage (MASH) $\Delta\Sigma$ modulator architecture.

poles and zeros. Such topologies are described in [18–23]. Multistage modulators are an attractive alternative [24–26]. They do not suffer from potential instabilities, as do the higher order single-stage modulators. They are composed of cascaded first- or second-order modulators. If ideally implemented, their noise transfer function is simply $(1 - z^{-1})^m$, where m is the overall order.

In a multistage converter architecture, two or more $\Delta\Sigma$ modulators are placed in cascaded stages with one another. The quantization noise from one stage is fed into the input of the following stage. This is shown in Figure 3.3. The quantization noise terms from all but the final modulator are subtracted by digital cancellation logic. This digital subtraction process produces a multibit output, which then must be low-pass filtered. This complicates the design and makes it more susceptible to circuit nonidealities. In the case of the multistage A/D converter, the analog parameters of the noise-shaping modulator must be predictable for the subtraction logic to work. For the multistage D/A architecture, some sort of multibit D/A converter needs to be part of the scheme, and it must precede the analog low-pass filter. This multibit D/A is commonly implemented by a high-speed pulse-width modulator that oversamples the already oversampled digital signal and turns the multibit samples into a 1-bit stream [24].

3.3 OBSERVABILITY OF PERIODIC SEQUENCES

Many skilled in the art of $\Delta\Sigma$ converter design have found idle channel tones and pattern noise behavior a problem that is difficult to observe and characterize [27]. Designers and end users claim to hear the phenomenon, but find that the tones are very far down or even absent when looking at the power spectrum, whether looking at results from a simulation or actually characterizing the behavior of a hardware implementation. There are several reasons for this.

The first concept to be addressed is the following: *strong periodicity can be readily apparent in the time domain, but not in the frequency domain*. This concept may at first seem counterintuitive. However, since the *reverse* is classically well known, that is, that the frequency domain is able to reveal periodicity when the time domain does not, then it seems plausible that a class of signals may exist where the concept applies in reverse of the classic notion. Consider a sine wave sequence and multiply it in the time domain with a random white-noise sequence.¹ The result is another random white-noise sequence [29,

30]. Multiplication in the time domain (or sample domain) is equivalent to convolution in the frequency domain. Since a sine wave is an impulse (delta function) in the frequency domain, and since a random white-noise sequence has a flat frequency spectrum¹ (i.e., a constant with respect to frequency), then the convolution of the two will also produce a flat spectrum. Now consider many sine waves of various frequencies, sum them together, and multiply by a random white-noise sequence. The result is still another random white-noise sequence. In general, *a random white noise multiplied in the time domain with any number of sinusoids results in yet another random white-noise sequence*.

The problem becomes more interesting when the frequencies are harmonically related, having zero phase with respect to one another. In this case, their sum is given by

$$\tilde{x}(n) = \sum_{k=1}^K \cos(k\omega_0 n) \quad (3.3)$$

Multiplication of this sum with a random white-noise sequence $s(n)$ will result in the frequency-domain convolution given by

$$S(e^{j\omega}) \otimes \tilde{X}(e^{j\omega}) = S(e^{j\omega}) \otimes \sum_{k=1}^K \frac{1}{2} \delta(\omega \pm k\omega_0) \quad (3.4)$$

The envelope of the waveform is simply a train of sinc pulses and the carrier is the noise. As the number of cosinoids increases, the sinc pulses become narrower and taller. The limit, as $K \rightarrow \infty$, will result in a train of impulses in the time domain. Consider an example where $K = 32$, that is, the frequencies are at

$$\omega_0, 2\omega_0, 3\omega_0, \dots, 32\omega_0 \quad (3.5)$$

Also, consider that the noise $s(n)$ has a rectangular probability density function (rpdf) bounded between ± 0.5 . Figure 3.4 shows the time-domain output from the multiplication of the noise with the cosinoidal sum. Initially, it appears from Figure 3.4 that the sequence is periodic with a fundamental period of 512 samples. More careful analysis reveals, however, that the peaks from Figure 3.4 do not always fall precisely at multiples of 512—sometimes they are off by ± 1 sample from their expected sample number. However, one can also readily see that the peak values themselves are random in amplitude, bounded between $\pm K/2$, and that the resulting pdf is no longer uniformly distributed between ± 0.5 . However, the calculated pdf of the output from simulation reveals that only about 5% of the samples exceed the bounds of ± 0.5 .

The peak-to-rms power ratio for the cosinoidally modulated noise of Eq. (3.4) can be readily derived. The power of a single sine wave is $\frac{1}{2}$ when its peak-to-peak amplitude is ± 1 . Therefore, K such sine waves will produce a power of $K/2$. The mean-square value of the noise is $\frac{1}{12}$ if it has a rpdf bounded between ± 0.5 [31]. Therefore,

$$\text{Peak-to-rms power ratio} = 10 \log_{10}(6K) \quad (3.6)$$

1. Any zero-mean process with a constant power spectrum is said to be a *white random process* [28].

1. It is assumed that the random white-noise signal is at least *wide-sense stationary*, and its samples are independent and identically distributed (i.i.d.) zero-mean random variables [28].

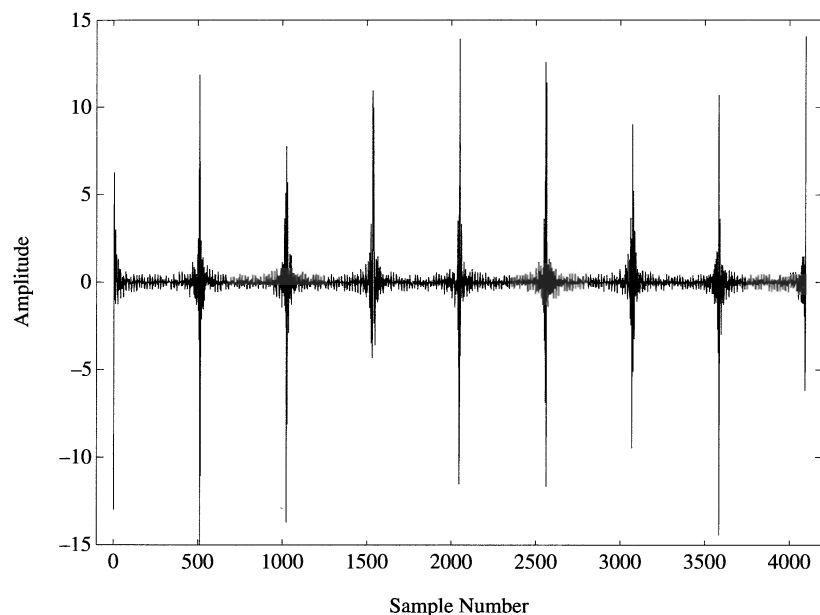


Figure 3.4 Harmonically related cosinoids ($K = 32$) modulated by pseudo-random noise.

For the example of $K = 32$, the calculated power ratio from Eq. (3.6) is 22.8 dB, and the measured value from the simulation was 22.5 dB, showing good agreement. In actual $\Delta\Sigma$ modulators, such high peak-to-rms power ratios of pattern noise sequences are typical, which will be shown in Section 3.4. Herein lies one of the reasons why this noise may be audibly perceived.¹ Careful perceptual listening tests by the author reveal that the example shown in Figure 3.4 will sound periodic if the fundamental frequency ω_0 is in the low- to mid-frequency audio range. The sound has an airy *buzzing* quality. If ω_0 is a relatively high frequency audio signal (>5 kHz), the sequence will tend to sound white.

In commercial practice, the power spectrum is often estimated by simply computing the discrete Fourier transform of a given sequence. This can sometimes produce misleading results. For example, Figure 3.5 illustrates a windowed 32K-point FFT of the sequence from Figure 3.4. It does not reveal any distinguishing tones at multiples of ω_0 . It is well known, however, that more elaborate methods are often needed for estimating the true power spectrum [31], particularly if the signal is aperiodic or random. Classic spectral estimation techniques fall into two categories: the *modified periodogram* method (based on overlapping windowed spectral averages) and the autocorrelation method. The autocorrelation² describes the time variation of a sequence [31]. For a finite-length real-valued

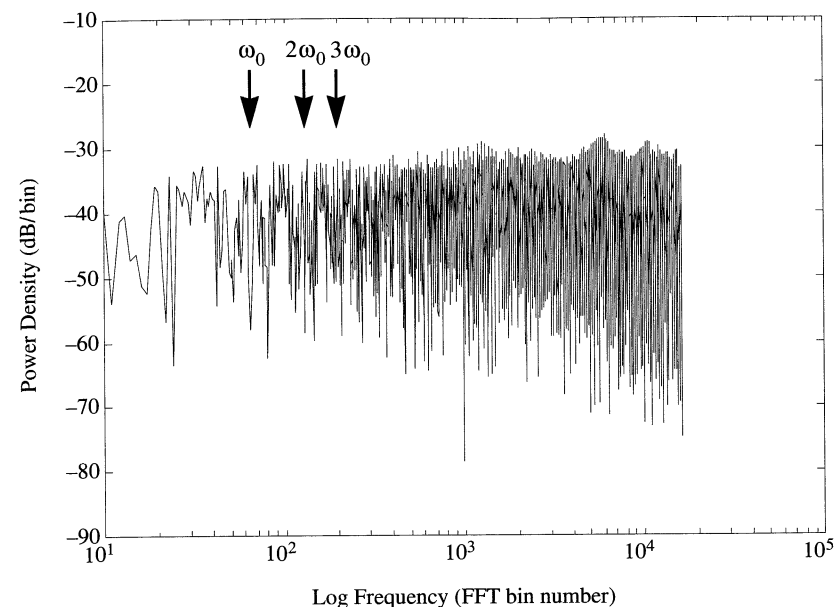


Figure 3.5 Power spectrum estimate of Figure 3.4 based simply on one 32K windowed FFT, revealing no outstanding periodicity at multiples of ω_0 .

sequence, the discrete-time autocorrelation *estimate* is given by

$$\phi_{xx}(m) = \frac{1}{2N} \sum_{n=-N}^{N-1} x(n)x(n+m) \quad (3.7)$$

The first value of the autocorrelation sequence $\phi_{xx}(0)$ is the *mean-square value* of the original sequence, or its *average power*. The remaining samples of the autocorrelation sequence $\phi_{xx}(m)$ for $m \neq 0$ reveal the interdependence of the signal with respect to itself for given shifts m in the sequence. If the original sequence is aperiodic, then these remaining autocorrelation samples will tend toward a constant. If the original sequence is perfectly white random noise with zero mean, then $\phi_{xx}(m) \rightarrow 0$ for $m \neq 0$.

The autocorrelation of the sequence of Figure 3.4 is shown in Figure 3.6, wherein Figure 3.6(a) is computed from 32K samples and Figure 3.6(b) is computed from 128K samples. It reveals a periodicity every 512 sample shifts, which is the period of the fundamental frequency ω_0 . The reason the autocorrelation appears periodic in this example is that the original noise sequence $s(n)$ has a finite length. The classical assumptions regarding random white noise require that the noise be *wide-sense stationary*, a condition that cannot be guaranteed with any finite-length sequence. This explains why Figure 3.6(b) shows smaller powered components at multiples of 512 sample shifts, since it was computed from a much longer sequence than that shown in Figure 3.6(a). Further simulations revealed that these periodic components tend to vanish as $N \rightarrow \infty$. It is important to note,

1. The peak-to-rms ratio of uniform white noise is only 4.77 dB.

2. The autocorrelation is often computed using Rader's algorithm [32], which is used exclusively for all autocorrelation simulation results reported in this chapter.

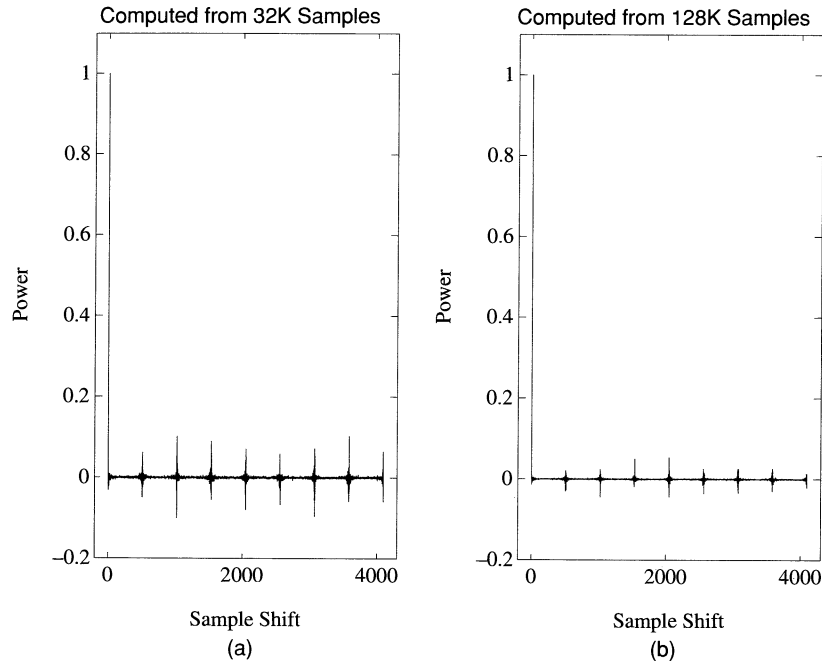


Figure 3.6 Autocorrelation of noise modulation of cosinoidal sequence from Figure 3.5, where (a) is computed from 32K samples and (b) is computed from 128K samples.

however, that if human perception is used as the test, then the wide-sense stationary principle no longer applies to the problem because the ear is able to detect relatively short-term periodicities with ease.

Now to illustrate the opposite sort of problem, suppose a pseudorandom noise sequence of finite-length N samples represents one period of a periodic sequence, that is, the N -length sequence were to continually repeat. If N is relatively long, it will be nearly impossible for an observer to see what is occurring in the time domain. However, Figure 3.7 shows an autocorrelation of such a sequence, where $N = 1024$. It can readily be seen that the mean-square value of the sequence is exactly repeated for every 1024-sample shifts; that is,

$$\phi_{xx}(0) = \phi_{xx}(1024) = \dots = \phi_{xx}(1024m) \quad m = 0, 1, 2, 3, \dots \quad (3.8)$$

This type of phenomenon also occurs in $\Delta\Sigma$ modulators, which will be shown in Section 3.4. In lower order modulators, the pattern noise is usually simpler and the correlation is easier to observe. In higher order loops, the structure of the pattern noise can be very long and complicated, but still it is correlated with itself over a finite length.

3.4 TONES IN SINGLE-STAGE $\Delta\Sigma$ MODULATORS

In this section, tonal behavior in single-stage $\Sigma\Delta$ modulators will be examined. Examples will be shown for second-, third-, and fifth-order structures.

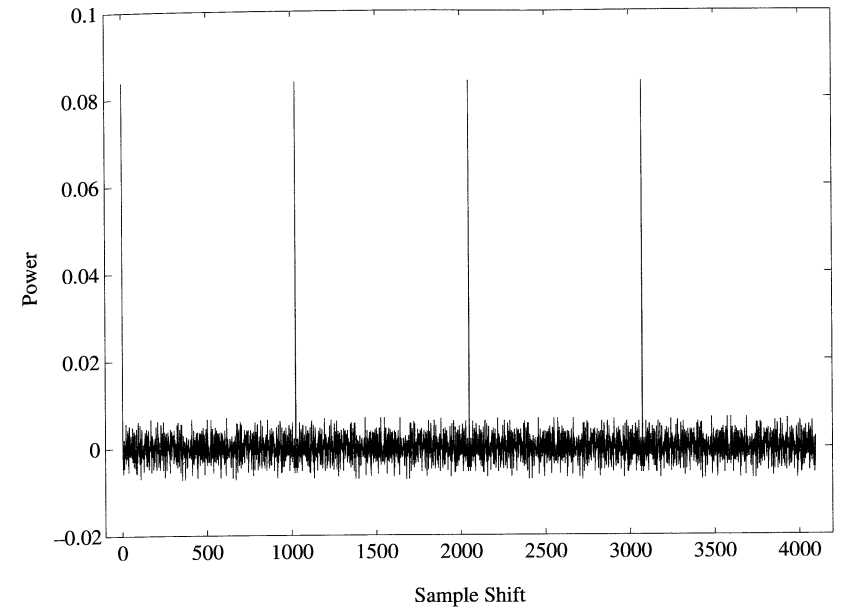


Figure 3.7 Autocorrelation of repeated pseudorandom noise sequence for $N = 1024$.

For many years, it was generally well understood that first-order modulators produced tones [33, 34]. Early proponents of higher order single-stage modulators reported that they were not tonal [35]. One reference [36] suggests that an analog second-order modulator will not sound tonal because the small thermal noises are enough to randomize the quantization noise. The use of small input dither, that is, one least significant bit (LSB) with respect to the input of the modulator, has been proposed in [37]. In this reference, it was shown that the PSD approaches white characteristics; however, this result is valid as the number of samples used in the PSD estimate approaches infinity. Pseudotones can be seen if the number of samples is relatively small [38].

As more practical experience was accumulated, however, tonal behavior was found even in fourth- and fifth-order single-stage analog modulators [27]. Since all analog modulators have small thermal noise sources, this leads to the conclusion that relatively small dither will be insufficient to remove short-term periodicity in the modulator, especially those that employ single-bit quantizers. However, a small input dither is much more effective if the modulator is multibit or multistage [2], but this is not necessarily the *optimum* dither, since it is not noise shaped (which will be discussed later in Section 3.8).

3.4.1 Second-Order Modulator

A second-order $\Delta\Sigma$ modulator having a 1-bit quantizer is first considered. The reference levels from the quantizer (which are fed back into the loop) are $\{-1, +1\}$, therefore, the quantization step size is $\Delta = 2$. A dc level of $\frac{1}{256}$ is applied to the modulator.

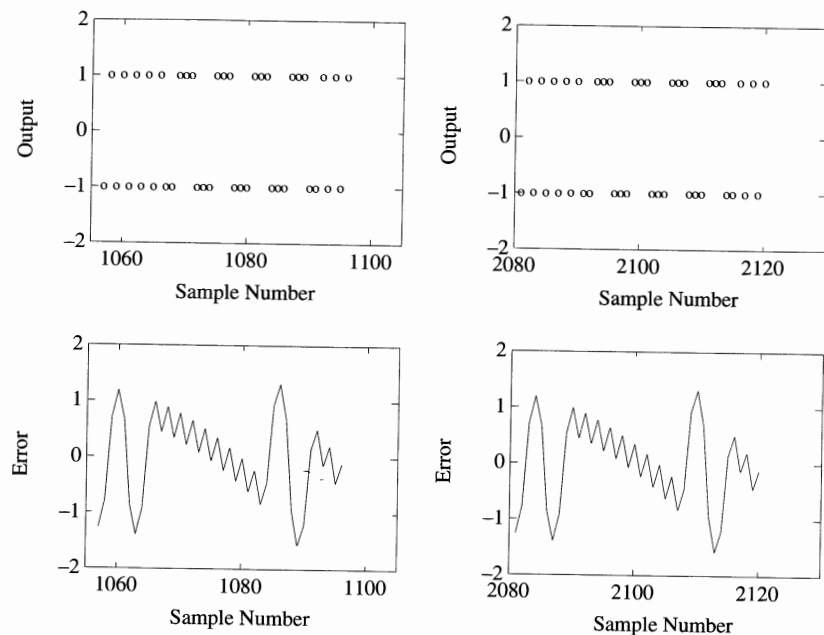


Figure 3.8 Output sequence $y(n)$ and quantization error $e(n)$ of second-order $\Delta\Sigma$ modulator for dc input.

Figure 3.8 shows the output sequence $y(n)$ and the quantization error $e(n)$. Note that these sequences are repeating exactly every 1024 samples. The output of the modulator is typically oscillating between ± 1 , except every 1024 cycles, when the pattern undergoes several repetitions of $\{1, 1, 1, -1, -1, -1\}$. The autocorrelation of the quantization error is shown in Figure 3.9. The mean-square value repeats every 1024 samples. The power spectrum of the modulator output $y(n)$ is shown in Figure 3.10. (The frequency axis is shown in terms of FFT bins.) This spectrum contains energy only at discrete multiples of the lowest frequency, that is, $f_0 = nf_s/1024$. Based on empirical observation for a given dc input level, the frequencies will be found at

$$f_0 = \frac{nf_s |A_{dc}|}{2\Delta} \quad n = \{0, 1, 2, \dots\} \quad (3.9)$$

where $|A_{dc}|$ is the magnitude of the dc input level.

Besides the dc input level, there are also other parameters that affect the tonal behavior. These include the initial conditions of the loop filter. Analog modulators (for A/D converters) and digital modulators (for D/A converters) can be algorithmically equivalent, but the analog modulator behaves somewhat differently because of nonideal effects, like small amounts of noise in the analog signal path, finite-gain amplifiers in the loop filter, and metal-oxide-semiconductor field-effect transistor (MOSFET) switch charge injection. Nevertheless, both the analog and digital modulators will generally exhibit tonal behavior as a function of dc input level.

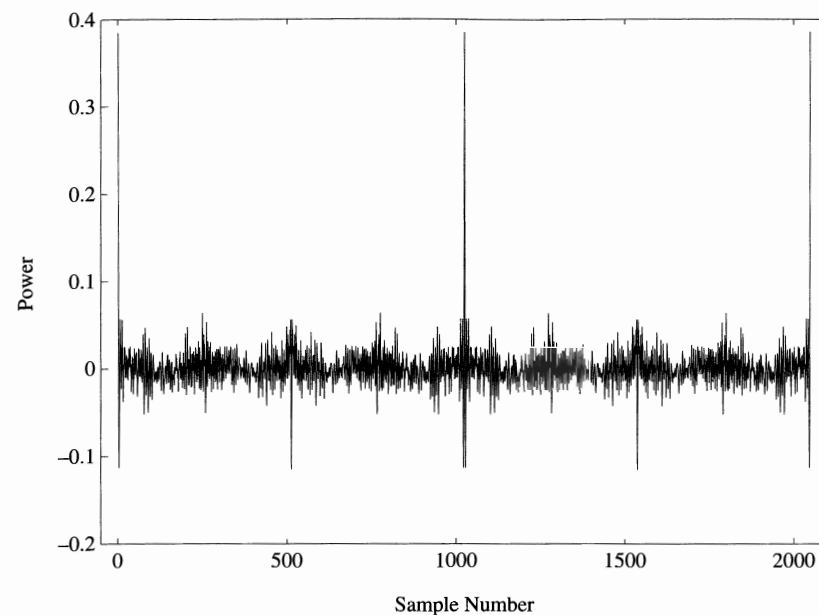


Figure 3.9 Autocorrelation of quantization error from Figure 3.8.

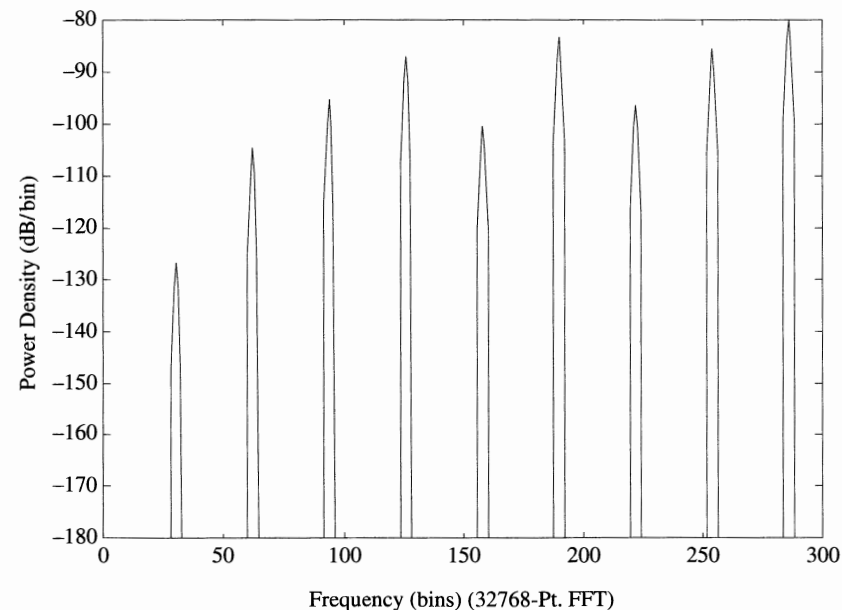


Figure 3.10 Low-frequency portion of spectrum from second-order modulator with dc input.

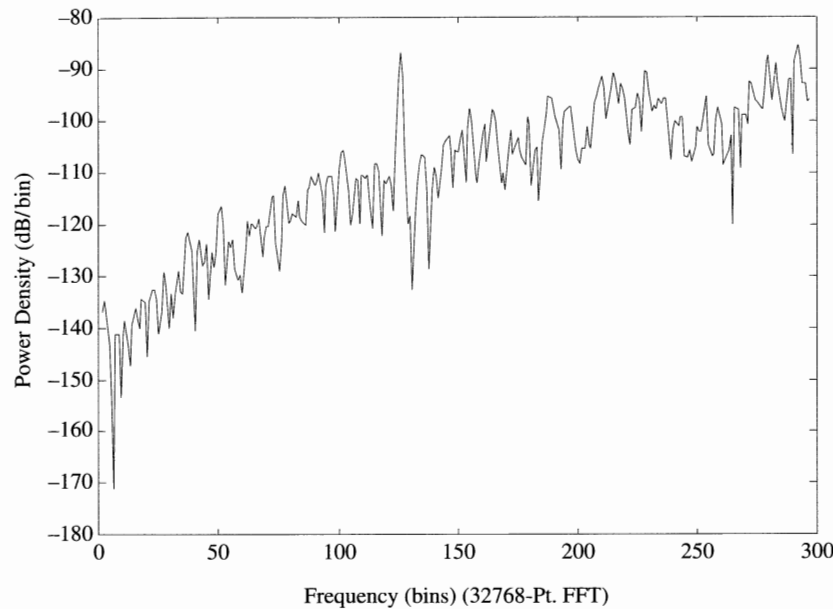


Figure 3.11 Same as Figure 3.10, but with 1-LSB dither added to the dc input of the modulator. The modulator is still tonal.

An experiment was conducted wherein a perceptual listening test and spectral analysis both revealed that an analog second-order modulator could easily produce audible tones in the baseband for given dc input levels. This result was verified through simulation, wherein a small dither was added to the input of the modulator along with the dc input in order to mimic the input-referred thermal noise of the analog components within the modulator. This was accomplished simply by a pseudorandom dither that spanned one LSB out of a 16-bit input range. The power spectrum from the simulation is seen in Figure 3.11. While many of the smaller baseband tones are smoothed away, at least one dominant tone at $n = 4$ remains high above the noise floor, while other baseband tones are buried beneath the noise floor. This one dominant tone is actually a little higher in amplitude than in the undithered case.

3.4.2 Third-Order Modulator

The next example illustrates the behavior of a third-order single-stage modulator having a single-bit quantizer. The practical application of this type of modulator has been previously reported [39–41]. This architecture can be realized either as that of Figure 3.1 or 3.2. In these simulations, an objective was to keep the rms quantization noise power below -100 dB. Therefore, an oversampling ratio of 192 was selected for the third-order modulator.

Unlike first- and second-order modulators, which typically have their noise transfer function zeros located at $z = 1$ (i.e., 0 Hz), higher order (>2) single-stage modulators may

have some zeros placed in conjugate pairs on the unit circle at $\omega > 0$ in the signal passband (the stopband of the noise-shaping filter). This helps reduce the quantization noise in the passband. Most commercial high-order modulators are designed in this manner [18–21]. For a third-order modulator, assuming a 20-kHz passband, this would entail placing one zero at 0 Hz while placing the other two as a conjugate pair near 17 kHz. An inverse Chebyshev high-pass filter design can be used to determine the near-optimal location of these zeros [42]. It also will generate a fair approximation of where the poles should be located. A simple design procedure for obtaining the noise-shaping filter transfer function using the MATLAB Signal Processing Toolbox [43] entails the following:

```
[B,A] = cheby2(3,80,1/192,'high'); %(Generate 3rd-order Chebyshev-II filter for 192-times OSR)
Bn = B/B(1); %(Normalize numerator polynomial)
```

This produces noise transfer function poles at

$$z_{1,2} = 0.87929419344178 \pm j0.1714566$$

$$z_3 = 0.80036125273101$$

while the zeros are located at

$$z_{1,2} = 0.99989960174966 \pm j0.01416991$$

$$z_3 = 1$$

Alternatively, all three zeros could be located at $z = 1$.

One technique has been reported to optimize the location of the poles [44], while another technique reports optimizing the location of both the poles and the zeros [45]. Another report suggests that idle tone behavior is suppressed better if all the zeros are placed $z_{1,2} = 1$ [40]. Therefore, the idle tone behavior for two cases will be examined: first, where the zeros are at $z_3 = 1$; then, where the zeros comply with the inverse Chebyshev placement. In both cases, the poles will be kept identical, their location being determined by the inverse Chebyshev function.

Figure 3.12 shows the audio baseband spectrum of the modulator for the case where all three zeros are located at $z = 1$. The baseband is 20 kHz, while the Nyquist rate is 44.1 kHz. The dc input level used in the simulation was $1/4096$, where the loop feedback levels were ± 1 . The spectrum is composed purely of discrete tones, clustered in groups at multiples of 2 kHz. The output of the modulator was then low-pass filtered to attenuate the high-frequency quantization noise above 20 kHz. Figure 3.13 shows the time-domain output after the low-pass filter. The impulse train effect is clearly evident. The impulses are spaced 0.5 ms apart, which is the period of the 2-kHz fundamental. While the 2-kHz tone has an rms power level of only -155 dB, it is the superposition of so many harmonically related frequencies that causes such a high peak amplitude in the time domain. The most significant contributions to the peak amplitude are the higher frequency harmonics near 20 kHz, whose rms power levels are about -115 dB. Simulations show that inadequate low-pass filtering attenuation above 20 kHz will cause the peak values of the impulses to be even higher. Therefore, a high-order reconstruction filter having a very narrow transition band is really needed.

The peak-to-peak impulse heights span at least 4 units of amplitude out of 65,536 units. In other words, given a 16-bit word size, the lower 3 bits are spanned by the impulse

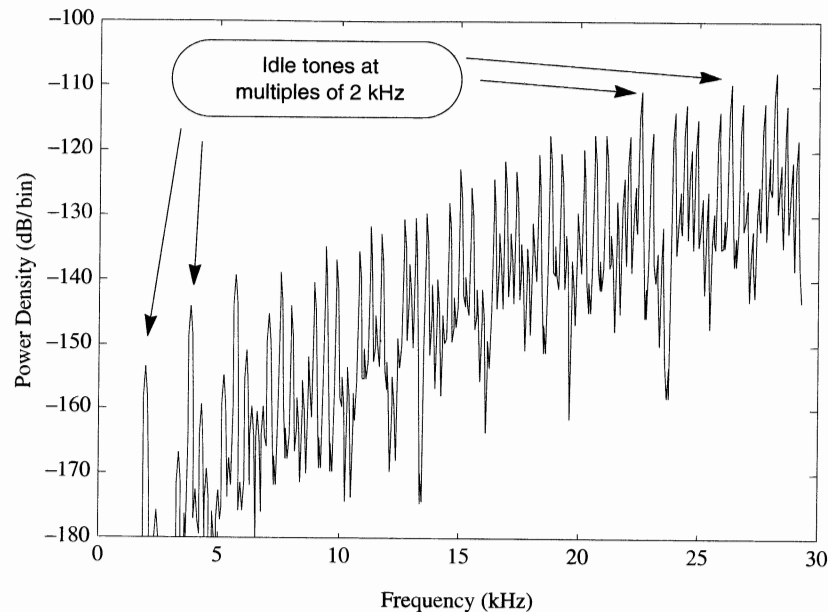


Figure 3.12 Baseband spectrum of third-order single-stage modulator for dc input. All three zeros at 0 Hz.

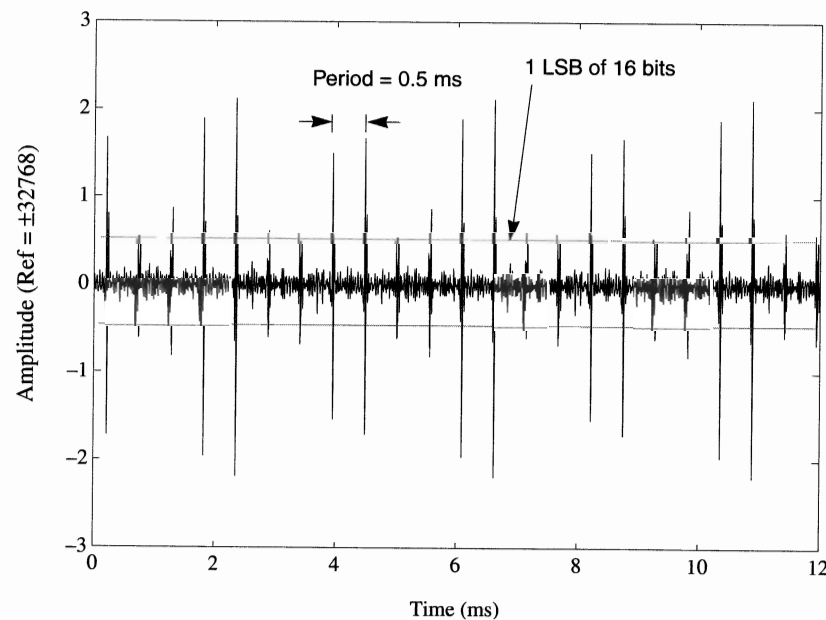


Figure 3.13 Low-pass filtered time output of third-order single-stage modulator for dc input. All three zeros at 0 Hz.

peaks. The rms power in the baseband, however, is about -100 dB. This indicates the peak-to-rms ratio is about 20 dB. The problem with the impulse train is that its pattern is so regular and the impulses so sharp that it could perceptually mask the sound of any real signal that occupies these lower bits. The information contained in these lower bits is not very useful while this impulse train is going on. *This time-domain distortion could cause an otherwise 16-bit converter to perform more like a 13-bit converter!*

Next, the third-order modulator was resimulated for the case where a complex zero is shifted up from 0 Hz to 17 kHz, as previously described. Figure 3.14 shows the audio baseband spectrum. The lower frequency tones are higher than before, but the tones in the vicinity of 20 kHz are lower because they are pulled down by the zeros at 17 kHz. However, the time-domain output from the low-pass filter as seen in Figure 3.15 indicates the same general behavior. There is actually a small improvement: The impulse heights are a little bit lower. In addition, the measured rms noise in the baseband is considerably better: about 8 dB lower than the previous case.

As seen from these simulations, the tones in the baseband spectrum are typically far down in magnitude individually, but the superposition of a great number of them taken in zero-phase harmonic relationship results in relatively large peak amplitudes in the time domain. Therefore, it is generally preferable to minimize the total baseband noise, especially near the band edge where higher frequency tone magnitudes are the greatest. Since these higher frequency tones are larger, they are contributing the most to the total height of the lower frequency impulses seen in the time domain.

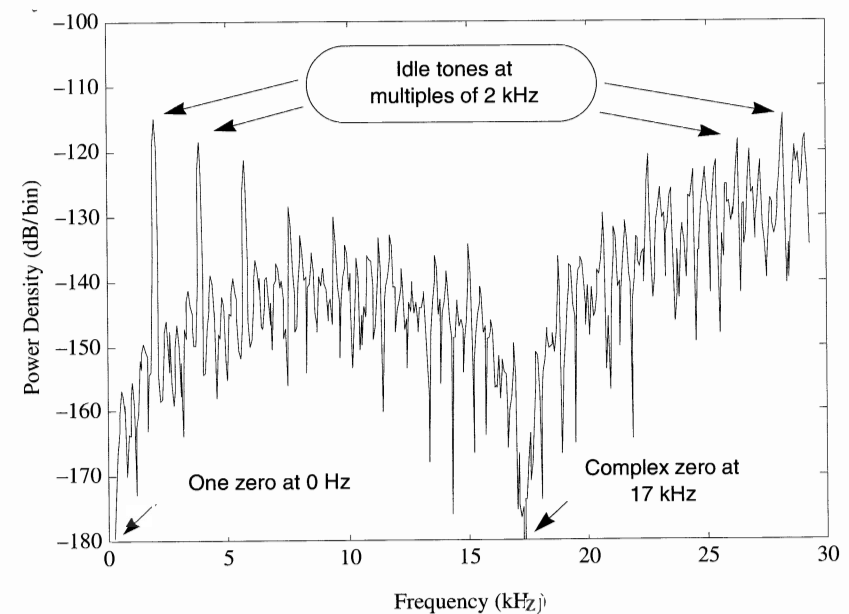


Figure 3.14 Similar to Figure 3.12, but with one zero at 0 Hz and complex zero pair at 17 kHz.

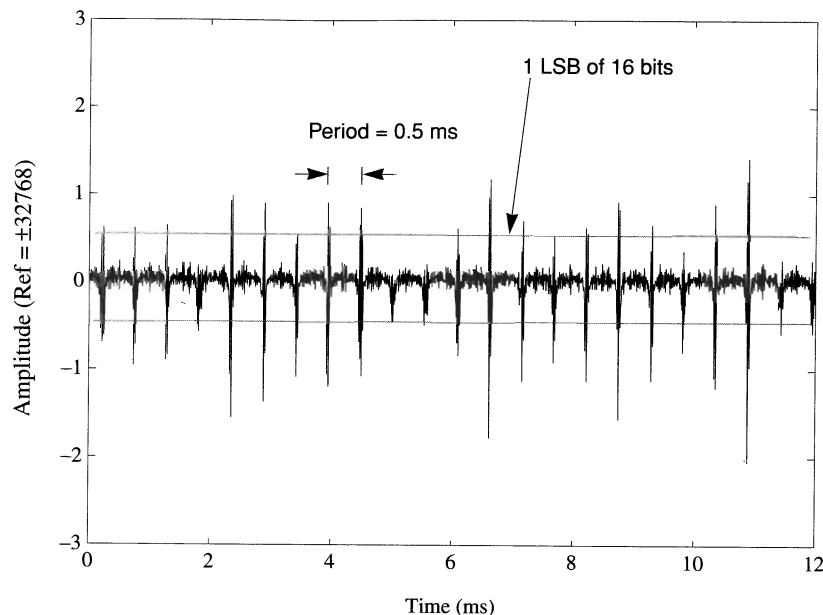


Figure 3.15 Similar to Figure 3.13, but with one zero at 0 Hz and complex zero pair at 17 kHz.

3.4.3 Fifth-Order Modulator

The next example illustrates the behavior of a fifth-order single-stage modulator having a single-bit quantizer. The practical application of this type of modulator has been previously reported [19–21]. This architecture can also be realized either as that of Figure 3.1 or 3.2. As with the third-order modulator case, an objective of these simulations was to keep the rms quantization noise power below -100 dB. Therefore, an oversampling ratio of 64 was selected for the fifth-order modulator. As before, the noise transfer function is obtained using MATLAB:

```
[B,A] = cheby2(5,80,1/64,'high'); % (Generate 5th-order Chebyshev-II filter for 64-times OSR)
Bn = B/B(1); % (Normalize numerator polynomial)
```

This produces noise transfer function poles at

$$\begin{aligned} z_{1,2} &= 0.93385118993510 \pm j0.16235498221603 \\ z_{3,4} &= 0.86349250359807 \pm j0.09274946005789 \\ z_5 &= 0.83947256388617 \end{aligned}$$

while the zeros are located at

$$\begin{aligned} z_{1,2} &= 0.99891041726966 \pm j0.04666881538284 \\ z_{3,4} &= 0.99958367616318 \pm j0.02885262886912 \\ z_5 &= 1 \end{aligned}$$

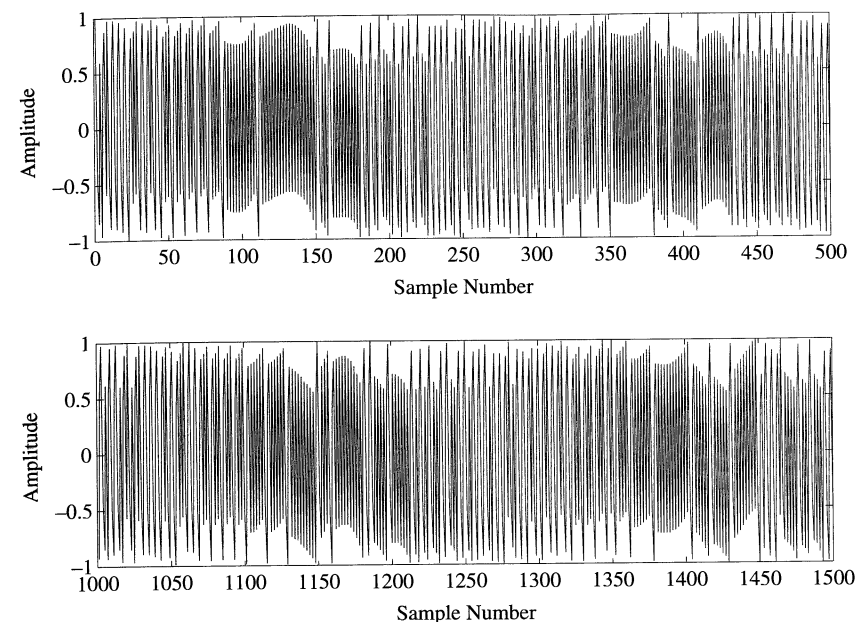


Figure 3.16 Quantization error sequence $e(n)$ of fifth-order single-stage modulator for dc input.

Figure 3.16 shows the quantization error sequence for a dc input level of $1/2048$ to the modulator. When comparing this figure with that shown for the second-order modulator in Figure 3.8, it is readily apparent that the structure of the noise is more complex, but certainly not random. However, the autocorrelation of the quantization error, shown in Figure 3.17, indicates strong periodicity every 512 sample shifts. The time-domain output from the modulator was filtered with either a sixth- or eighth-order Butterworth low-pass filter having a 3-dB corner frequency of 24 kHz, assuming the sampling rate is 64 times the Nyquist rate of 44.1 kHz. This type of filter has been proposed as a practical analog reconstruction filter for a fifth-order $\Delta\Sigma$ D/A converter [20]. Figure 3.18 shows an extended portion of the low-pass filtered time-domain output for the cases of (a) the sixth- and (b) the eighth-order Butterworth filter. The impulse train effect is clearly seen as in the lower-order modulators. The relatively high peaks of the impulses in the figures indicate more noise than that which is really present in the audio baseband and illustrate once again the issue of high peak-to-RMS ratio. This is due to the very large amount of out-of-band quantization noise that has not been completely removed by the filters. It is implicitly clear from the figure that the major contributions to the impulse peaks come from the higher frequency harmonics of the fundamental tone. The complexity of the postmodulator filter and the difficulty in completely removing the out-of-band noise are the principal disadvantages of higher order modulators. This is not a serious drawback for A/D design, where the filter is digital. However, for the D/A design, the low-pass filter is analog. It must have a sharp stopband rolloff. Figure 3.19 shows the passband power spectrum of the

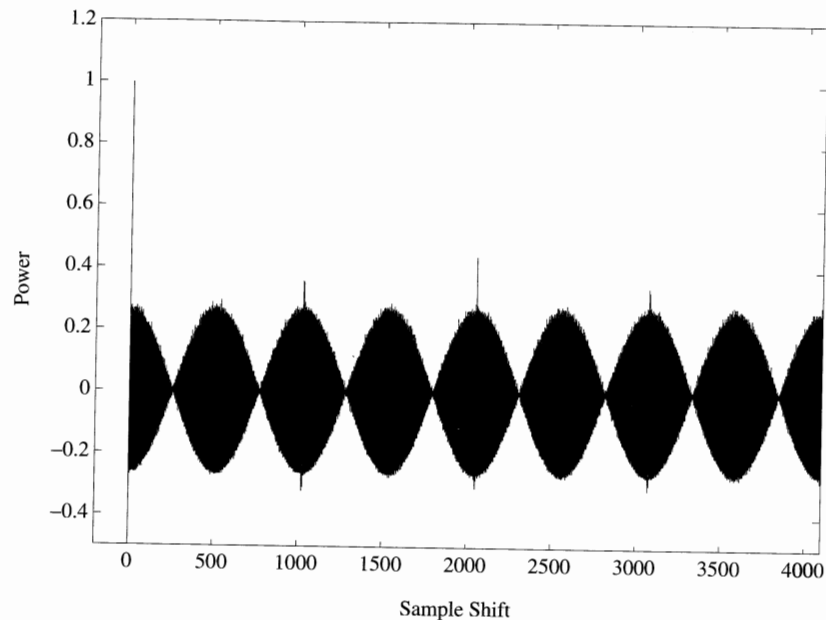


Figure 3.17 Autocorrelation of fifth-order modulator for dc input. Pattern noise repeats every 512 shift samples.

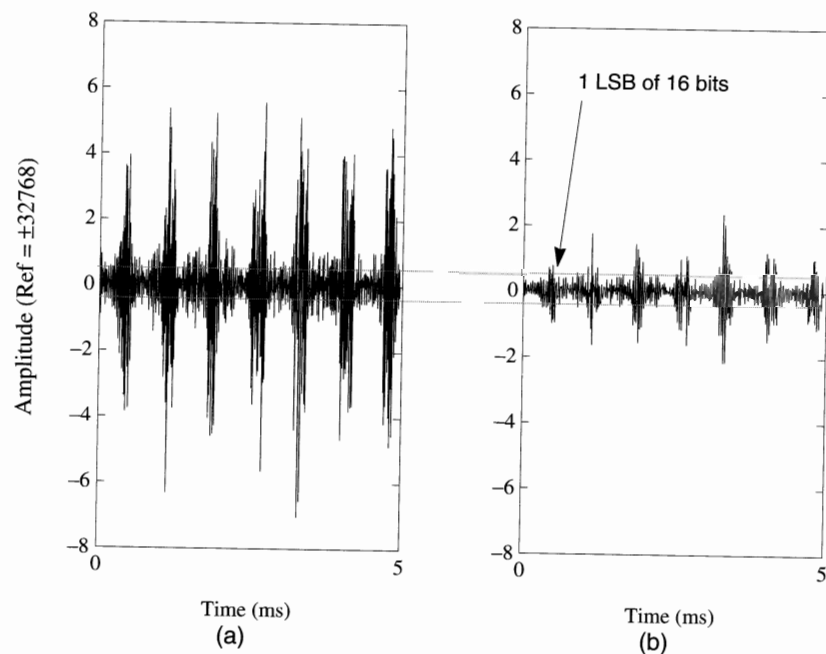


Figure 3.18 Time-domain sequence of fifth-order modulator for dc input, filtered after (a) sixth-order and (b) eighth-order Butterworth low pass.

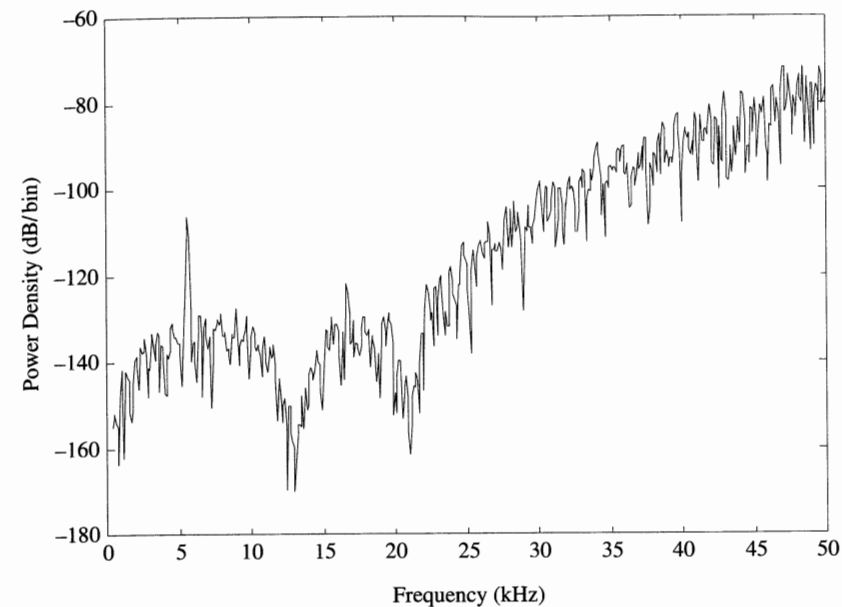


Figure 3.19 Audio baseband power spectrum of fifth-order modulator for dc input.

modulator for a dc input level of $\frac{1}{512}$ to the modulator. Using Eq. (3.9), the two visible tones in the baseband occur at $n = 4$ and its third harmonic $n = 12$.

3.4.4 Baseband Demodulation of Tones Near $f_s/2$

While each individual tone in the baseband has a relatively small magnitude, $\Delta\Sigma$ modulators produce very high powered tones near $f_s/2$ for nearly all classes of inputs. Figure 3.20 shows the spectrum near $f_s/2$ for the fifth-order modulator previously described. As with the simulation of Figure 3.19, the input signal to the modulator in the simulation is a dc level of $\frac{1}{512}$. The largest tone near $f_s/2$ occurs at $n = 1022$, based on Eq. (3.9). The rms power of this tone is -2.7 dB.

Based on Eq. (3.9), a more convenient expression for determining the tone frequencies near $f_s/2$ is given by

$$f'_0 = \frac{f_s}{2} \left(1 - \frac{k |A_{dc}|}{\Delta} \right) \quad k = \{1, 2, 3, \dots\} \quad (3.10)$$

From various simulations of this fifth-order modulator, it was determined that the largest tone near $f_s/2$ occurs at $k = 2$, and all other subharmonic tones visible in the spectrum occur at odd multiples of $k = 2$; that is,

$$k = \{2(1, 3, 5, \dots)\} \quad (3.11)$$

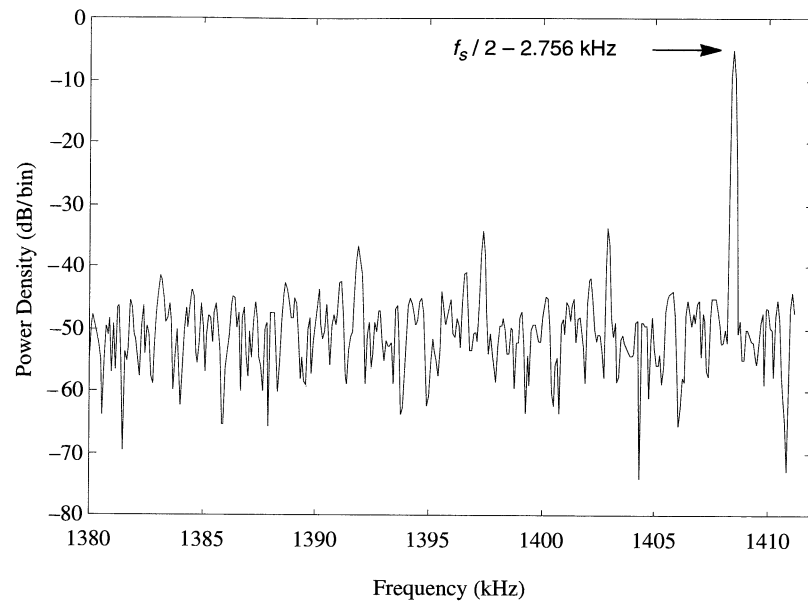


Figure 3.20 Quantization noise near $f_s/2$ for fifth-order modulator with dc input.

For this particular simulation, assuming that $f_s = 64$ (44.1) kHz, $|A_{dc}| = \frac{1}{512}$, and $\Delta = 2$, this largest tone is located at 1,408.444 kHz, which is about 2.756 kHz below $f_s/2$.

There are two physical mechanisms that can potentially cause these tones to demodulate and fold down into the baseband. The first such mechanism is clock noise at $f_s/2$ [46, 47]. Nearly any practical $\Delta\Sigma$ converter will include various clock dividers either on chip or off chip in the system, and these clock dividers will typically include a chain of binary dividers. Even the slightest amount of coupling of $f_s/2$ into the analog voltage references will perform this demodulation and potentially destroy the baseband performance. Figure 3.21 illustrates a simple mathematical model of how this demodulation occurs. In the illustration, an $f_s/2$ clock is attenuated by a value c before entering the voltage refer-

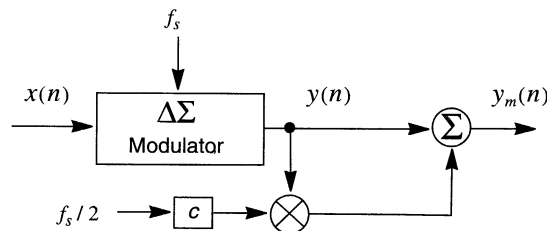


Figure 3.21 Model of coupling mechanism of an $f_s/2$ clock on the analog voltage reference.

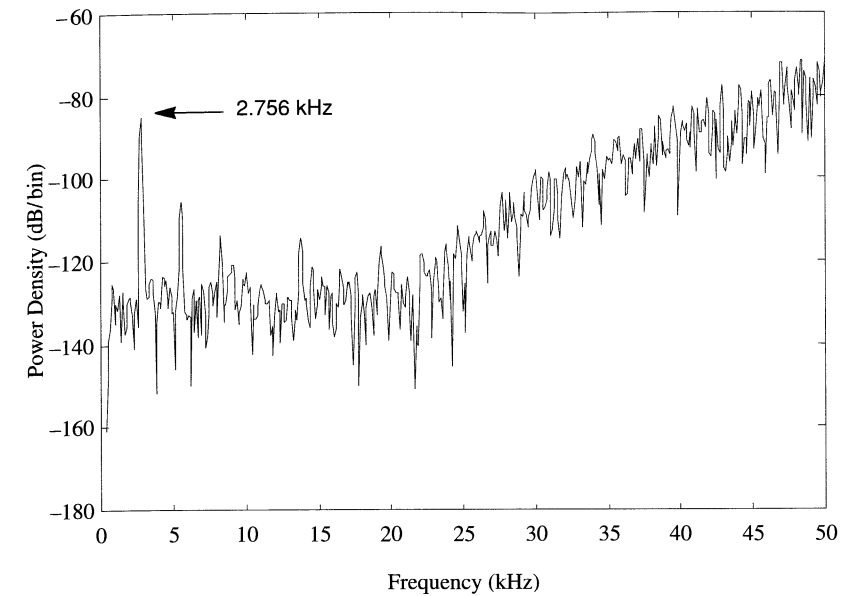


Figure 3.22 Corrupted baseband of fifth-order modulator by intermodulation with an $f_s/2$ clock with coupling coefficient $c = -80$ dB.

ence, where it effectively is multiplied by the bit stream output of the modulator and superimposed (summed) with the output. Figure 3.22 illustrates the baseband spectrum for the case where the coupling coefficient $c = -80$ dB. The entire noise floor near $f_s/2$ is folded down to baseband. The large tone at $(f_s/2) - 2.756$ kHz whose magnitude was originally at -2.7 dB is now located at 2.756 kHz with a magnitude of -82.7 dB.

The second physical mechanism potentially capable of demodulating the tones to the baseband is amplifier intermodulation distortion [48]. Two closely spaced tones anywhere in the spectrum, especially at high frequencies near $f_s/2$ where the amplifier is most non-ideal, could potentially result in their difference frequency in the baseband. This is a strong argument for trying to reduce or even eliminate *all* tonal behavior in $\Delta\Sigma$ modulators.

3.4.5 Higher-Order and Multibit Single-Stage Modulators

Other single-stage architectures were investigated. All single-bit architectures were tonal, even when the order was extended to eight. In addition, architectures with multibit quantizers were investigated. All showed the presence of idle channel tones for rational dc inputs. This is partly because dc conversion patterns only occupy a few levels when the dc input is small. Under these conditions, the behavior of multibit $\Delta\Sigma$ modulators is similar to that of single-bit $\Delta\Sigma$ modulators. For the cases when the input is either irrational or when the input has a small dither, most of the multibit architectures investigated produced a smooth-looking baseband spectrum, but their short-term autocorrelations often indicated

small periodicities or nonwhite characteristics. In at least one reported hardware implementation of a multibit $\Delta\Sigma$ A/D modulator [49], baseband tones were reported at levels between -80 and -90 dB.

3.5 TONES IN MULTISTAGE $\Delta\Sigma$ MODULATORS

It has been previously reported [3, 50] that certain types of multistage architectures are nontonal for dc inputs, but only under certain special conditions. Unfortunately these conditions are nearly impossible to realize [30]. The first condition is that the dc level must not be a rational number relative to the reference feedback level. This is generally impossible for digital modulators, since any practical implementation uses rational arithmetic. The second condition is that the cancellation circuitry (which removes the quantization noise from the earlier modulator stages) must perfectly imitate the noise transfer function of the loop filters in the prior stages. A digital modulator uses a digital loop filter which is perfectly deterministic; therefore, this is readily straightforward. For the analog modulator, however, this is impossible, since the loop filters are analog and their nonideal effects are never deterministic.

A variety of multistage configurations involving first- and second-order modulators are possible. One such architecture, which is commercially popular for audio D/A converters [24], entails a first-order modulator having a five-level quantizer, followed by a second-order modulator having a two-level quantizer. (The combination of a first-order modulator followed by a second-order modulator is sometimes called a 1–2 multistage structure.) This was simulated. As seen in Figure 3.23, the audio baseband spectrum shows the purely discrete nature of the tones. The low-pass filtered time-domain output, seen in Figure 3.24, shows the repetitious and impulsive nature of the pattern noise. As before, the fundamental period of the pattern corresponds to the lowest frequency (5.5 kHz) for this particular dc input level. The higher frequencies are located at multiples of 5.5 kHz. These higher frequencies, summing in phase with the lower frequencies, are most responsible for the high-amplitude distortion in the time domain. The reason for this is that the magnitudes of the higher frequencies are much greater. As seen with the third-order single-stage modulator in Figures 3.13 and 3.15, the peak-to-peak amplitude occupies between 2 and 3 bits of dynamic range out of 16 bits.

Section 3.4 discussed the addition of small input dither to the modulator as insufficient for removing all baseband tones in single-stage modulators. This type of dither has previously been explored for multistage modulators [50], wherein it was shown that a small input dither will generally smooth the baseband spectrum but will not make the quantization noise white.

Various multistage architectures were simulated, including two first-order modulators (1–1), three first-order modulators (1–1–1), a second- followed by a first-order modulator (2–1), two second-order modulators (2–2), and so forth. In each simulation model where the modulators were digital finite-state machines with no additive noise, all were tonal for undithered rational inputs. Further simulations revealed that small input noise or dither is much more effective if the multistage architecture employs a second-order first stage, such as a 2–1 or 2–2 structure, as opposed to a first-order first stage. However, noise or dither that is not shaped is less optimal. This will be discussed in Section 3.8.

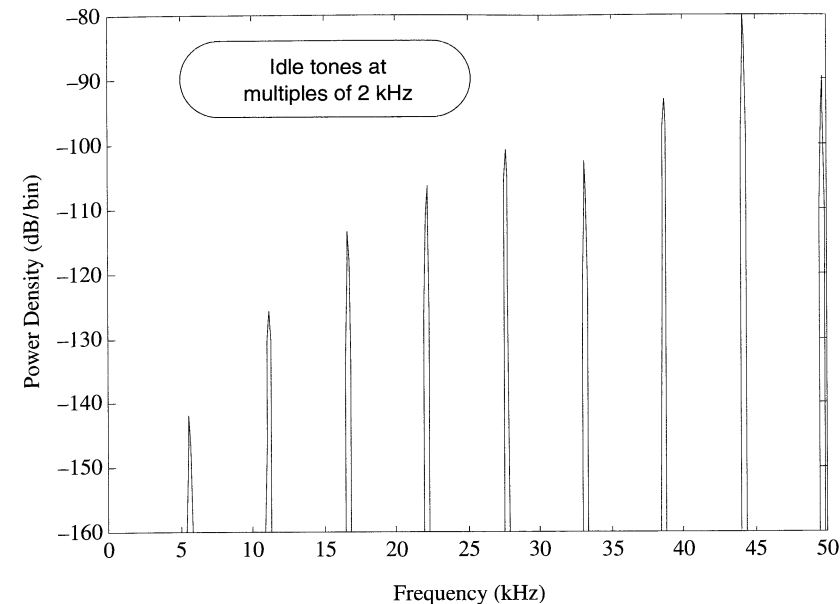


Figure 3.23 Baseband spectrum of 1–2 multistage modulator for dc input.

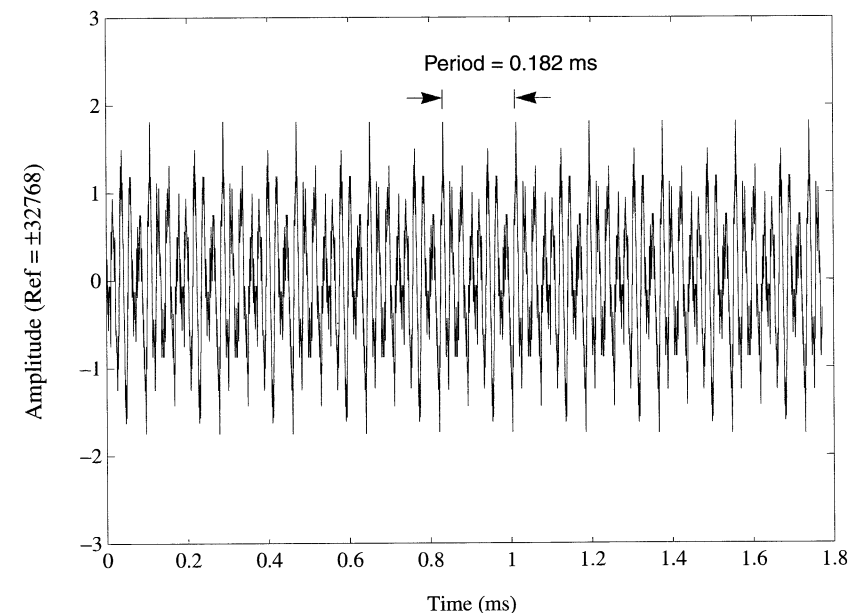


Figure 3.24 Low-pass filtered time output of 1–2 multistage modulator for dc input.

3.6 TONES IN $\Delta\Sigma$ CONVERTER HARDWARE¹

In-band sinusoidal test signals are typically not useful for detecting the audible presence of idle channel tones. Tones are also normally not present during null input signal tests.² Tones are most easily identified using a slowly moving (1 Hz or below) sine wave input. This is such a low frequency that it is nearly equivalent to gradually sweeping the dc level up and down. This allows quick indication of the amplitude ranges in which the modulator will be most sensitive to idle tones. The peak amplitude of the sine wave is varied from a few LSBs to full scale. Tones are usually most apparent with sine waves having a peak amplitude 20–60 dB below full scale. Typically tones will vary in pitch as the amplitude of the sine wave varies. At high amplitude levels, the tones will eventually move to a high enough frequency such that they are no longer in the audible band. At a low enough amplitude level, the tones will be subharmonic. This explains the absence of idle tones during null input signal tests. Idle tones may also manifest themselves as raspy sounds, whistling “birdie” sounds [51], motorboatlike sounds, or an increase in the noise floor as the sine wave passes through critical values that excite the modulator’s idle tones. Low-level tones can be identified by amplifying the output of the converter. Therefore, the converter output should first be high-pass filtered before amplification in order to remove the low-frequency input stimulus; this will prevent overloading the amplifier. The remaining noise is then played acoustically. It is also A/D converted and captured digitally onto digital audio tape (DAT). The digital output from the DAT is then loaded into computer memory for further analysis and postprocessing. A key element in verification of these tests entailed producing long simulations of similar $\Delta\Sigma$ architectures, producing the equivalent of many seconds of real-time output data at 44.1 kHz. The output from the simulations was then captured onto DAT so that it could be played back easily and repeatedly. The audible quality of the tones from the simulations could then be readily compared with the actual tones generated by the $\Delta\Sigma$ converter hardware.

While the above description helps identify tones related to dc input conditions, it is also possible to perceptually identify (hear) quantization errors related to ac input conditions where the ac input itself is in the audible range. Low-level ac inputs may excite periodic noise patterns resulting in a sound quality that is fuzzy or raspy in some manner.

All D/A converters tested were fed triangular probability density function (tpdf) dithered PCM input words, where the dither spanned two LSBs. As an experimental control, a multibit D/A unit having 18-bit resolution was subjected to the same tests to ensure that quantization effects from the PCM input stimulus were not causing any of the tonality observed in the D/A converter units. Converters that could accept only 16-bit inputs were fed dithered 16-bit words, while others that could accept 18-bit inputs were fed dithered 18-bit words.

1. A methodology for determining tonal behavior in $\Delta\Sigma$ converter hardware has previously been described by the author in [30]. In this same reference, results were presented for various hardware implementations of $\Delta\Sigma$ converters. This material is again presented here, along with some additional new or revised results.

2. Some D/A converters have a mute circuit that senses zero PCM, which then shuts off the analog output stage. This causes the converter to have an artificially low noise floor during this test. Here, converters without this function are discussed.

3.6.1 Third-Order Digital Modulator Test

Two commercial outboard D/A converter units were characterized, each containing the same third-order 192-times oversampling single-stage modulator chip. One unit had 18-bit input capability, so its performance was analyzed in order to resolve idle channel noise below the 16-bit level of input quantization and dither. Its noise floor was first measured: –103 dB rms for a –80-dB sine wave input at 1 kHz. Using the testing methodology previously described, the power spectrum and time domain were obtained, as shown in Figures 3.25 and 3.26, respectively. (The amplitude reference given in Figure 3.26 for the time domain is based on full-scale 18-bit levels, i.e., $\pm 2^{17}$.) The tone amplitude spans ± 2 amplitude units, which is 3 bits (8 LSBs) out of 18 bits (or 1 LSB out of 16 bits). The measured baseband tone levels from Figure 3.25 reveal the most dominant tone at –106 dB, with one nearby tone a few decibels lower. From a perceptual (aural) viewpoint, the idle channel tones were easy to hear above the noise floor.

A third D/A converter unit containing this same chip had significantly reduced idle channel tones when compared with the other two. This better unit had two separate physical subunits, one for the analog and one for the digital, with optical isolation of data and clocks between them. This most probably eliminated any effects due to high-frequency tones near $f_s/2$ aliasing into the baseband. This unit had only 16-bit input capability, so that the noise from the dithered 16-bit input (or possibly noise from the analog output stage) may have masked any idle channel tones that were actually present.

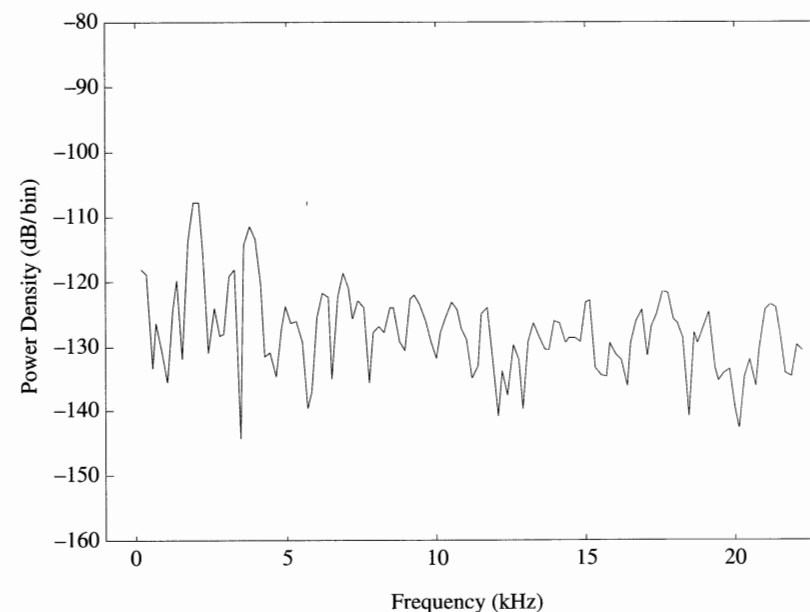


Figure 3.25 Measured power spectrum of commercial third-order audio D/A converter with 18-bit input.

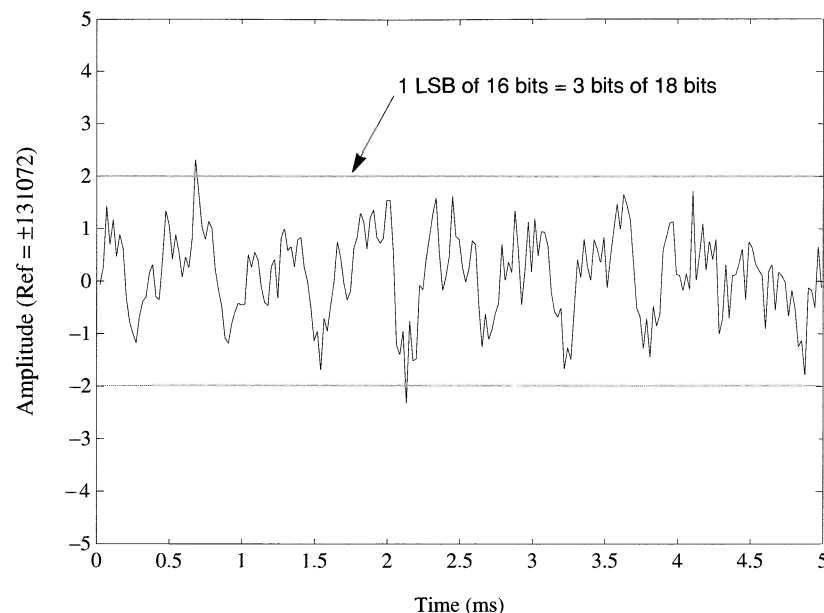


Figure 3.26 Captured time-domain output of commercial third-order audio D/A converter with 18-bit input.

3.6.2 Fifth-Order Digital Modulator Test

A fifth-order audio $\Delta\Sigma$ D/A converter was characterized. Test results showed that the converter was clearly tonal, as seen in Figures 3.27 and 3.28. Its noise floor was first measured: -100 dB rms for a -80 -dB sine wave input at 1 kHz. Like the third-order modulator just described, this one could accept 18-bit inputs, so the amplitude reference given in Figure 3.28 is also $\pm 2^{17}$. The tone amplitude spans ± 4 amplitude units, which is 4 bits out of 18 bits (or 2 bits out of 16 bits). Figure 3.28 has generally the same sawtooth behavior as seen in the simulated fifth-order modulator in Figure 3.18, and the relative amplitudes are nearly the same. The measured baseband tone levels from Figure 3.27, as seen in the frequency domain, reveal the most dominant tone at -97 dB, with several nearby tones only a few decibels lower. These levels are somewhat higher in amplitude than seen in the simulation from Figure 3.19. When the input words were limited to 16 bits and dithered at the 16-bit level instead of the 18-bit level, this 12 dB of additional input dither noise was still not enough to mask the tonal sound from this converter.

Regarding the nature of the sawtooth time-domain output pattern seen in Figures 3.18 and 3.28, nearly any $\Delta\Sigma$ converter will exhibit this type of output if it is not low-pass filtered heavily enough. The period of the sawtooth will generally be the difference period between $f_s/2$ and that of the dominant idle tone nearest to $f_s/2$, as described in Section 3.4.4. This low-frequency periodicity in the time domain can be seen whether or not there is any aliasing of these higher frequency tones into the baseband.

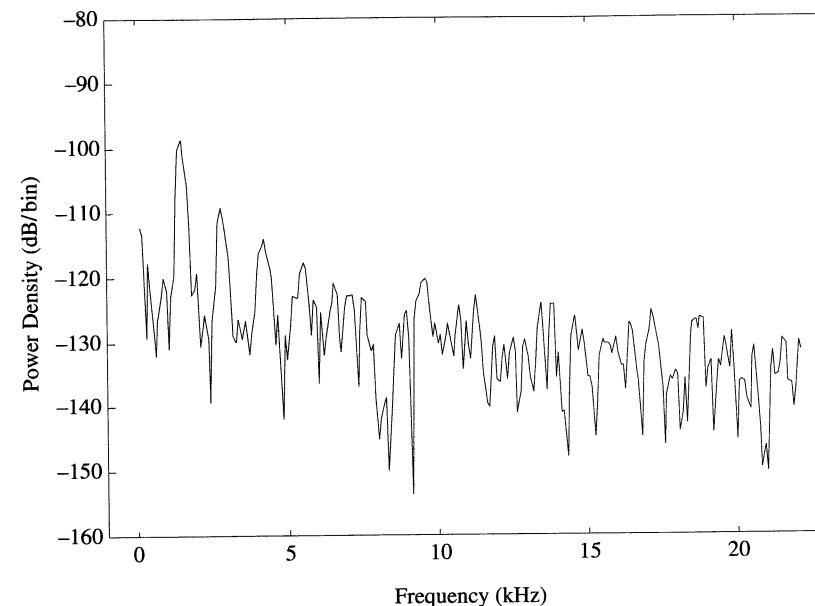


Figure 3.27 Measured power spectrum of commercial fifth-order $\Delta\Sigma$ audio D/A converter with 18-bit input.

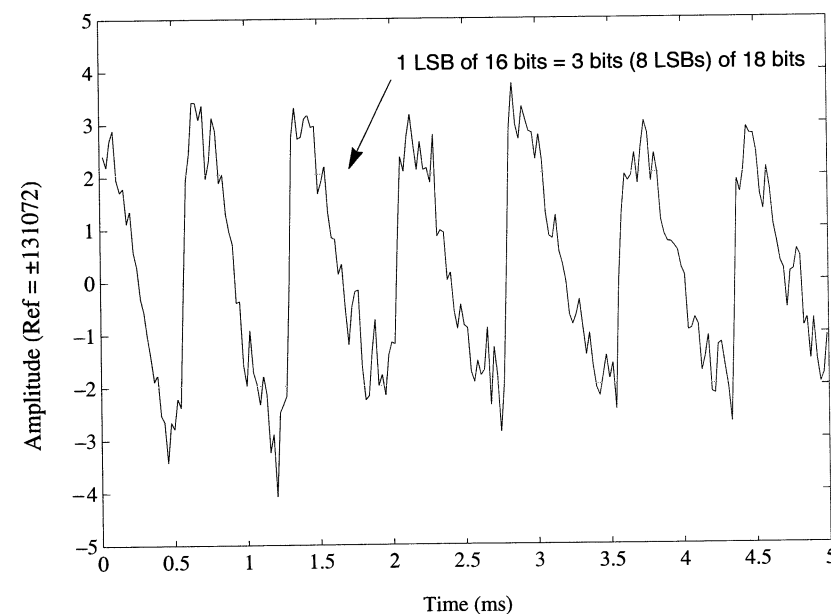


Figure 3.28 Captured time-domain output of commercial fifth-order $\Delta\Sigma$ audio D/A converter with 18-bit input.

3.6.3 Multistage Modulator Test

A 1–2 multistage $\Delta\Sigma$ D/A converter was obtained. Test results showed that the converter was not perceptually tonal. No tones were visible in the autocorrelation nor in the spectrum above -115 dB. There is a clear explanation for this result. As previously mentioned in Section 3.5, a rational input to an ideal multistage modulator will still produce a discrete spectrum of tones, whereas a small input dither to a multistage architecture will smooth the baseband spectrum. In all tests conducted in the hardware studies, the digital test signals were dithered with a 2-LSB tpdf dither signal. This multistage $\Delta\Sigma$ D/A converter only had 16-bit input capability rather than 18 bits. Therefore, the input dither spanned the 15th and 16th bits. This caused the baseband noise floor to degrade to -93 dB. Simulations confirmed that this was enough input dither to significantly smooth the baseband spectrum of this architecture. It was not known to the author whether the modulator employed some sort of built-in dither as well.

While no hardware implementations of multistage A/D converters were tested, the issue of idle channel tones is more important here than for multistage D/A converters, since imperfect matching and imperfect error subtraction result in uncanceled quantization errors. This will be discussed further in Sections 3.8.2 and 3.11.2.

3.7 DITHER IN PCM QUANTIZERS

Dithered PCM is a mature subject that has been carefully examined and characterized [52–65]. In order to establish a relationship between dithered PCM and dithered $\Delta\Sigma$, this section will briefly review a few main points concerning dithered PCM.

3.7.1 Nonsubtractive Dither

Chapter 2 reviewed the spectral characteristics of undithered PCM. An example of undithered PCM is shown in Figure 3.29. Without dither, the quantization process creates a highly correlated error composed purely of harmonic distortion components with no additional energy in between. Techniques for dithering PCM quantizers entail adding a pseudorandom noise source to the input of the quantizer. This is illustrated in Figure 3.30. The statistical properties of the output signal from the quantizer are dependent on the amplitude and probability density function of the dither. The dither amplitude typically spans the range of one LSB for rpdf dither, or two LSBs for tpdf dither. A rpdf 1-LSB dither will only decorrelate the first moment of the error with the input, whereas a tpdf 2-LSB dither will decorrelate the second moment of the error with the input [60]. Therefore, the autocorrelation of the error should appear decorrelated if at least a 2-LSB tpdf dither is used, since the autocorrelation function entails a second-order moment. The added noise penalty for a 2-LSB tpdf dither can be readily calculated as 4.77 dB.

The proper use of dither enables the resolution of input signals having amplitudes smaller than one LSB [57]. In undithered PCM, this is simply not possible, since the signal level is below the LSB level and cannot trigger the quantizer. It should be noted that if a proper dither is added at the input to a PCM A/D converter, then no additional dither is needed for the reconstruction of this quantized signal through a PCM D/A conversion.

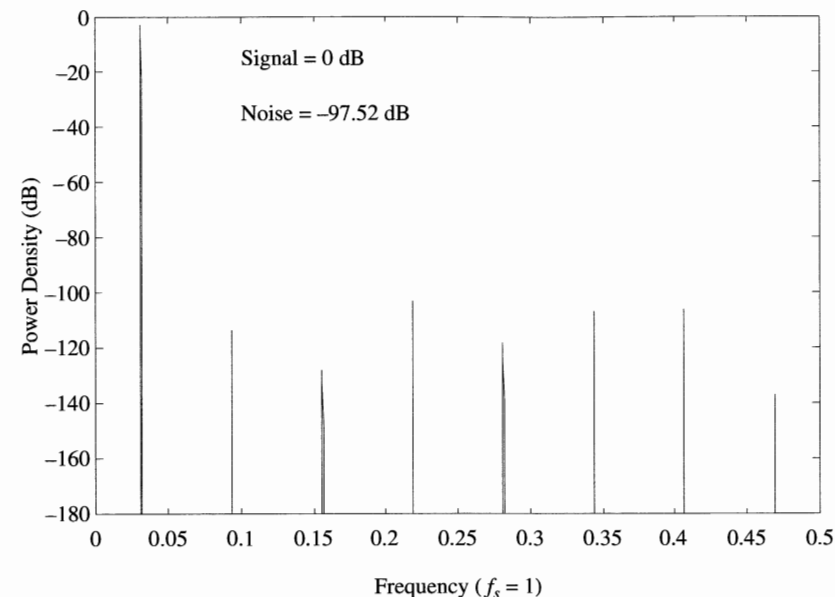


Figure 3.29 A 16-bit PCM without dither.

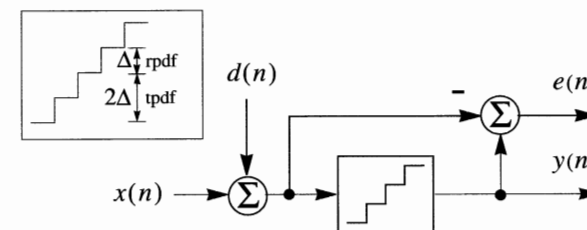


Figure 3.30 PCM quantization with dithering.

However, if the reconstruction process entails any digital signal processing, then additional dither may be required for truncation or rounding operations that occur to the signal while it is being processed in the digital domain.

3.7.2 Subtractive Dither

Subtractive dither [52–56, 59, 63–65] can be used to recover the loss of dynamic range that results from the addition of dither energy. In its simplest form, subtractive dithering is accomplished by subtracting the dither signal from the quantizer output, as shown in Figure 3.31. By way of example, Figure 3.32 shows the spectrum of a quantized sinusoid with nonsubtractive and subtractive dithering, wherein the input signal amplitude is smaller than one LSB. In the example, the quantizer has 16 bits, while the input is

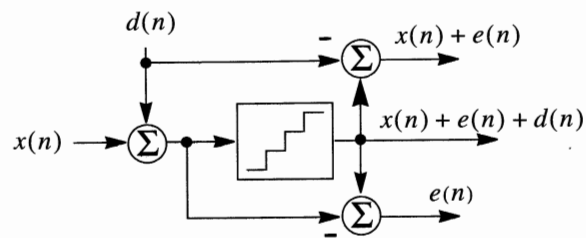


Figure 3.31 PCM quantization with subtractive dithering.

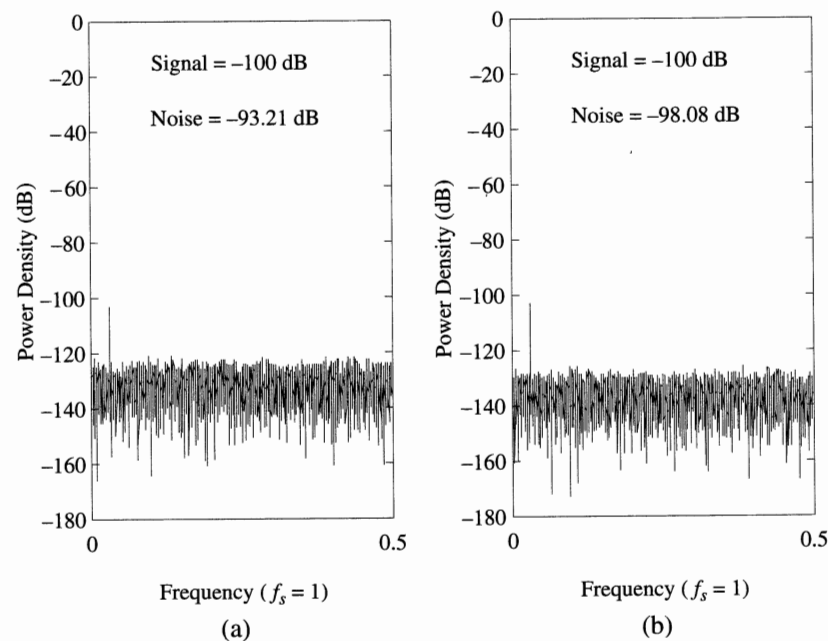


Figure 3.32 A 16-bit PCM with input below the LSB level and tpdf dither spanning 2 LSBs: (a) nonsubtractive dither; (b) subtractive dither.

-100 dB. The nonsubtractively dithered output has an integrated output noise power of -93.21 dB noise floor, while the subtractively dithered output noise is -98.08 dB. The output power is identical to the ideal theoretical value predicted for 16-bit quantization noise, assuming a white uncorrelated quantization error (which is impossible for undithered systems), whose output power is given by

$$\text{Error (dB)} = -(6.02b + 1.76) \quad (3.12)$$

There are several drawbacks to subtractive dithering. First, in an actual hardware implementation, the dither is typically a quantized digital signal, and the output of the

subtraction forms a quantized signal that requires more bits of precision than what comes out of the quantizer. Second, the input signal should be kept low enough to prevent the quantizer from overloading. For a 2-LSB dither, the peak input amplitude must be kept 2 LSBs below the overload level. If the quantizer overloads, the dither and the input signal are effectively clipped with a nonlinearity, hence, the dither cannot be completely removed with a simple linear subtraction element. Third, in an actual recording and playback setup, the dither must be synchronized at each end (transmit and receive) to be properly subtracted. A practical alternative to circumvent these problems has been proposed in [65].

3.8 DITHER TOPOLOGIES FOR $\Delta\Sigma$ MODULATORS

As seen in the previous section, nonsubtractive dithering in PCM quantizers improves the input-dependent errors due to quantization but causes the rms noise power in the baseband to degrade by several decibels. While subtractive dithering can remove this increased noise, it may sometimes be impractical to implement. An alternative is to shape the dither with a high-pass characteristic such that the dither energy is pushed out of the baseband.

The first example of such noise-shaped dither was reported by Limb in 1969 [54]. In this reference, dither was added to the quantizer input of a first-order multibit differential PCM coder. Other early examples of noise-shaped dither include dithered 1-bit Δ modulation [55]. However, the input to the quantizer of a Δ modulator or differential PCM coder is the same as the input to the modulator, so that the input is also shaped. In a $\Delta\Sigma$ modulator or noise-shaped coder, the input is not shaped.

3.8.1 Dither Topologies for Single-Stage Modulators

High-pass filtering of the dither in a $\Delta\Sigma$ modulator or noise-shaping coder architecture can be accomplished simply by adding the dither to the quantizer input, *not* to the input of the modulator [10, 13–15, 27–29, 35, 66, 67]. This is simply a direct extension of dithered PCM. In dithered PCM, the magnitude of the dither is established relative to the LSB Δ of the quantizer and spans 1 or 2 LSBs. If this same criteria is used in single-bit $\Delta\Sigma$ modulation, then it immediately becomes obvious that the dither magnitude is as large as the maximum input, while the portion of dither in the baseband is small, like that of the quantization error. What is not initially obvious, however, is just how much dither magnitude can actually be tolerated in a single-bit modulator before serious overloading of the quantizer occurs. This issue was previously addressed by the author in [29, 30], and later by others in [67]. It will be addressed in Section 3.9 of this chapter.

A block diagram showing the addition of the dither $d(n)$ into the quantizer of the $\Delta\Sigma$ modulator is shown in Figure 3.33. More generally, the goal is to make the dither transfer function (from dither input to the final output of the modulator) proportional to the quantization noise transfer function given in Eq. (3.1). Thus,

$$H(z) = \frac{Y(z)}{E(z)} = \frac{Y(z)}{D(z)} = \frac{1}{1 + F(z)G(z)} \quad (3.13)$$

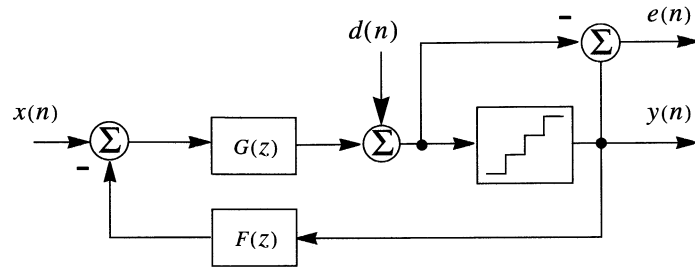


Figure 3.33 General $\Delta\Sigma$ modulator with noise-shaped dither.

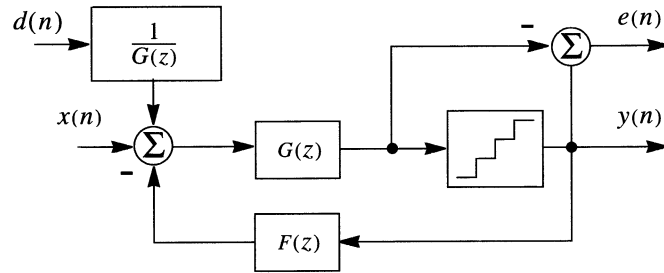


Figure 3.34 Equivalent system to Figure 3.33 but requiring separate noise-shaping dither prefilter at input.

Provided Eq. (3.13) is satisfied, the dither can be added anywhere inside the loop filter, as long as an appropriate prefilter processes the dither to produce the desired dither transfer function [29]. A generalized diagram for a single-stage modulator with such dither is shown in Figure 3.34. In this example, the dither $d(n)$ is prefiltered by $1/G(z)$ and then added to the input of the modulator. Equation (3.13) holds for this system. The penalties, however, are increased hardware complexity as well as the need for greater dynamic range inside the loop filter. When the dither is added to the input with prefiltering in this manner, the exact dither shape can be controlled independently of the quantization noise shape. Equivalent figures for noise-shaping coders with dither are shown in Figures 3.35 and 3.36.

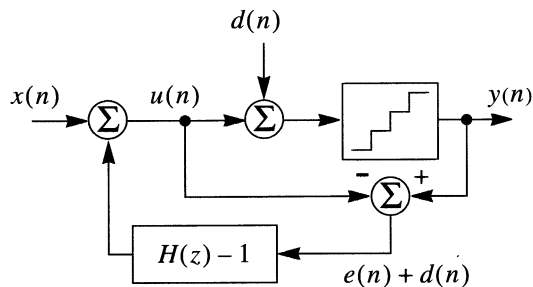


Figure 3.35 General noise-shaping coder with dither.

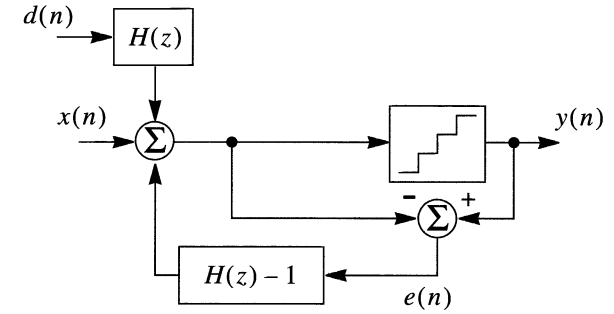


Figure 3.36 Equivalent system to Figure 3.35, but requiring separate noise-shaping dither prefilter at input.

3.8.2 Dither Topologies for Multistage Modulators

The first issue concerns why large-amplitude noise-shaped dither should be considered for multistage modulators when a small input dither may suffice. The answer is straightforward: Any added dither at the input that is not noise shaped may possibly degrade the noise floor in the baseband even more than if a larger noise-shaped dither is used. Input dither that is not noise shaped is considered less optimal in nearly any implementation.

In multistage $\Delta\Sigma$ architectures, each modulator should have its own independent dither to provide the most decorrelation of the quantization errors [29]. Initially one might argue that dither in stages other than the first stage is unnecessary because the quantization error from the first modulator in the chain is sufficiently *busy* so as not to excite tones in the stages that follow. However, there are less obvious reasons for dithering the following stages. In some architectures, this is more of an issue than in others. For example, in an analog implementation, there will be imperfect matching between stages, which will make the overall outcome more prone to uncanceled tones. Also, each stage is potentially capable of coupling higher frequency tones near $f_s/2$ into other stages.

In any case, the goal is to make the transfer function of the dither(s), as seen at the final output, the same as the quantization noise transfer function at the final output. This requires that each dither transfer function must be examined with respect to the final output of the overall architecture, *not* with respect to the output of its own modulator. Therefore, each dither (except the one in the final stage) must be prefiltered. Examples are shown in Figure 3.37 for a 1–2 architecture and in Figure 3.38 for a 2–1 architecture. The transfer functions for the dither signals and the quantization noise sources of the 1–2 architecture are given as

$$Y_1(z) = X(z)z^{-1} + D_1(z)(1 - z^{-1})^3 + E_1(z)(1 - z^{-1}) \quad (3.14)$$

$$Y'_1(z) = Y_1(z)z^{-1} \quad (3.15)$$

$$Y_2(z) = [E_1(z)z^{-1}]/\beta + [D_2(z) + E_2(z)](1 - z^{-1})^2 \quad (3.16)$$

$$Y'_2(z) = \beta(1 - z^{-1})Y_2(z) \quad (3.17)$$

$$Y(z) = Y'_1(z) - Y'_2(z) \quad (3.18)$$

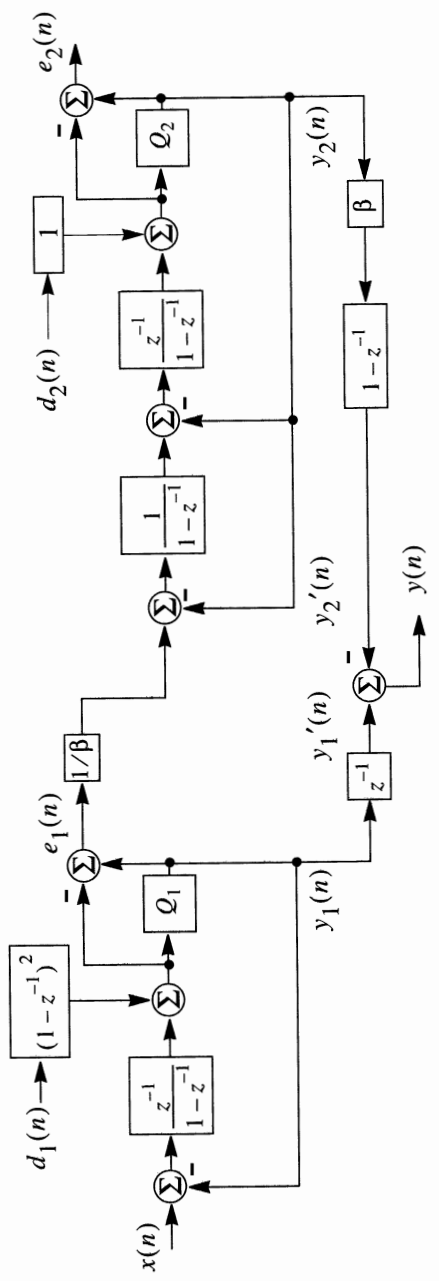


Figure 3.37 Multistage 1-2 structure with dither in both stages.

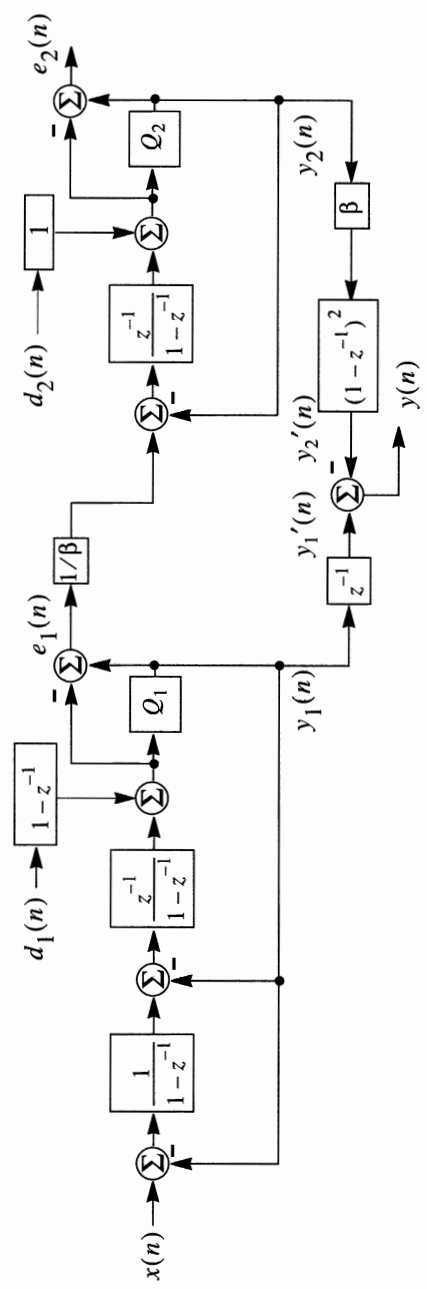


Figure 3.38 Multistage 2-1 structure with dither in both stages.

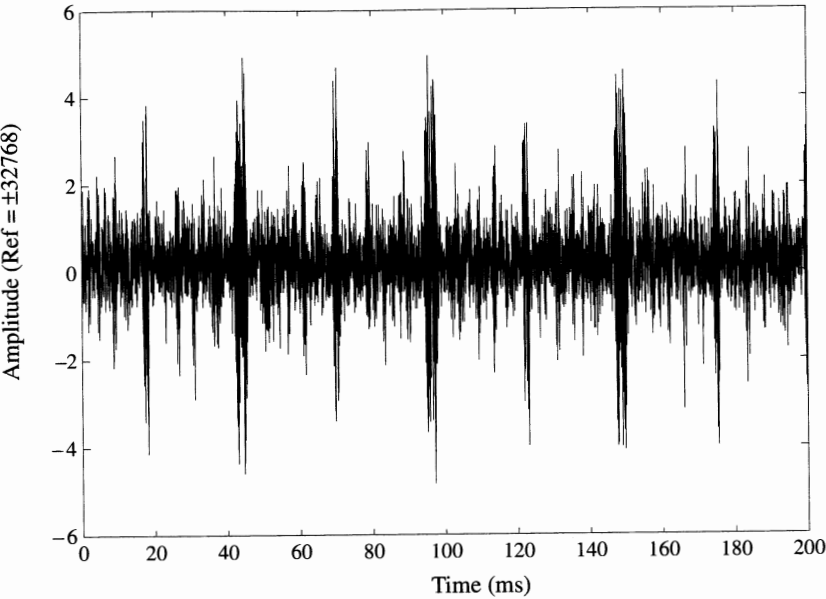


Figure 3.40 Captured time-domain output from undithered second-order $\Delta\Sigma$ D/A converter chip.

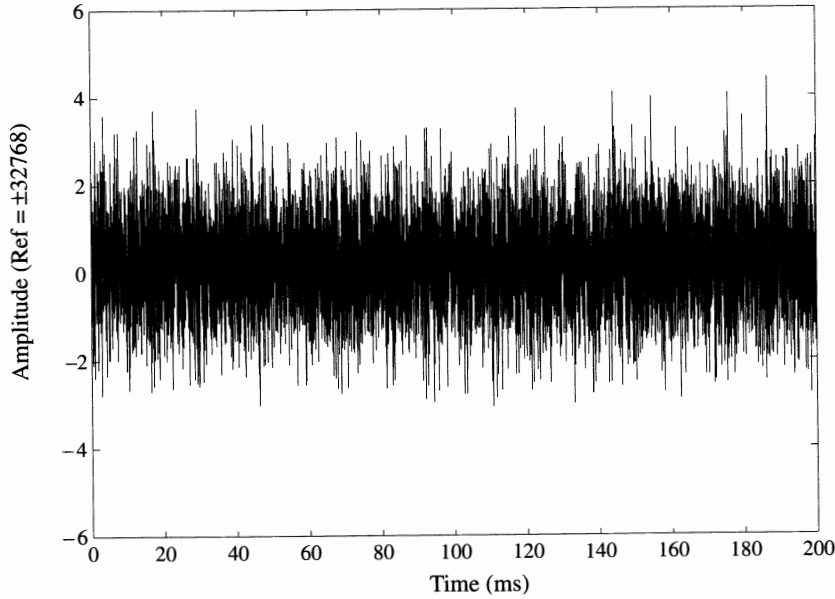


Figure 3.41 Captured time-domain output of dithered second-order $\Delta\Sigma$ D/A converter chip.

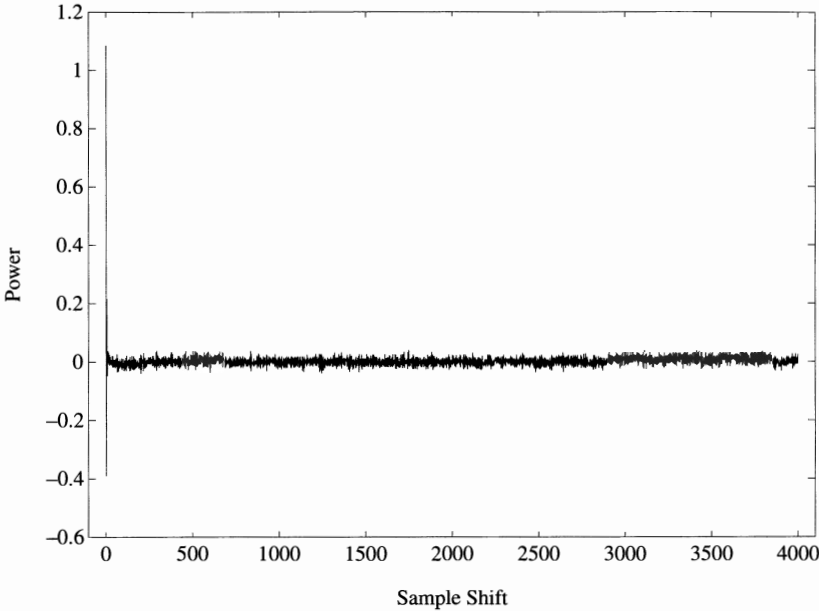


Figure 3.42 Autocorrelation of captured time-domain output of dithered second-order $\Delta\Sigma$ D/A converter chip.

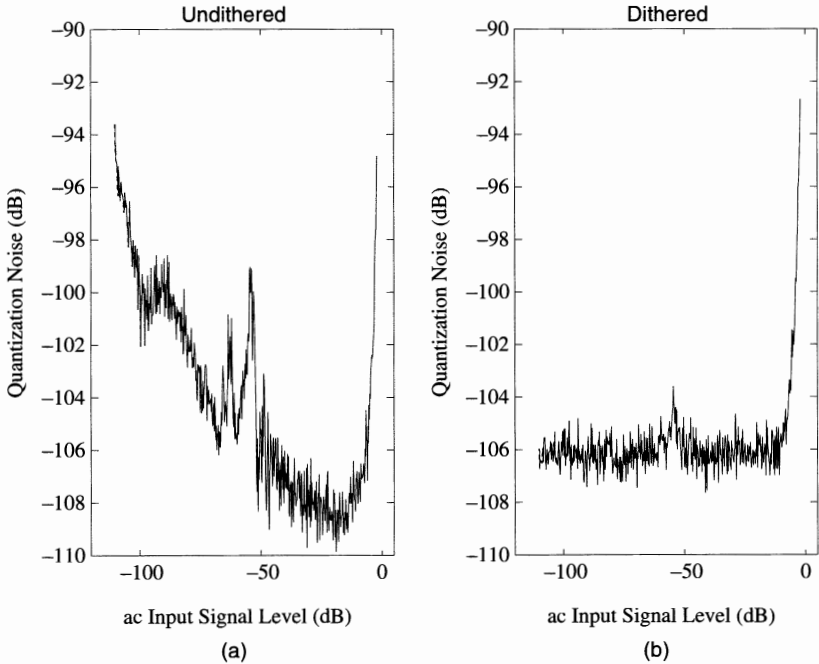


Figure 3.43 Quantization noise vs. ac input level for second-order modulator with 256-times oversampling ratio: (a) undithered; (b) dithered.

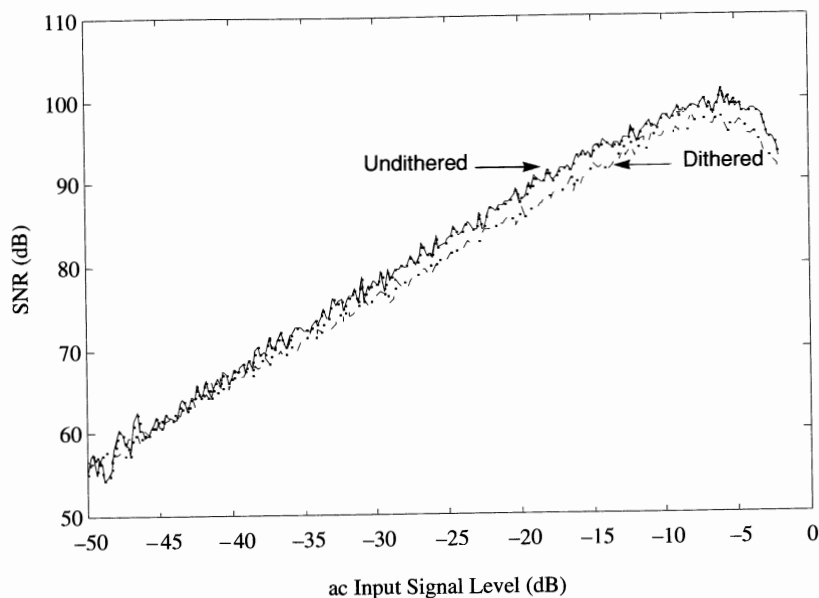


Figure 3.44 SNR vs. ac input level for second-order modulator with 256-times oversampling ratio: undithered (solid) and dithered (dashed).

below which the dither is not very useful. This was generally found true on all orders of modulators. With a tpdf dither of $\delta/\Delta = 1$, the quantization noise power further degrades, especially near the top of the dynamic range.

From empirical examination of the autocorrelation, use of tpdf dither was never found to be any better at whitening the quantization error than when rpdf dither was used. However, this comparison was made by forcing the power levels of the tpdf and rpdf dithers to be equal (i.e., rather than forcing their amplitude ranges to equivalency).

3.9.2 Third-Order Modulator

The same dither characteristics as in the previous case were also used for a third-order $\Delta\Sigma$ modulator: a rpdf dither spanning the interval $\delta/\Delta = 0.5$. The loop filter used in this simulation is identical to the one used for the simulation in Figures 3.14 and 3.15 as described in Section 3.4.2. Figure 3.45 shows the autocorrelation of the low-pass filtered output. Little or no periodic structure can be identified. The power spectrum appeared very smooth with no identifiable tones, and the low-pass filtered time-domain output revealed no impulsive effects.

In Figure 3.46(a), the rms baseband noise of the third-order modulator is plotted for small dc inputs ranging from -100 to -40 dB. It can be seen that the noise varies from -118 to -101 dB, depending on the level of the input signal. Figure 3.46(b) shows the noise of the converter with dither added. As can be seen, the noise level is almost constant

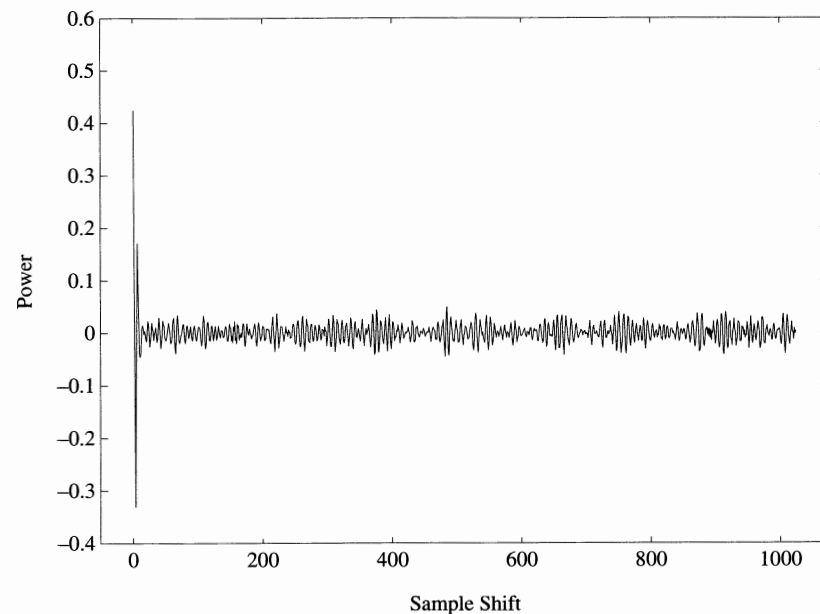


Figure 3.45 Dithered third-order modulator: autocorrelation of low-pass filtered output.

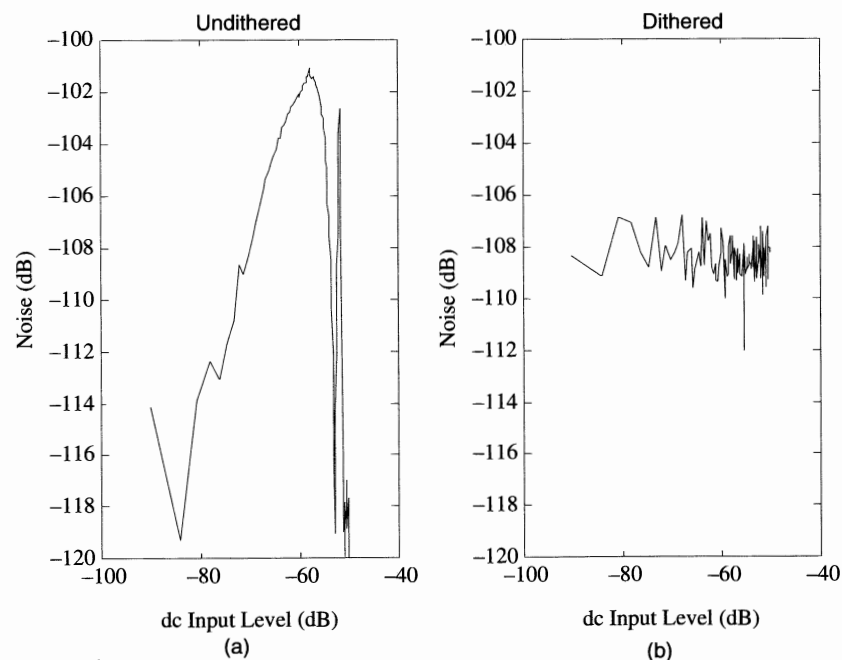


Figure 3.46 Noise vs. dc input level for third-order modulator: (a) undithered; (b) dithered.

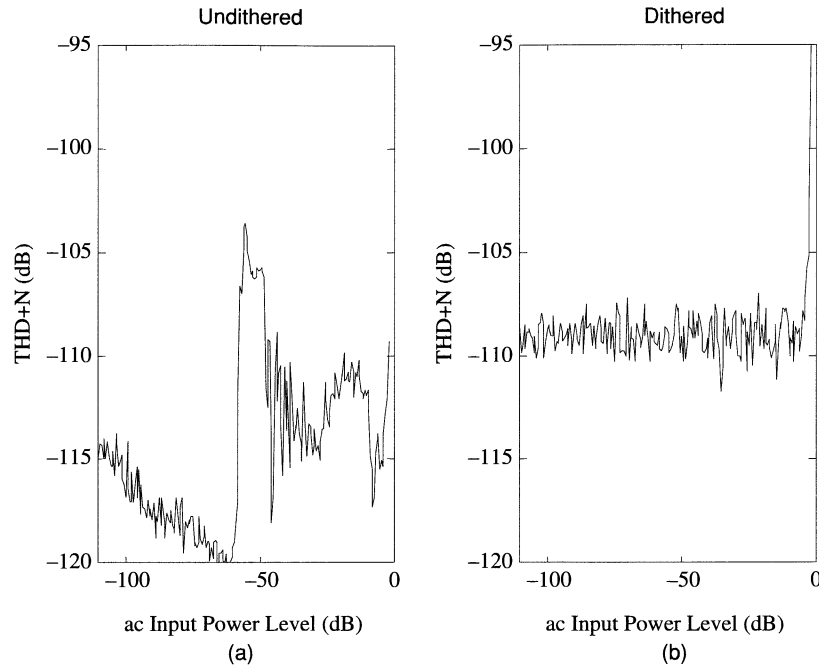


Figure 3.47 THD + noise vs. ac input power for third-order modulator: (a) undithered; (b) dithered.

at -108 dB. It is therefore evident that for some input levels the dithered converter has lower quantization noise.

Figure 3.47 shows a similar effect for an ac input signal. The rms baseband noise of the undithered coder ranges from about -120 to -105 dB. The dithered modulator remains constant at -109 dB, except near 0 dB, where the instability point occurs about 1 or 2 dB sooner as a result of the added power of the dither. In addition, for large inputs near full scale, the SNR degrades more significantly than in the case of the second-order modulator: The undithered modulator reaches a maximum SNR of 110 dB, but the dithered modulator reaches only 103 dB. In most practical implementations, it is usually difficult to achieve distortion below -100 dB for full-scale analog signals anyway, so this may not pose a serious problem. However, for the highest performance systems, this increase in noise for high input amplitudes may be a more serious drawback, especially for high-order single-stage single-bit modulators. Fortunately, this loss of high-amplitude dynamic range can be completely mitigated through the use of *dynamic* dither, which will be described in Section 3.13.

3.9.3 Fifth-Order Modulator

The same dither characteristics as in the previous two cases were also used for a fifth-order $\Delta\Sigma$ modulator. The autocorrelation of the quantization error is shown in Figure 3.48.

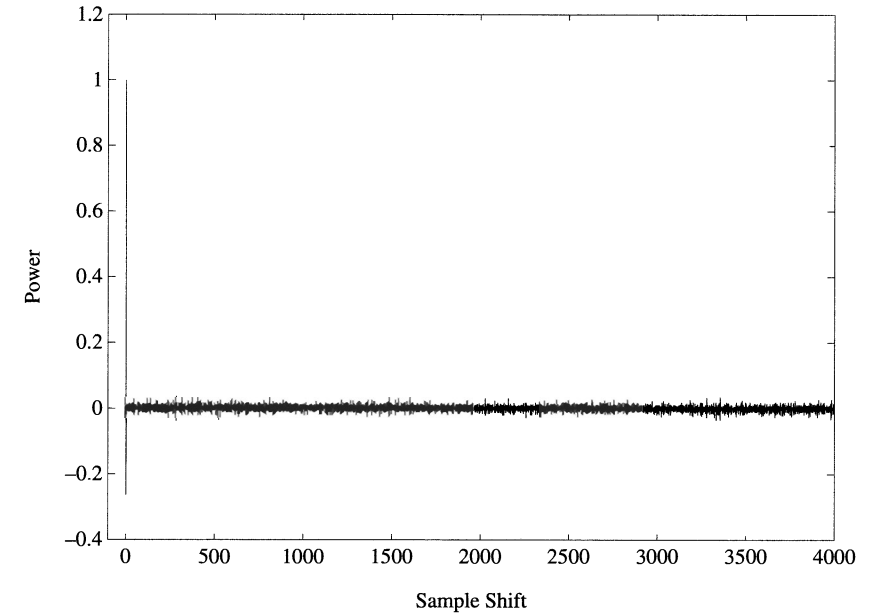


Figure 3.48 Dithered fifth-order modulator: autocorrelation of quantization error, $e(n)$. Same dc input conditions as in Figure 3.17.

This figure shows that it is free of periodic behavior. The loop filter used in this simulation is identical to the one used in the simulation of Figure 3.17.

3.9.4 Effect of Dither on Tones Near $f_s/2$

In Section 3.4.4, the detrimental effect of high-powered tones near $f_s/2$ was described, and an example was shown in Figures 3.20 and 3.21. Returning to that same fifth-order modulator example, the modulator was resimulated with dither having the same characteristic: rpdf of $\delta/\Delta = 0.5$. The result is a dramatic reduction and near elimination of these tones. This is shown in Figure 3.49(a) and (b). Figure 3.49(a) is the undithered result (the same as Figure 3.20), while Figure 3.49(b) is the dithered result. The most dominant tone at -2.7 dB is reduced to -30 dB with dither. The other subharmonics of the dominant tone are gone. Such reduction of these tones makes the modulator much less sensitive to intermodulation and aliasing of the tones into the baseband. In this example, the aliasing effects are nearly 30 dB less sensitive due to dither. Also, this use of dither is nearly the equivalent of providing an extra 30 dB of electrical isolation between the analog and digital sections at these frequencies in an actual hardware implementation.

Other modulator architectures were simulated for the purpose of examining reduction of tones near $f_s/2$ with dither. All single-bit modulators higher than second order responded with similarly good results, but the second-order single-bit modulator showed almost no reduction of these higher frequency tones with dither applied. However, when at least one extra quantization level was added, the dither significantly reduced these

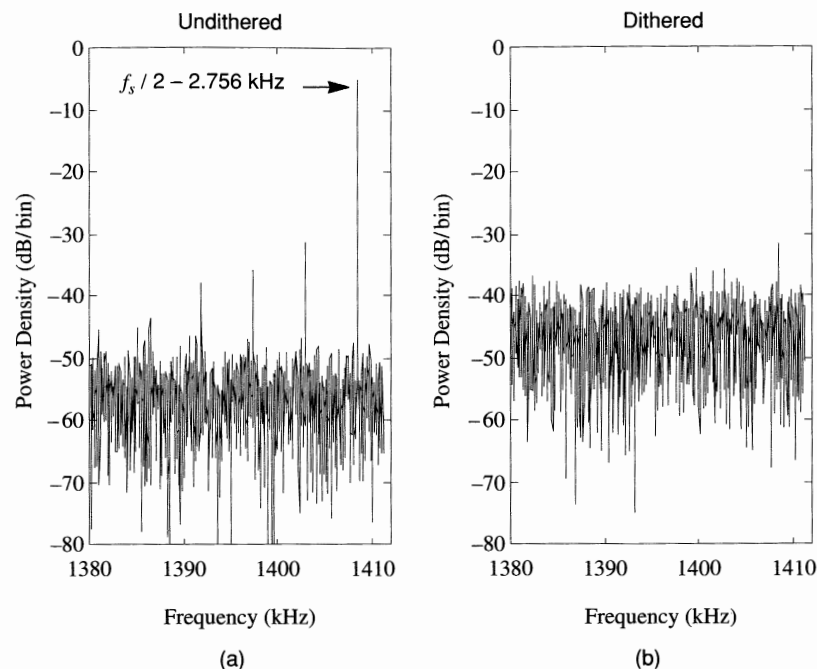


Figure 3.49 Quantization noise near $f_s/2$ for fifth-order modulator: (a) undithered; (b) dithered.

tones, but more relative dither was required. For example, given a three-level quantizer and $|A_{dc}| = \frac{1}{256}$, the dominant tone magnitude was -15 dB for the undithered case; for rpdf dither of $\delta/\Delta = 0.5$, the tone was -20 dB; for tpdf dither of $\delta/\Delta = 1$, the tone was -25 dB; when the tpdf dither was increased to $\delta/\Delta = 2$, the tone was -30 dB. Only in this last case was the tone reduced to the level of the surrounding frequency bins. Not surprisingly, higher relative dither levels for multilevel quantizers are tolerable in terms of the dynamic range penalties, since the additional levels allow the quantizer to operate within the no-overload region.

With the previous example in mind, it may first seem contradictory that the autocorrelation for a second-order single-bit dithered modulator in Section 3.9.1 was seen to have a smooth and nonperiodic characteristic, as shown in Figure 3.42. This is actually not a contradiction at all because the data was taken at the analog low-pass filtered output where it would normally be seen. Dithering this modulator adequately smooths the baseband. Also, in this particular hardware implementation, there was no measurable intermodulation of the higher frequency tones near $f_s/2$ into the baseband.

3.9.5 Multistage Modulators

Multistage modulators respond in a similarly good fashion if the dither is properly applied to each quantizer, as discussed in Section 3.8. The spectra of a 1–2, 1–1–1, 2–1, and 2–2 were all observed to be free of tones with the dither applied.

3.10 DITHER GENERATION

In most practical designs, the dither will be generated digitally, therefore it will be a quantized signal. A maximal-length sequence generator, otherwise known as a PN sequence generator [70], can be used to conveniently generate a pseudorandom noise. In order to eliminate any audible artifacts of the repetition of the sequence, it is wise to choose a sequence length that spans at least several seconds in real-time implementation. For example, assuming the modulator oversampling frequency is 10 MHz and the sequence generator is 26 bits, then the repetition time for the pseudorandom sequence would be $2^{26}/(10^7 \text{ Hz}) = 6.71 \text{ sec}$.

Simulations show that the dither will be ineffective if it is quantized down to just 1 bit. The dither performs reasonably well with only three levels, but as many as eight quantization levels may be needed to asymptotically approach the quality of effectiveness of unquantized dither. This finding has little consequence for digital modulators, but is highly significant for analog modulator implementation.

3.11 DITHER IN A/D MODULATORS

Up to this point, this chapter has been primarily focused on the elimination of tones found in D/A modulators. Analog-to-digital modulators also benefit from dither, as the next example will show.

3.11.1 Single-Stage A/D Modulator Example

A second-order single-stage single-bit A/D modulator test chip was built. As previously described for the D/A, pseudorandom rpdf dither of $\delta/\Delta = 0.5$ was injected into the quantizer input. Three bits of the digital dither are converted into eight analog levels that are summed with the output of the second integrator (input to the quantizer). Figure 3.50 shows the spectrum of the undithered modulator. The input signal is a 1-kHz sinusoid at -70 dB. Discrete unwanted tones are clearly seen near dc and in the vicinity of the input signal. Figure 3.51 shows the same modulator under the same test conditions but with dither. The tones are reduced by at least 10 dB, where they are virtually indistinguishable from the noise floor. The total rms noise floor of this dithered modulator is actually improved by 2 dB. This should not be surprising since most $\Delta\Sigma$ modulators have sensitive ac amplitude levels that cause more colored noise, as previously seen in the simulations of Figures 3.43 and 3.47.

3.11.2 Multistage A/D Modulators

As previously discussed in Section 3.5, small gain mismatches will result in imperfect noise cancellation in multistage modulators. The effect can be seen in Eqs. (3.14)–(3.19). With a gain mismatching, some of the quantization noise $e_1(n)$ from the first stage will appear at the output of the modulator. The noise at the output can therefore have the undesirable appearance of a lower order modulator. As previously reported in [29], the 2–1 multistage modulator of Figure 3.38 was simulated with coefficient and

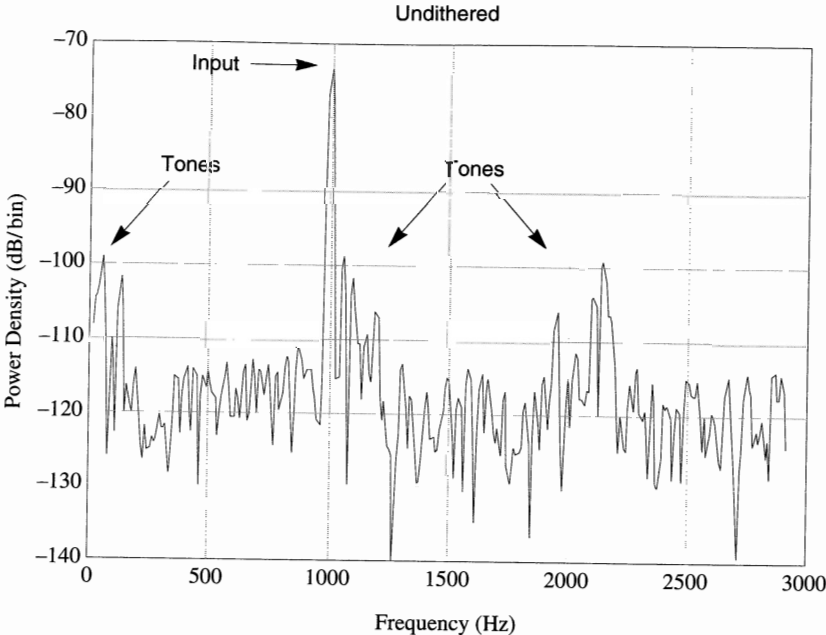


Figure 3.50 Measured power spectrum of *undithered* $\Delta\Sigma$ A/D converter test chip. Input signal is 1 kHz at -70 dB.

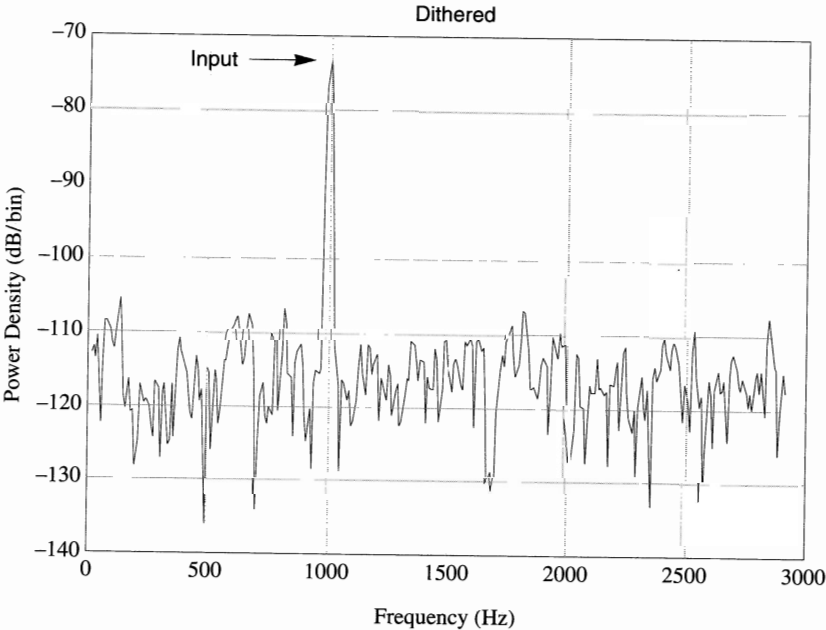


Figure 3.51 Measured power spectrum of *dithered* $\Delta\Sigma$ A/D converter test chip. Input signal is 1 kHz at -70 dB.

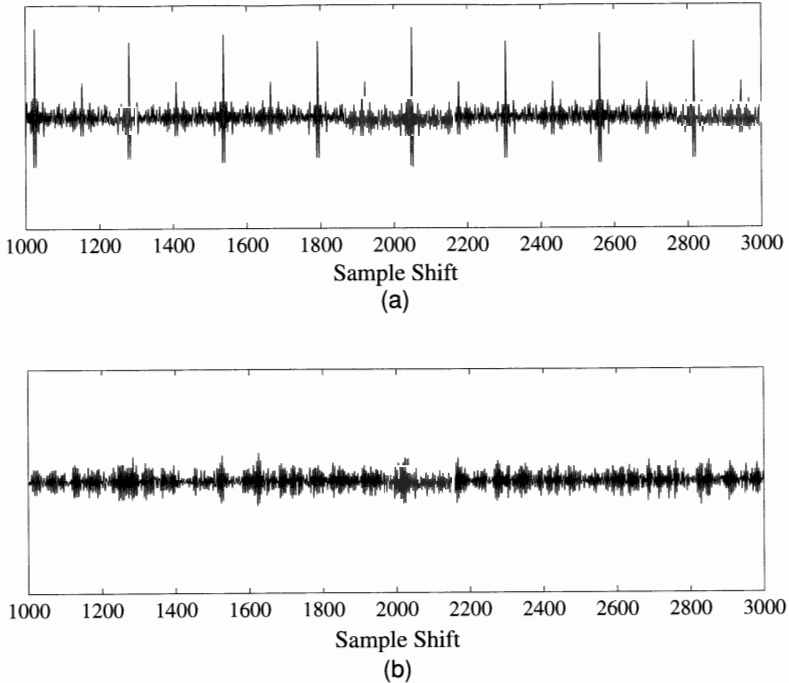


Figure 3.52 Autocorrelation of quantization error as seen at the output of 2–1 multistage modulator due to 0.5% gain mismatching: (a) undithered; (b) dithered.

gain mismatches ranging from 0.1 to 0.5%. A $\delta/\Delta = 0.5$ was used for each dither source. Figure 3.52 shows the result of the simulations. Noise periodicity is clearly seen in the autocorrelation, but with dithering the periodicity is clearly gone. These results suggest that dither is an important consideration in the implementation of multistage A/D architectures.

3.12 SUBTRACTIVE NOISE-SHAPED DITHERING

In Section 3.7.2, the practical problems associated with subtractive dithering for PCM were discussed. When applying the concept of subtractive dithering to $\Delta\Sigma$ modulation, there are new problems that have to be addressed. First of all, dithering a single-bit modulator will normally produce overloading effects in the quantizer. The result is a clipped dither that cannot be well removed by subtraction. One solution would be to employ multibit quantization to keep the modulator from overloading. Of course, the problems associated with precise multilevel quantization are well known [71]. In addition, another noise-shaping filter must be built in order to shape the dither prior to subtraction at the modulator output. This will work theoretically, but the resulting output signal must contain many more bits of quantization than that of the original quantizer within the loop. Simulations indicate that if these additional bits were simply applied toward increasing the

number of bits in the loop quantizer instead, and then nonsubtractive dithering is employed, the result will be far better than using subtractive dithering.

An alternative that employs another form of subtractive dithering has been described in [72]. In this system, two parallel differential first-order modulators are employed as the first stage of a multistage 1–1–1 A/D modulator. The dither signal (a square wave dither is used in [72]) is added to both modulators as a common mode signal, while the main input signal is added differentially. Then the outputs of the two modulators are subtracted, cancelling the dither while doubling the gain of the main input.

In order for this scheme to work, however, the modulators have to be very well matched, and the input must be kept low enough to prevent each quantizer from overloading and also prevent uncorrelated nonlinearities from dominating. By having to restrict the input level or the dither level in this manner, the potential benefits are somewhat diminished. The same basic scheme has also been proposed for a D/A modulator [73], wherein the modulators are purely digital, therefore the signal correlation between modulators is closer to ideal.

3.13 DYNAMIC NOISE-SHAPED DITHERING¹

3.13.1 Theory of Dynamic Dither

As previously described in Section 3.9, when a $\Delta\Sigma$ modulator is dithered with a sufficient amount of power to remove tonal behavior in the quantization error, the noise floor increases by several decibels for the largest input levels. This occurs because the quantizer becomes heavily overloaded with the combination of a large input and a large dither. The desired objective, therefore, is to find a way of dithering that preserves the basic quality of removing tonal behavior from the modulator, while also preventing additional degradation due to overloading and instability effects [74]. Subtractive dithering per se cannot accomplish this because the dither is removed after the overloading damage has already been done.

To begin with, the quantization error will be most tonal and correlated with the input when the input is relatively small. With most $\Delta\Sigma$ modulator architectures, the quantization error spectrum becomes increasingly smoother and less correlated for larger input signal levels. In addition, quantization errors tend to be perceptually masked by the human ear [75] when the input is a certain level above the quantization error.² This leads to the idea that the dither can somehow be decreased as the input magnitude increases.

The overloading of the quantizer occurs instantaneously on a sample-by-sample basis, not on an rms basis, since the quantizer is memoryless. Therefore, what is needed is instantaneous dynamic modulation of the dither by a function of the input magnitude.³

1. The material in this section was drawn from an earlier work by the author as found in [74].

2. The perceptual masking of quantization errors is a well-known phenomenon that forms the basis for low bit-rate coding of audio signals.

3. This approach should not be confused with automatic gain control (AGC), which entails low-pass filtering or long-term rms averaging, which is inappropriate and useless in this context.

The objective is to modulate the pseudorandom dither $d(n)$ by the input magnitude $|x(n)|$ in a manner that produces full dither as $|x(n)| \rightarrow 0$, a continuously diminishing amount of dither as $|x(n)|$ increases, and zero dither as $|x(n)| \rightarrow 1$. An expression that produces this objective is given by

$$d_m(n) = d(n)(1 - |x(n)|^\alpha) \quad 0 < \alpha \leq 1 \quad |x(n)| \leq 1 \quad (3.20)$$

where α is an exponential parameter used for compressing the dither in magnitude with increasing input magnitude. In typical practice, almost no dither is needed for the top 20 dB of the modulator's dynamic range. In order to compress the dither enough when the largest input samples are present, it has been found empirically that α should be less than unity. There is a trade-off in choosing this value: If α is too small, not enough dither is present to remove tones for low-level inputs, while if α is too large, the benefits of the technique are diminished for high-level inputs.

The input/output relation of a $\Delta\Sigma$ modulator or noise-shaping coder with this modulated dither is given by

$$Y(z) = X(z) + E(z)H(z) + D_m(z)H(z) \quad (3.21)$$

where $D_m(z)$ is the z -transform of the modulated dither term $d_m(n)$, given by

$$D_m(z) = D(z) - \{D(z) \otimes \mathcal{Z}[|x(n)|^\alpha]\} \quad (3.22)$$

where $\mathcal{Z}[|x(n)|^\alpha]$ is the z -transform of $|x(n)|^\alpha$. The multiplication of the dither in the time (sample) domain results in convolution in the frequency domain. Since the dither is assumed a wide-sense stationary uniform noise source, then the convolution $D(z) \otimes \mathcal{Z}[|x(n)|^\alpha]$ will result in another noise that is white and uncorrelated with the input! This result is similar to that derived in Section 3.3. At first glance, this result may seem somewhat surprising and counterintuitive. Some might immediately respond that modulating the dither in this manner will produce a pumping of the noise floor with the input, but this effect is inaudible in all practical situations. Only in the most extremely unrealistic case could it be audible for ultra-low-frequency subaudio signals that are heavily oversampled and varying widely in amplitude with no other signals superimposed.

The third-order modulator previously described and simulated in Sections 3.4.2 and 3.9.2 was resimulated using the dynamically modulated dither of Eq. (3.20). Figure 3.53 shows the baseband SNR versus ac input signal level, while Figure 3.54 shows the variation of baseband noise, which is a combination of quantization error and dither. For the case of $\alpha = \frac{1}{4}$ in particular, the SNR is smoothly rising with increasing input level, reaching a peak SNR of 110, the same as the undithered case. The overload point occurs at the same input level as the undithered case. The noise level can be seen as gradually diminishing from -108 to -115 dB as the input level increases near full scale. For this case, it was observed that the dither was still sufficient at low signal levels to remove tones and correlated noise. For values of α smaller than $\frac{1}{4}$, little or no further benefit results in terms of SNR improvement, while there is a danger that too small a value of α will produce an insufficient amount of dither, especially at lower signal levels. Somewhat larger values of α still result in SNR performance that is better than that produced by static dither, as shown for the case $\alpha = \frac{1}{2}$, however, there was no observed benefit of making the value this large.

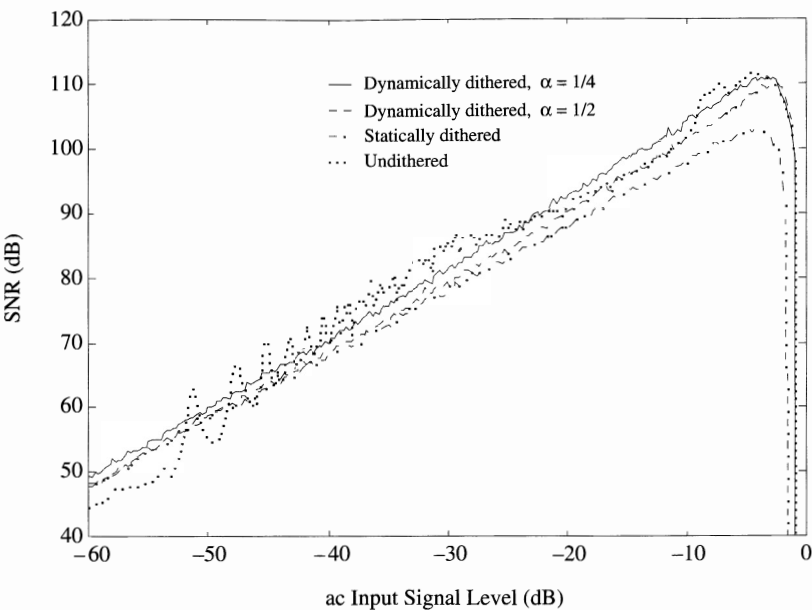


Figure 3.53 SNR vs. ac input signal level for dynamically dithered, statically dithered, and undithered third-order modulator.

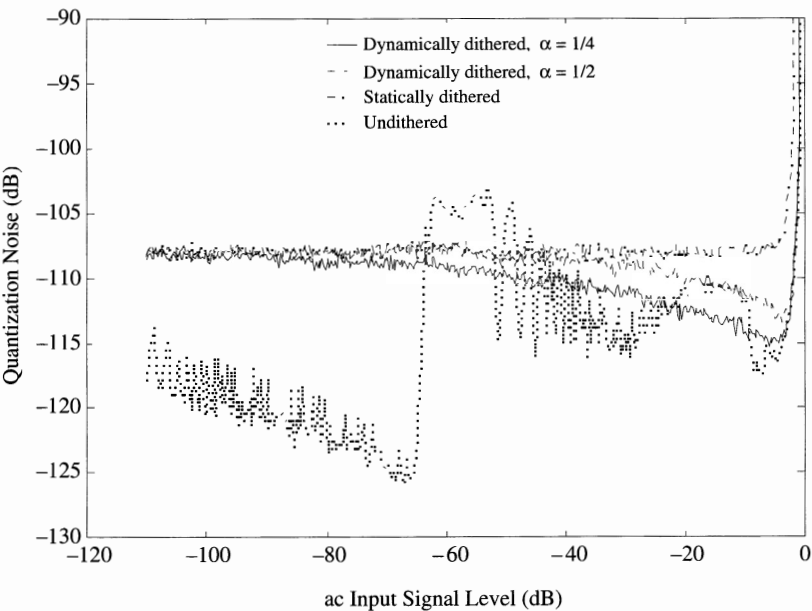


Figure 3.54 Quantization noise vs. ac input signal level for dynamically dithered, statically dithered, and undithered third-order modulator.

3.13.2 Implementation Considerations of Dynamic Dither

Both A/D and D/A implementations of a dynamically dithered modulator need to be considered. For both cases, there is a fundamental problem that must first be addressed. The dynamic dither function expressed in Eq. (3.20) is a continuous-amplitude expression. Since we are assuming that the dither source itself is digital and discrete in amplitude, this implies that the input magnitude $|x(n)|$ needs to effectively be discrete in amplitude. Of course, this is the case for the D/A implementation, assuming the input is represented in finite word lengths of, say, 16–24 bits. Simulation shows, however, that $|x(n)|$ only needs to be quantized into a few discrete levels. In other words, only a few multiplicands need to be applied to the dither $d(n)$, based on the range of values of $|x(n)|$, in order for the dynamic dithering scheme to be fully successful.

Figure 3.55 illustrates a general block diagram of a dynamically dithered modulator. The signal input $|x(n)|$ is input to a magnitude block, whose output is quantized nonlinearly to produce one of a few discrete levels that are used to multiply the dither.

For the D/A modulator implementation, it is desirable to eliminate the need for a hardware multiplier in order to keep the complexity minimal. Therefore, it is preferable to quantize the multiplicands into simple binary-weighted factors. The first step in such a practical realization is to find the values of $|x(n)|$ that result in factors of 2 decreases from the function $(1 - |x(n)|^\alpha)$, beginning with $x(n) = 0$. A simple example is shown in Table 3.1. The example illustrates a 2-bit quantization of $(1 - |x(n)|^\alpha)$. This 2-bit output

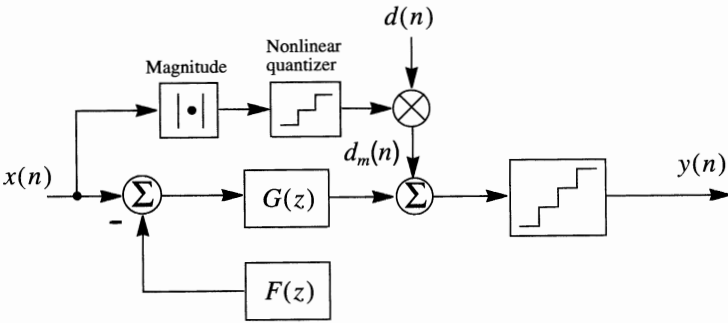


Figure 3.55 General $\Delta\Sigma$ modulator with *dynamic* noise-shaped dither.

TABLE 3.1 EXAMPLE OF THRESHOLDS AND QUANTIZED VALUES FOR DYNAMIC DITHER LEVEL ADJUSTMENT

Level adjustment of dither	Threshold level of $ x(n) $ (dB)	
	Based on $(1 - x(n) ^{1/4})$	Optimized based on 6-dB multiples
1	$-\infty$	$-\infty$
$\frac{1}{2}$	-24.08	-36
$\frac{1}{4}$	-10.00	-18
$\frac{1}{8}$	-4.64	-6

is then used for selecting arithmetically shifted versions of the dither words $d(n)$ to produce divisions of $d(n)$ by factors of 2. In a practical implementation, this can be accomplished with a small number of logic gates per dither bit.

It can be seen in this example that the first threshold occurs when the input is about -24 dB. From the previous simulations shown in Figures 3.53 and 3.54, it can be seen that a small amount of dynamic range is already lost at -24 dB, therefore it may be desirable to move this threshold back to a smaller value, say -30 to -40 dB, in order to start the dynamic range recovery at a lower input level. Caution must be observed, however, not to move this first threshold too low, since it is not desirable for the modulator to behave in a less dithered manner for low-input signal levels. Wherever the final choice of the first threshold resides, the lower portion of the dynamic range (below that point) will be identical to the statically dithered result. Optimizing the higher threshold values can be performed with the objective of making the SNR curve follow the desired curve resulting from continuous unquantized dither level adjustment at higher input signal level. For this example, the desired SNR curve for the optimization is the one shown by the solid line in Figure 3.53 for $\alpha = \frac{1}{4}$. The values in the center column of Table 3.1 only represent starting points for optimization. Simulations revealed that the thresholds generally must be lower than these initial starting points in order to produce the desired results. After a few iterations of experimentation with the thresholds, it was determined that the difference between the SNR curves was only about ± 1 dB at any point along the curve for the upper 20 dB of dynamic range, thereby accomplishing the desired result. The final thresholds from the experiment are shown in the right-most column of Table 3.1.

The tolerance of the threshold values is not at all critical. Virtually the same performance can be achieved by moving the thresholds many decibels in either direction from these values.

For an A/D modulator case, there are two basic methods of implementation. The first case is shown in Figure 3.56, wherein an estimate of the input magnitude is obtained digitally by processing the output of the first-stage decimation filter. Recall that the objective of dynamic dither is to prevent instantaneous overloading of the quantizer for large inputs. Therefore, one concern will be the phase delay through the decimation filter, which could potentially cause a misalignment. This can be minimized by taking the output from the first stage of decimation, since the group delay of this filter will typically be small relative to the more stringent filtering performed in the latter stages of decimation that follow. In addition, since the input signal is heavily oversampled, it is moving relatively slowly with respect to the sample clock, which will further alleviate the cause for concern over phase misalignment through the filter.

The second approach to A/D implementation is one involving processing the analog input signal directly. While the parameters are not as easily controlled as in the first all-digital case, the benefits are that there is no need to constrain the choices of dither multiplicands to binary-weighted values, since analog-related parameters will be involved. The values of the desired thresholds, which are determined by simulation, need not be compromised as much, even though there is a very wide tolerance to the sensitivities of the values.

Additional simplifications can also be made to minimize the number of level adjustments. While four levels are shown in the example, simulations reveal that as few as three or possibly two levels are possible with beneficial results. This result can be combined

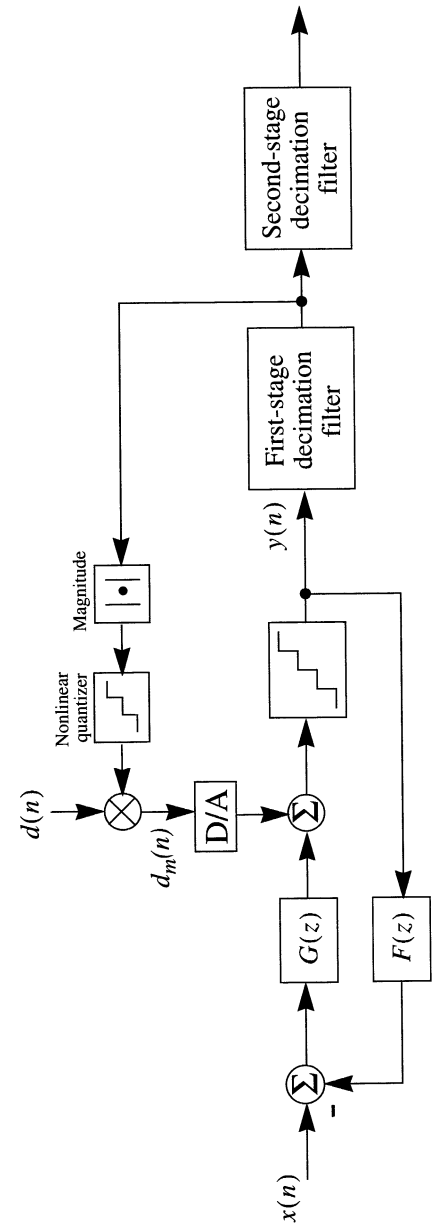


Figure 3.56 A $\Delta\Sigma$ A/D converter with dynamic dither, having input signal magnitude estimate taken digitally at the output of the first-stage decimation filter.

with the fact that only a few bits of quantized dither are actually needed for $d(n)$, as mentioned in Section 3.10. When all these parameters are taken into consideration, many degrees of freedom and flexibility are available for the designer to make an effective realization of dynamic dither with little additional hardware.

3.14 DITHERED MULTIBIT NOISE-SHAPING CODERS

Single-bit modulators are popular primarily because of the following simple principle: *Two points define a straight line*. Hence, they are inherently linear. However, there are many problems associated with them: They are inherently unstable for orders greater than 2; their out-of-band noise density is relatively high, which is exacerbated at out-of-band frequencies through aggressive noise shaping; they produce very large tones near $f_s/2$; there are problems implementing the out-of-band filters that must follow them; they require relatively high oversampling rates, which may cause high power dissipation. An alternative is multibit noise-shaped coding. Various schemes have been reported to alleviate the strict matching requirements within the multibit D/A converter [49, 71, 76–78]. As a result of bringing multibit capability into the design process, many parameters may be traded off and relaxed.

Multibit noise-shaping coders are also popular for digital requantization, as previously stated. In these applications, the signal is only processed in the digital domain using an inherently precise digital quantizer.

While the combination of higher order noise shaping and multibit quantization makes it easier to decorrelate the quantization noise, there is still reason to be concerned about this issue [15], especially when rational digital arithmetic is involved. The next section sets forth an analytical relationship between the dither, the coder stability, and other parameters that affect the performance of such a system.

3.14.1 Stability Test with Dither

An analysis of the general stability of multibit noise-shaping coders and modulators with dither is given in [10]. This stability test, based on the $\mathcal{L}_1$ norm of the impulse response of the filter [79, 80], makes the assumption that the quantizer must operate in the no-overload region in order to remain stable. For quantizers having a small number of bits, this assumption is sometimes too conservative, since many $\Delta\Sigma$ modulator architectures operate with their internal quantizers outside of the no-overload region. Nevertheless, this type of stability analysis is still useful for examining the issues and trade-offs.

The signal variables in the following analysis apply to Figure 3.35. As before, the quantization error occupies one quantization step interval Δ . If the dither occupies a range of δ , then the relative peak dither amplitude is δ/Δ . Therefore, $|e(n)| \leq \Delta/2$ and $|d(n)| = (\delta/\Delta)(\Delta/2)$, which leads to

$$|u(n)| \leq |x(n)| + \left| \sum_{k=1}^{\infty} h(k)e(n-k) \right| + \left| \sum_{k=1}^{\infty} h(k) d(n-k) \right|$$

$$\begin{aligned} |u(n)| &\leq \|x\|_{\infty} + \frac{\Delta}{2} \sum_{k=1}^{\infty} |h(k)| + \left(\frac{\delta}{\Delta}\right) \frac{\Delta}{2} \sum_{k=1}^{\infty} |h(k)| \\ |u(n)| &\leq \|x\|_{\infty} + \frac{\Delta}{2} \left(1 + \frac{\delta}{\Delta}\right) (\|h\|_1 - 1) \end{aligned} \quad (3.23)$$

where $\|h\|_1$ is the $\mathcal{L}_1$ norm of the scaled impulse given by

$$\|h\|_1 \equiv \sum_{k=1}^{\infty} |h(k)| \quad (3.24)$$

and $\|x\|_{\infty}$ is the $\mathcal{L}_{\infty}$ norm of the input, which is simply the maximum peak value

$$\|x\|_{\infty} \equiv |x|_{\max}$$

For an L -level quantizer that is never overloaded, Eq. (3.23) becomes

$$\frac{L}{L-1} \geq \|x\|_{\infty} + \frac{1}{L-1} \left(1 + \frac{\delta}{\Delta}\right) (\|h\|_1 - 1)$$

so that

$$\|h\|_1 \leq 1 + \frac{L - (L-1)\|x\|_{\infty}}{1 + \delta/\Delta} \quad (3.25)$$

Rearranging Eq. (3.25) for the convenience of specifying L ,

$$L \geq \frac{(1 + \delta/\Delta)(\|h\|_1 - 1) - \|x\|_{\infty}}{1 - \|x\|_{\infty}} \quad (3.26)$$

Therefore, if the quantizer is to remain within the no-overload region, the quantizer must have enough dynamic range (steps) to *contain* a simultaneous occurrence of both the largest possible output value of the filter *plus* the largest input sample. Otherwise, the quantizer will overload, followed by potential loop instability.

From Eq. (3.26), the penalty for added dither can readily be seen, but this penalty is actually rather modest. If the relative peak dither δ/Δ is unity, the resulting value of L is exactly twice that of the undithered case, requiring one more bit of dynamic range in the quantizer.

The value of L can be minimized by choosing a minimum phase noise-shaping filter characteristic for $H(z)$ [10, 11, 13, 14].

3.15 CHAOS VERSUS NOISE-SHAPED DITHER

Chaos (theory and implementation) has been proposed as a means of destabilizing idle tones in $\Delta\Sigma$ modulators [80–82]. A $\Delta\Sigma$ modulator design can be converted into a chaotic one by moving one or more noise transfer function zeros outside the unit circle. It has been suggested as an alternative to dither [81].

Consider the following example taken from [10] and explained with additional detail here. An 11-tap optimal minimum phase finite impulse response (FIR) noise-shaping filter [11, 83] was designed to operate at an oversampling ratio of 128. The transfer function of the noise-shaping filter is given by

$$H(z) = 1 - 1.66594029170833z^{-1} + 0.02233756298529z^{-2} + 0.47360174745650z^{-3} \\ + 0.31021368876584z^{-4} + 0.04808564352797z^{-5} - 0.07764587017800z^{-6} \\ - 0.07378416822311z^{-7} - 0.03128167052321z^{-8} - 0.00570431159920z^{-9} \\ + 0.00011767z^{-10}$$

The zeros are therefore located at

$$\begin{aligned} z_{1,2} &= 0.99976434891650 \pm j0.02170828201245 \\ z_{3,4} &= -0.03280199398754 \pm j0.48068734966016 \\ z_{5,6} &= -0.26546543889787 \pm j0.32291270561850 \\ z_{7,8} &= -0.37784570618261 \pm j0.11345716538027 \\ z_9 &= 0.01863787201124 \\ z_{10} &= 1 \end{aligned}$$

The $\mathcal{L}_1$ norm of this impulse response is 3.7. This indicates that about three levels of quantization are needed for stability. If dither is added, one extra bit is added to the quantizer so that six levels are used.

In order to form the chaotic modulator, one or more zeros need to be reciprocated. Various experiments resulted in the choice of the pair $z_{5,6}$ to be reciprocated. The reciprocated zero is therefore located at $z_{5,6} = -1.51916321582391 \pm j1.84791325881984$. All of the other zeros were left in their original positions.

The resulting transfer function is therefore given by

$$H(z) = 1 + 0.84145526214375z^{-1} + 0.06180833392046z^{-2} - 9.17196212037102z^{-3} \\ + 6.73587746638514z^{-4} + 1.72733326784214z^{-5} - 0.25045684218615z^{-6} \\ - 0.20338756948562z^{-7} - 0.54805074384321z^{-8} - 0.19647056706729z^{-9} \\ + 0.00385351266180z^{-10}$$

The $\mathcal{L}_1$ norm of this impulse response is 20.7. This indicates that about 21 levels are needed in the quantizer to keep the coder stable. This is approximately 2 additional bits over the dithered minimum phase design. The magnitude response of the noise transfer function remains proportional to the previous one.

Figure 3.57 shows the spectrographs of the simulations. These are the result of sweeping up the dc input level, that is, holding the input level constant for 64K samples, performing an FFT, and restarting the system with a newly increased dc input level. Each sweep plots the magnitude squared of the quantization error versus frequency. Figure 3.57(a) shows the 3-level minimum phase coder without dither. The tones clearly track the increasing input level as it is being swept up. Figure 3.57(b) shows the same coder but with 6-level quantization and with dither. This clearly results in an output that has no tonal

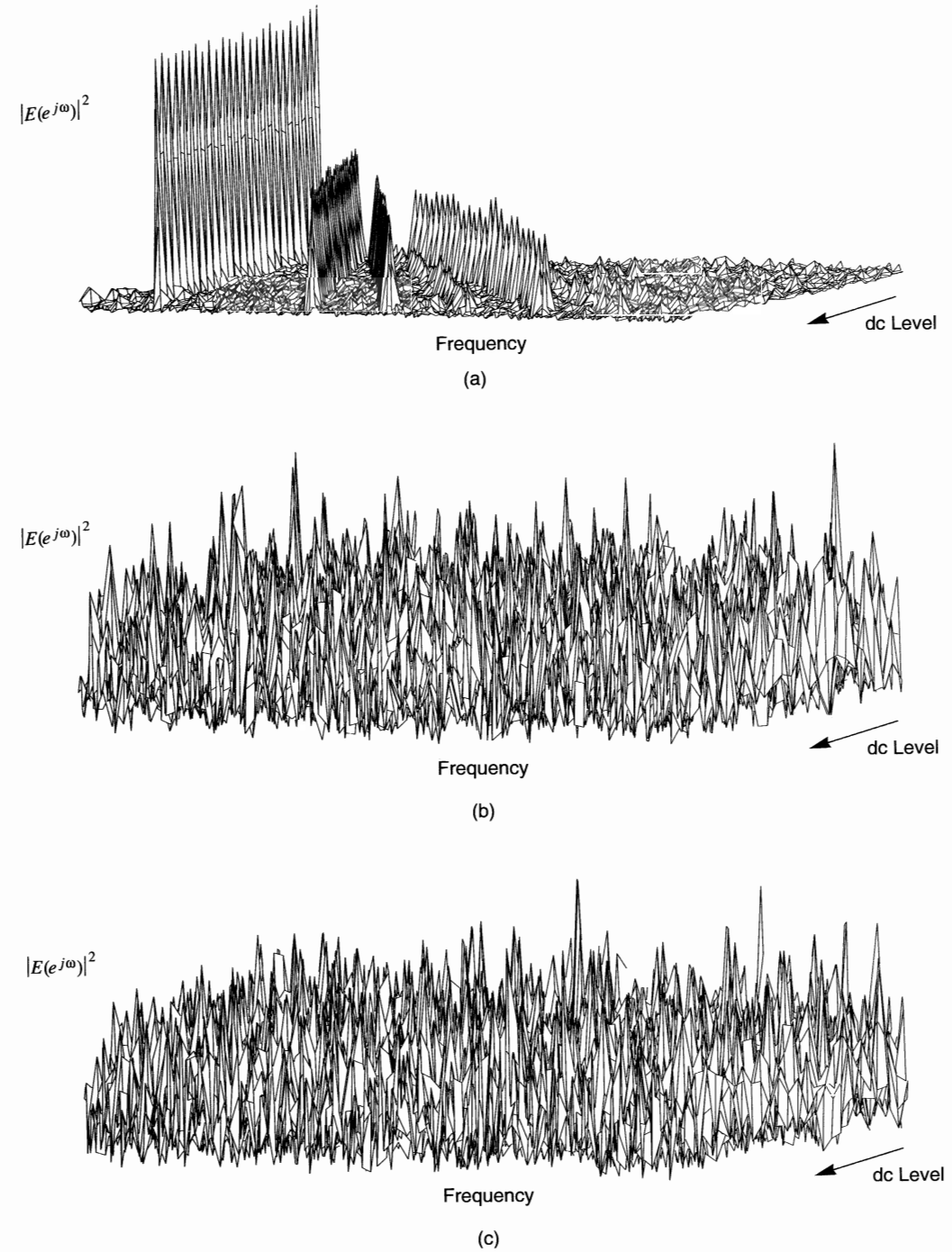


Figure 3.57 Optimal minimum phase FIR noise-shaping coder spectrographs [(a)–(b)] of quantization error for dc input sweep: (a) undithered 3-level quantizer; (b) dithered 6-level quantizer; (c) mixed-phase (chaotic) FIR noise-shaping coder with 21-level undithered quantizer. The noise-shaping filter has the same relative magnitude characteristic in each.

correlation to the input level. The coder was also simulated with 6-level quantization without dither (not shown). The strongly tonal behavior was nearly identical to the undithered 3-level coder. Figure 3.57(c) shows the chaotic modulator with 21-level quantization and no dither. As with the dithered 6-level minimum phase design, there is clearly no tonal correlation to the input. The average baseband noise floors for the two cases are similar: -127 dB for (b) versus -132 dB for (c). Attempts to reduce the number of quantization levels in the chaotic modulator below 21 resulted in instability.

Another form of comparison entails increasing the number of quantization levels in the dithered minimum phase coder to 21, so that the two designs are equivalent in terms of complexity.¹ This results in a baseband noise floor of -146 dB for the dithered minimum phase modulator design, which is a significantly better result than that produced by the chaotic modulator.

Other examples were tried as well. In these types of simulation studies, it is always difficult to create an example that fairly compares two concepts with one another. Throughout these studies, the criteria for a *fair* comparison was defined as keeping the magnitude response of the noise transfer function proportionally identical in each case, but letting the number of quantization levels vary for reasons of stability, as shown in the previous example. Each example indicated that a dithered minimum phase coder always produces a more efficient result, that is, it will outperform a chaotic one for a given noise transfer function characteristic and a given number of quantization levels in the coder.

3.16 OTHER TECHNIQUES

In addition to the chaos method discussed in the previous section, other methods for tone decorrelation in $\Delta\Sigma$ modulators include: (1) adding an out-of-band sine or square wave [72, 84, 85]; (2) adding a dc offset to the input of the modulator; (3) adding a small amount of random noise to the input [36, 37]; (4) using analog sources of noise within the modulator as a dither [36, 86].

An out-of-band sine or square wave dither needs to be filtered out. It is typically common in practice to use a frequency that is an exact submultiple of the oversampling clock, which makes the generation of the signal relatively easy. In addition, the frequency is typically chosen to fall on a null of the sinc^K decimation filter for A/D implementation. Unfortunately, the ideal frequency of such a dither is at some irrational frequency with respect to the sampling frequency of the modulator [50]. This makes both the generation and the filtering of the dither more complicated. In addition, the square wave (or other periodic signal) dither needs to be relatively large to be fully effective, which reduces the dynamic range of the modulator.

A more common technique entails adding a dc offset. This pushes the idle channel noise spectra to higher frequencies outside the baseband [3, 33, 34]. However, this complicates the design if dc conversion is required, but this is not a problem for audio convert-

1. It is implied that the complexity of a dither generator is trivial since it requires a small number of gates of logic, while the complexity of a D/A converter with two additional bits is substantially more complex than a few digital gates, and also less robust. Under these circumstances, the comparison of the two systems is approximately *fair*.

ers. However, it is not a reliable technique because the idle channel tone frequencies are also a function of the initial conditions of the integrators. In addition, for the case of A/D converters, variations in analog components tend to generate unpredictable internal dc offsets of their own. Furthermore, there is no guarantee that a single dc value will move all tones out of the passband. It is also possible that the dc offset will pull tones outside the passband down into the passband. In addition, the dc offset reduces the dynamic range of the modulator.

Another technique previously discussed entails adding a small amount of white noise to the input and/or assuming that the small inherent thermal noises in an analog implementation will suffice as an appropriate dither. As previously discussed, this is generally ineffective except for some specific architectures, such as ideally matched multistage modulators or possibly some multibit high-order modulators.

In another reference [86], the noise floor of an analog modulator is intentionally degraded at the quantizer input. This technique has the potential of producing a result similar to that of noise-shaped dither, but only if the noise magnitude is sufficient and predictable in the actual hardware implementation. As discussed previously in the chapter, there are magnitude levels below which the added dither will not be very effective at decorrelating tones, and this concept also applies here.

3.17 CONCLUSION

The tonal nature of the quantization error from $\Delta\Sigma$ modulation is somewhat elusive because it is not easily observable in long-term power spectral estimates. However, relatively short-term autocorrelation estimates (or their corresponding PSD estimates), as well as time-domain analysis, were shown to be more useful for examining the nature of the quantization error. Tones are observable both on computer simulations and commercial $\Delta\Sigma$ modulators, for dc and ac inputs. Close correlation was observed between the simulations and actual hardware performance. These tones may be detectable by a human listener, and may have an annoying quality. In the time domain, the quantization error is manifested at the converter output as an impulse train pattern typically occupying as many as 3 bits of baseband dynamic range. This impulse train can obscure a real signal occupying these lower bits.

The purpose of dithering is to effectively decorrelate and whiten the quantization error. Various dithering methods and their associated penalties were discussed. The main method proposed in the chapter entails applying a sufficient amount of noise-shaped dither such that the dither transfer function is the same as the quantization noise transfer function of the $\Delta\Sigma$ modulator, thereby minimizing the amount of additional noise power in the baseband. This can be accomplished with a general improvement in baseband noise power for low signal levels. In addition, the dither significantly reduces very strong tones near $f_s/2$, which can fold down into the baseband and cause additional audible tones. For rpdf dither, the amplitude of the dither needs to span as much as half the quantizer interval. This comes with a penalty of reduced dynamic range of several decibels (depending on the order of the modulator) for large inputs.

Dynamic dithering was introduced as a method of recovering all the lost dynamic range for high-input signal levels due to dithering. The dynamic dithering technique

entails modulating the dither with the input signal in a manner that still produces white uncorrelated noise. It was shown that this technique can be simplified for practical implementation, resulting in minimal hardware.

REFERENCES

- [1] N. He, F. Kuhlmann, and A. Buzo, "Multiloop sigma delta quantization," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1015–1028, May 1992.
- [2] I. Galton, "Granular quantization noise in a class of delta-sigma modulators," *IEEE Trans. Inform. Theory*, vol. 40, no. 3, pp. 848–859, May 1994.
- [3] R. M. Gray, "Quantization noise spectra," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 1220–1244, Nov. 1990.
- [4] B. Widrow, "A study of rough amplitude quantization by means of Nyquist sampling theory," *IRE Trans. Circuit Theory*, vol. CT-3, pp. 266–276, 1956.
- [5] A. B. Sripad and D. L. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Trans. Acoust. Speech Sig. Proc.*, vol. ASSP-25, pp. 442–448, Oct. 1977.
- [6] C. C. Cutler, "Transmission systems employing quantization," U.S. Patent No. 2,927,962, filed 1954, issued 1960.
- [7] P. J. Naus et al., "A CMOS stereo 16-bit D/A converter for digital audio," *IEEE J. Solid-State Circuits*, vol. 22, pp. 390–395, June 1987.
- [8] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, March 1985.
- [9] S. K. Tewksbury and R. W. Hallock, "Oversampled, linear predictive and noise-shaping coders of order $N > 1$," *IEEE Trans. Circuits Sys.*, vol. 25, pp. 436–447, July 1978.
- [10] S. R. Norsworthy, "Optimal nonrecursive noise shaping filters for oversampling data converters, part 1: theory," *IEEE Proc. ISCAS'93*, vol. 2, pp. 1353–1356, May 1993.
- [11] S. R. Norsworthy, "Optimal nonrecursive noise shaping filters for oversampling data converters, part 2: applications," *IEEE Proc. ISCAS'93*, vol. 2, pp. 1357–1360, May 1993.
- [12] R. Schreier and Y. Yang, "Stability tests for single-bit sigma-delta modulators with second-order FIR noise transfer functions," *IEEE Proc. ISCAS'92*, vol. 3, pp. 1316–1319, May 1992.
- [13] M. A. Gerzon and P. G. Craven, "Optimal noise shaping and dither of digital signals," presented at the 87th Convention of the Audio Engineering Society, vol. 37, p. 1072, May 1989, Audio Engineering Society, preprint 2822.
- [14] R. A. Wannamaker, "Psychoacoustically optimal noise shaping," *J. Audio Eng. Soc.*, vol. 40, pp. 611–620, July/Aug. 1992.
- [15] J. A. Moorer and J. C. Wen, "Whither dither: experience with high-order dithering algorithms in the studio," presented at the 95th Convention of the Audio Engineering Society, New York, Oct. 1993, Audio Engineering Society, preprint 3747.
- [16] V. Friedman, et al., "A dual-channel voice-band PCM code using $\Sigma\Delta$ modulation technique," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 274–280, April 1989.
- [17] S. R. Norsworthy, I. G. Post, and H. S. Fetterman, "A 14-bit 80-kHz sigma-delta A/D converter: modeling, design, and performance evaluation," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 256–266, April 1989.

- [18] D. R. Welland, et al., "A stereo 16-bit delta-sigma A/D converter for digital audio," *J. Audio Eng. Soc.*, vol. 37, pp. 476–4865, June 1989.
- [19] N. S. Sooch and J. W. Scott, "18-bit stereo D/A converter with integrated digital and analog filters," presented at the 91st Convention of the Audio Engineering Society, New York, Oct. 1991, Audio Engineering Society, preprint 3113.
- [20] S. R. Green, S. Harris, and B. Wilson, "An 18-bit delta-sigma D/A processor achieving full-scale THD+N > 100 dB," presented at the 93rd Convention of the Audio Engineering Society, San Francisco, Oct. 1992, Audio Engineering Society, preprint 3416.
- [21] P. Ferguson, et al., "An 18-bit 20 kHz dual sigma-delta A/D converter," *ISSCC Dig. Tech. Pap.*, pp. 68–69, Feb. 1991.
- [22] K. C.-H. Chao, D. Nadeem, W. L. Lee, and C. G. Sodini, "A higher order topology for interpolative modulators for oversampling A/D converters," *IEEE Trans. Circuits Sys.*, vol. CAS-37, pp. 309–318, March 1990.
- [23] T. Ritiemi, T. Karema, and H. Tenhunen, "Design of stable high-order 1-bit sigma-delta modulators," *IEEE Proc. ISCAS'90*, pp. 3267–3270, May 1990.
- [24] Y. Matsuya, et al., "A 17-bit oversampling D-to-A conversion technology using multistage noise shaping," *IEEE J. Solid-State Circuits*, vol. 24, pp. 969–975, Aug. 1989.
- [25] G. Yen, S. Stubbe, and W. Sansen, "16-bit 320 kHz CMOS A/D converter using two-stage third-order $\Sigma\Delta$ noise shaping," *IEEE J. Solid-State Circuits*, vol. 28, pp. 640–647, June 1993.
- [26] L. A. Williams, III and B. A. Wooley, "Third-order cascaded sigma-delta modulators," *IEEE Trans. Circuits Sys.*, vol. CAS-38, pp. 489–498, May 1991.
- [27] Lectures and discussions at a course entitled, "Oversampled sigma-delta modulators and data converters," held at the University of California at Los Angeles, March 1991, and again at Portland, OR, in June 1992.
- [28] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, Kluwer Academic Publishers, Norwell, MA, 1988.
- [29] S. R. Norsworthy, "Effective dithering of sigma-delta modulators," *IEEE Proc. ISCAS'92*, vol. 3, pp. 1304–1307, May 1992.
- [30] S. R. Norsworthy and D. A. Rich, "Idle channel tones and dithering in delta-sigma modulators," presented at the 95th Convention of the Audio Engineering Society, New York, Oct. 1993, Audio Engineering Society, preprint 3711.
- [31] A. V. Oppenheim and R. W. Shafer, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- [32] C. M. Rader, "An improved algorithm for high-speed autocorrelation with applications to spectral estimation," *IEEE Trans. Audio Electroacoust.*, vol. AU-18, pp. 439–441, Dec. 1970.
- [33] J. C. Candy and O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-29, pp. 1316–1323, Sept. 1981.
- [34] V. Friedman, "Structure of the limit cycles in sigma-delta modulation," *IEEE Trans. Commun.*, vol. 36, pp. 972–979, Aug. 1988.
- [35] General discussions at ISCAS 1990 conference short course taught by S. R. Norsworthy, "Oversampled $\Sigma-\Delta$ Data Converters," New Orleans, LA, April 1990.
- [36] P. J. A. Naus and E. C. Dijkmans, "Low signal-level distortion in sigma-delta modulators," presented at 84th Convention of the Audio Engineering Society, Paris, March 1988, Audio Engineering Society, preprint 2584 (D-4).

- [37] I. Galton, "One-bit dithering in delta-sigma modulator-based D/A conversion," *IEEE Proc. ISCAS'93*, pp. 1310–1313, May 1993.
- [38] I. Galton, private correspondence.
- [39] B. Kup, et al., "A bitstream digital-to-analog converter with 18b resolution," *ISSCC Dig. Tech. Pap.*, pp. 70–71, Feb. 1991.
- [40] W. Bradinal, "Audio performance at low signal levels in digital audio systems," application note description of Philips SAA7350 D/A converter in *Philips Bitstream Conversion*, edited by S. Nethisinghe, Jan. 1991.
- [41] Philips SAA7350 D/A Converter Data Sheet.
- [42] R. W. Adams, P. F. Ferguson, Jr., and A. Ganesan, "Design of single-bit noise-shaping loops with high-order loop filters," presented at the 89th Convention of the Audio Engineering Society, Los Angeles, Sept. 1990, Audio Engineering Society, preprint 2974.
- [43] *Signal Processing Toolbox for Use with MATLAB*, MathWorks, Natick, MA.
- [44] J. G. Kenney and L. R. Carley, "CLANS: A high-level synthesis tool for high resolution data converters," *IEEE Int. Conf. Computer-Aided Design*, Nov. 1988.
- [45] R. Schreier, "An empirical study of high-order single-bit delta-sigma modulators," *IEEE Trans. Circuits Sys.*, vol. 40, no. 8, pp. 461–466, Aug. 1993.
- [46] S. Harris, "How to achieve optimum performance from delta-sigma A/D and D/A converters," presented at the 93rd Convention of the Audio Engineering Society, San Francisco, Oct. 1992, Audio Engineering Society, preprint 3417.
- [47] J. L. LaMay and H. T. Bogard, "How to obtain maximum practical performance from state-of-the-art delta-sigma analog-to-digital converters," *IEEE Trans. Instr. Meas.*, vol. 41, no. 6, pp. 861–867, Dec. 1992.
- [48] Comments by Robert Adams.
- [49] J. W. Fattarusio, et al., "Self-calibration techniques for a second-order multibit sigma-delta modulator," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, Dec. 1993.
- [50] W. Chou and R. M. Gray, "Dithering and its effects on sigma-delta and multi-stage sigma-delta modulation," *IEEE Proc. ISCAS'90*, pp. 368–371, May 1990.
- [51] E. Stikvoort, "Higher-order one bit coder for audio applications," presented at 84th Convention of the Audio Engineering Society, Paris, March 1988, Audio Engineering Society, preprint 2583 (D-3).
- [52] L. G. Roberts, "Picture coding using pseudo-random noise" *IRE Trans. Inform. Theory*, vol. IT-8, no. 2, pp. 145–154, Feb. 1962.
- [53] L. Schuchman, "Dither signals and their effect on quantization noise," *IEEE Trans. Commun. Tech.*, vol. COM-12, pp. 162–165, Dec. 1964.
- [54] J. O. Limb, "Design of dither waveforms for quantized visual signals," *Bell Syst. Tech. J.*, vol. 48, no. 7, pp. 2555–2584, Sept. 1969.
- [55] N. S. Jayant and L. R. Rabiner, "Application of dither to the quantization of speech signals," *Bell Syst. Tech. J.*, vol. 51, no. 6, pp. 1293–1304, Jul.–Aug. 1972.
- [56] L. R. Rabiner and J. A. Johnson, "Perceptual evaluation of the effects of dither on low bit rate PCM systems," *Bell Syst. Tech. J.*, vol. 51, no. 7, pp. 1487–1494, Sept. 1972.
- [57] J. Vanderkooy and S. P. Lipshitz, "Resolution below the least significant bit in digital systems with dither," *J. Audio Eng. Soc.*, vol. 32, no. 3, pp. 106–113, March 1984.
- [58] J. Vanderkooy and S. P. Lipshitz, "Dither in digital audio," *J. Audio Eng. Soc.*, vol. 35, No. 12, pp. 966–975, Dec. 1987.
- [59] S. P. Lipshitz, R. A. Wannamaker, and J. Vanderkooy, "Quantization and dither: a theoretical survey," *J. Audio Eng. Soc.*, vol. 40, no. 5, pp. 355–375, May 1992.
- [60] R. A. Wannamaker and S. P. Lipshitz, "Time domain behavior of dithered quantizers," presented at the 93rd Convention of the Audio Engineering Society, San Francisco, Oct. 1992, Audio Engineering Society, preprint 3418.
- [61] R. M. Gray and T. G. Stockham, Jr., "Dithered quantizers," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 805–812, May 1993.
- [62] L. R. Carley, "An oversampling analog-to-digital converter topology for high-resolution signal acquisition systems," *IEEE Trans. Circuits Sys.*, vol. CAS-34, pp. 83–90, Jan. 1987.
- [63] T. Araki and M. Kubo, "Wide dynamic range analog to digital conversion method and system," U.S. Patent 4,751,496, assigned to Teac Corporation, June 14, 1988.
- [64] O. Takabayashi, "Dither circuit having dither level changing function," U.S. Patent 4,857,927, assigned to Yamaha Corporation, Aug. 15, 1989.
- [65] P. G. Craven and M. A. Gerzon, "Compatible improvement of 16-bit systems using subtractive dither," presented at the 93rd Convention of the Audio Engineering Society, San Francisco, Oct. 1992, Audio Engineering Society, preprint 3356.
- [66] W. Chou, "Sigma-delta and multi-stage sigma-delta modulation with inside loop dithering," *IEEE Proc. ICASSP'91*, pp. 1953–1956, April 1991.
- [67] C. Dunn and M. Sandler, "A simulated comparison of dithered and chaotic sigma-delta modulators," presented at the 97th Convention of the Audio Engineering Society, San Francisco, Nov. 1994, Audio Engineering Society, preprint 3926.
- [68] AT&T Microelectronics DSP16C Digital Signal Processor / Codec Data Sheet.
- [69] AT&T Microelectronics CSP1027 Codec Data Sheet.
- [70] S. Golomb, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, 1982.
- [71] M. Sarhang-Nejad and G. Temes, "A high-resolution multibit $\Sigma\Delta$ ADC with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, pp. 648–660, June 1993.
- [72] Y. Matsuya, et al., "A 16-bit oversampling A/D conversion technology using triple-integration noise shaping," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 921–929, Dec. 1987.
- [73] M. Ueki, N. Yasuda, and T. Masuda, "Digital to analog converter with dither using two parallel paths," U.S. Patent 5,073,778, assigned to Sony Corporation, Dec. 17, 1991.
- [74] S. R. Norsworthy, "Dynamic dithering of delta-sigma modulators," presented at the 99th Convention of the Audio Engineering Society, New York, Oct. 1995, Audio Engineering Society, preprint 4103.
- [75] N. Jayant, J. Johnston, and R. Safranek, "Signal compression based on models of human perception," *Proc. IEEE*, vol. 81, no. 10, Oct. 1993.
- [76] R. T. Baird and T. S. Fiez, "Improved $\Delta\Sigma$ DAC linearity using data weighted averaging," *Proc. IEEE Int. Symp. Circuits Sys.* '95, pp. 13–16, May 1995.
- [77] R. Schreier and B. Zhang, "Noise-shaped multi-bit D/A convertor employing unit elements," *Electron. Lett.*
- [78] R. W. Adams and T. W. Kwan, "Data-directed scrambler for multi-bit noise shaping D/A converters," U.S. Patent 5,404,142, April 4, 1995.
- [79] D. Anastassiou, "Error diffusion coding for A/D conversion," *IEEE Trans. Circuits Sys.*, vol. CAS-36, pp. 1175–1186, Sept. 1989.

- [80] R. Schreier, "Noise shaped coding," Ph.D. Dissertation, University of Toronto, 1991.
- [81] R. Schreier, "Destabilizing limit cycles in delta-sigma modulators," *IEEE Proc. ISCAS'93*, vol. 2, pp. 1369–1372, May 1993.
- [82] M. Motamed, A. Zakhor, and S. Sanders, "Tones, saturation, and SNR in double loop $\Sigma\Delta$ modulators," *IEEE Proc. ISCAS'93*, vol. 2, pp. 1345–1348, May 1993.
- [83] T. W. Parks and C. S. Burrus, *Digital Filter Design*, John Wiley & Sons, New York, 1987.
- [84] B. Leung, R. Neff, P. Gray, and R. Brodersen, "Area-efficient multichannel oversampled PCM voice-band coder," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1351–1357, Dec. 1988.
- [85] T. Okamoto, et al., "A 16b oversampling CODEC with filtering DSP," *ISSCC Dig. Tech. Pap.*, pp. 74–75, Feb. 1991.
- [86] D. Welland, "Method for tone avoidance in delta-sigma converters," U.S. Patent 5,055,846, assigned to Crystal Semiconductor, Oct. 8, 1991.

Robert W. Adams
Richard Schreier

Stability Theory for $\Delta\Sigma$ Modulators

4.1 INTRODUCTION

In Chapters 2 and 3, fundamental ideas regarding the nature of quantization noise in $\Delta\Sigma$ modulators were discussed. This chapter likewise deals with a fundamental and often misunderstood aspect of $\Delta\Sigma$ modulator operation: stability. Like the white-noise assumption, the assumption of modulator stability is an unquestioned premise in many discussions of $\Delta\Sigma$ modulation.

High-order $\Delta\Sigma$ modulators have been successfully designed and marketed [1, 2] despite the absence of a rigorous theory for their stability. The driving force behind this seemingly reckless act is a powerful one: As Figure 4.1 shows, noise transfer functions (NTFs) of the form

$$H(z) = (1 - z^{-1})^n \quad (4.1)$$

exhibit exceedingly high SNR when n is large, even at low oversampling ratios. This level of performance is not achievable in practice because a modulator with an NTF of the form given by Eq. (4.1) is not stable with a binary quantizer for orders greater than 2. By "not stable" we mean that the modulator exhibits large, although not necessarily unbounded, states and a poor SNR compared with that predicted by linear models. A further characteristic of an unstable modulator is that it generally has an *oscillation frequency* that is rather low, producing an output of alternating long strings of 1's and 0's.

This chapter sheds light on parts of the stability problem but does not provide a complete solution to it. The discussion focuses on single-loop, discrete-time, and generally single-bit designs. Multiloop (cascade) designs can be handled by simply ignoring the

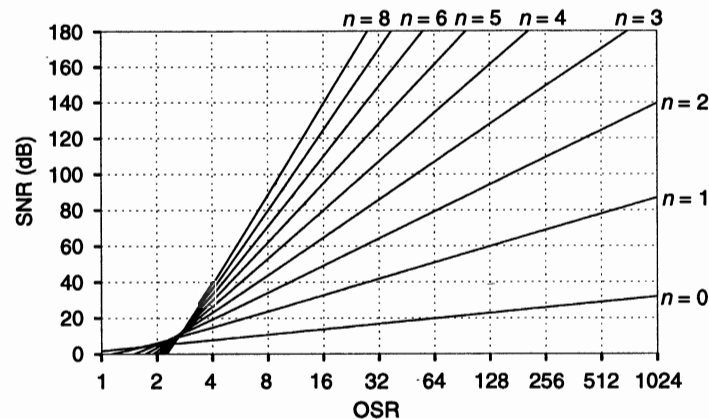


Figure 4.1 Theoretical SNR of a $(1 - z^{-1})^n$ modulator vs. the oversampling ratio for $n = 0, \dots, 8$. High-order modulators achieve very high SNR at moderate values of OSR.

interactions between loops and analyzing their component modulators separately. Similarly, continuous-time modulators can be converted into equivalent discrete-time modulators and the discrete-time model subjected to stability analysis (see Section 4.5). The multibit case is handled in Chapter 14.

In the following section, the standard linear analysis of a single-loop modulator is used to describe the phenomenon of instability in linear system terms, since this terminology is familiar to most designers. Following this simple linear model, Section 4.2.3 presents a highly accurate model for a $\Delta\Sigma$ modulator incorporating two linear systems coupled by a nonlinear integral equation. In Section 4.3, rigorous results on the stability of first- and second-order modulators are given. Derivation of these results requires mathematical tools that are likely to be unfamiliar to many designers and so the emphasis is placed on illustration rather than detailed derivations. Section 4.4 describes a practical design methodology that has been successfully used to design stable, high-order modulators and includes a design example. Section 4.5 presents the equivalence relations between continuous-time and discrete-time modulators. Section 4.6 describes nonlinear stabilization techniques, and Section 4.7 summarizes the discussion.

4.2 LINEAR ANALYSIS

In this section we apply linear system concepts to a $\Delta\Sigma$ modulator. Although the discussion is thereby flawed from the outset, it yields valuable insights into the nature of instability in $\Delta\Sigma$ modulators.

4.2.1 The Linear Model

Figure 4.2 shows a general block diagram for a single-quantizer $\Delta\Sigma$ modulator. The modulator is split into a linear block (the loop filter) and a nonlinear block (the quantizer), with the linear block having arbitrary transfer functions from its two inputs U and V to its

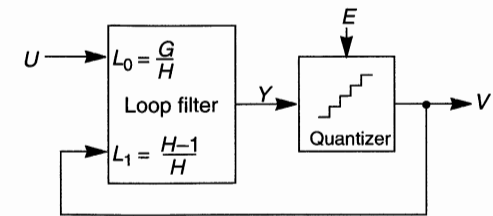


Figure 4.2 General block diagram of a single-quantizer $\Delta\Sigma$ modulator.

single output y . These transfer functions have been labeled for convenience as

$$L_0(z) = G(z)/H(z) \quad (4.2)$$

and

$$L_1(z) = [H(z) - 1]/H(z) \quad (4.3)$$

With these assignments, the output of the linear block is

$$Y(z) = L_0(z)U(z) + L_1(z)V(z) \quad (4.4)$$

By defining the error signal e as $E(z) = V(z) - Y(z)$, Eq. (4.4) can be re-arranged to give the familiar formula for the output of the modulator in terms of its input and the error signal:

$$V(z) = G(z)U(z) + H(z)E(z) \quad (4.5)$$

By showing that the output consists of independently filtered signal and noise components, Eq. (4.5) captures the essence of noise-shaping loops. Note that Eq. (4.5) indicates that a $\Delta\Sigma$ modulator with a signal transfer function $G(z)$ and an input $U(z)$ is equivalent to a $\Delta\Sigma$ modulator with a signal transfer function (STF) of unity and an input $G(z)U(z)$. This observation allows us to focus on the NTF $H(z)$ in our discussions of loop stability since $G(z)$ merely acts as a prefilter on the input.

One drawback of Eq. (4.5) is that it hides the fact that the noise is signal dependent. This omission can lead to serious modeling errors. As one example of such an error, consider the linear model of a $\Delta\Sigma$ model with a binary quantizer shown in Figure 4.3. This model is identical to that of Figure 4.2 except for the addition of an arbitrary gain $k > 0$ at the input of the quantizer. Since the quantizer is binary, this addition does not affect the operation of the modulator, but it *does* affect the linear model. Following an analysis similar to that performed previously, we find that the output of the modulator is now

$$V(z) = G'(z)U(z) + H'(z)E'(z) \quad (4.6)$$

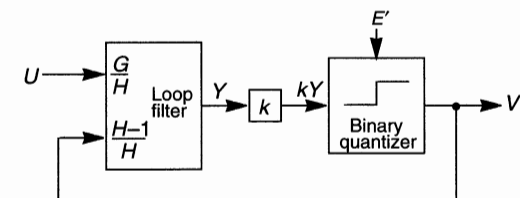


Figure 4.3 General single-loop $\Delta\Sigma$ modulator with a binary quantizer. The gain block k represents the quantizer gain.

where

$$E'(z) = V(z) - kY(z) \quad (4.7)$$

$$G'(z) = \frac{kG(z)}{k + (1-k)H(z)} \quad (4.8)$$

and

$$H'(z) = \frac{H(z)}{k + (1-k)H(z)} \quad (4.9)$$

Thus, the NTF and STF are different and may even be unstable! The reader should note that there is no contradiction here. Both Eqs. (4.5) and (4.6) are exact descriptions of the modulator, *provided* that $E(z)$ and $E'(z)$ are defined correctly. The issue at hand is “What is the best definition for $E(z)$?” or equivalently “What is the gain of the quantizer?”

This question can be answered by finding the value of k that minimizes the error signal’s power. This optimum value k_{opt} decorrelates the error and signal components and is given by

$$k_{\text{opt}} = \frac{\langle y, v \rangle}{\langle y, y \rangle} = \frac{\text{cov}(y, v)}{\text{var}(y)} = \lim_{N \rightarrow \infty} \frac{\sum_{n=0}^N y(n)v(n)}{\sum_{n=0}^N y(n)^2} \quad (4.10)$$

This formula clearly shows that k_{opt} depends on y , which in turn depends on the modulator input u . Consequently one must have a priori knowledge of signal statistics in order to find the optimum linear model, and this model varies as the input varies. A fixed value of k would be preferable since the designer could then talk about *the* noise and signal transfer functions, without having to qualify such statements by specifying the input. Unfortunately, this is not possible.

We shall see in the following section that the variability of k_{opt} can be viewed as being a cause of instability in high-order modulators.

4.2.2 Root Locus of a High-Order Modulator [3]

Figure 4.4 plots the root locus of the NTF of a sixth-order modulator as the quantizer gain varies from 1 to 0. As this figure indicates, two of the closed-loop poles move outside the unit circle when $k < 0.572$. Thus, if the quantizer gain is less than 0.572, then the linear model is unstable. In loose terms, when the input to the quantizer is large, k falls, and this in turn results in larger quantizer inputs. Intuitively at least, it can be seen that this may lead to runaway states.

The root locus thus suggests that in order for a modulator to be stable, the input to the quantizer must not be allowed to become too large. Since the input to the quantizer is given by

$$Y(z) = G(z)U(z) + [H(z) - 1]E(z) \quad (4.11)$$

this requirement leads to the conclusion that the “gain” of $H(z) - 1$, or more simply the “gain” of $H(z)$, must not be too large. Since an NTF of the form $H(z) = (1 - z^{-1})^n$ has a

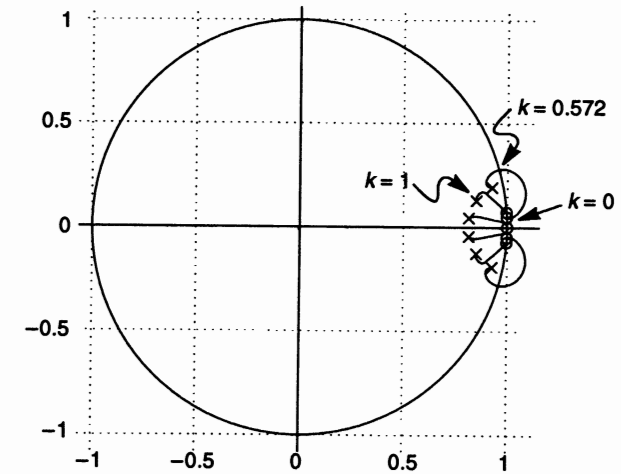


Figure 4.4 Root locus of a sixth-order modulator. Poles move outside the unit circle when $k < 0.572$.

peak gain of $|H(-1)| = 2^n$, we gain insight into why high-order modulators with simple NTFs are unstable. To quantify this further requires a more sophisticated model of the quantizer, one that varies the quantizer gain in accordance with the statistics of its input.

4.2.3 Describing Function Method

In [4], Ardalan and Paulos develop such a model. Their technique is based on a generalization of the describing function method. Their method splits the modulator into two linear systems, one for the signal and one for the noise. This section summarizes their approach and compares the predicted SNR curve against simulations for the modulator whose NTF was used to construct Figure 4.4. In the interest of brevity, only the dc input case is described.

For dc inputs, signal components are dc components. Likewise, the noise components are the ac components. The input, y , to the quantizer is thus split into dc and ac components according to

$$y = y_0 + y_1 \quad (4.12)$$

where y_0 is the mean value of y and y_1 is a random component with zero mean. For inputs other than dc, y_0 would be the component of y that is correlated with the input and y_1 would be the remainder and, as such, uncorrelated with both the input and y_0 .

The quantizer is modeled as shown in Figure 4.5. The signal passes through one gain, while the noise passes through another. The gains k_0 and k_1 are chosen such that the modeling error $e = \text{sgn}(y_0 + y_1) - k_0 y_0 - k_1 y_1$ is as small as possible, in the least-square sense. This, in turn, makes e uncorrelated with both y_0 and y_1 . When this model is used to replace the quantizer in Figure 4.2, the modulator is split into two linear systems, as illustrated in Figure 4.6. The first system is driven by the input u , whereas the second system is driven by the error term e .

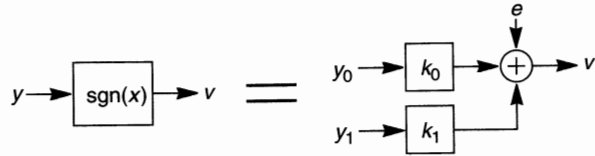


Figure 4.5 Modeling the quantizer with two linear gains and a noise source; $v = \text{sgn}(y)$ becomes $v = k_0 y_0 + k_1 y_1 + e$.

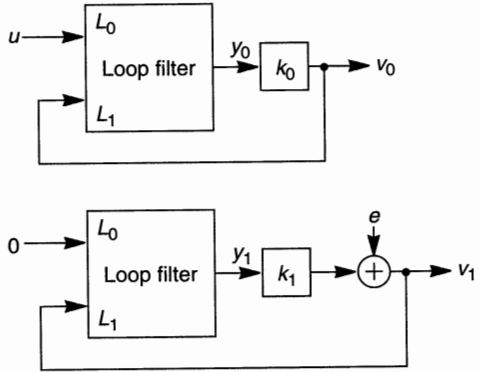


Figure 4.6 Applying the describing function method (which uses the quantizer model of Figure 4.5) to the general modulator of Figure 4.2 splits the modulator into two linear systems.

By assuming that e is white noise, that y_1 has a Gaussian probability density function, and that the dc gain of the STF is one (so that $v_0 = u$), Ardalan and Paulos derived the following relationships.

The variance σ_e^2 of the error signal is

$$\sigma_e^2 = 1 - u^2 - \frac{2}{\pi} \exp(-2\text{erf}^{-1}(u)^2) \quad \text{where } \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-x^2} dx \quad (4.13)$$

The variance σ_1^2 of y_1 and the gain k_1 form the solution to the coupled equations

$$\sigma_1^2 = \sigma_e^2 \|\hat{H}(z)\|_2^2 \quad (4.14)$$

$$k_1 \sigma_1 = \sqrt{2/\pi} \exp[-\text{erf}^{-1}(u)^2] \quad (4.15)$$

where $\|\hat{H}(z)\|_2^2$ is the power gain¹ of

$$\hat{H}(z) = \frac{L_1(z)}{1 - k_1 L_1(z)} \quad (4.16)$$

1. The notation reflects the fact that the power gain is the square of the “two norm,” which, by Parseval’s theorem, is given by

$$\|H\|_2^2 = \frac{1}{\pi} \int_0^\pi |H(e^{j\omega})|^2 d\omega = \sum_{n=0}^{\infty} h(n)^2$$

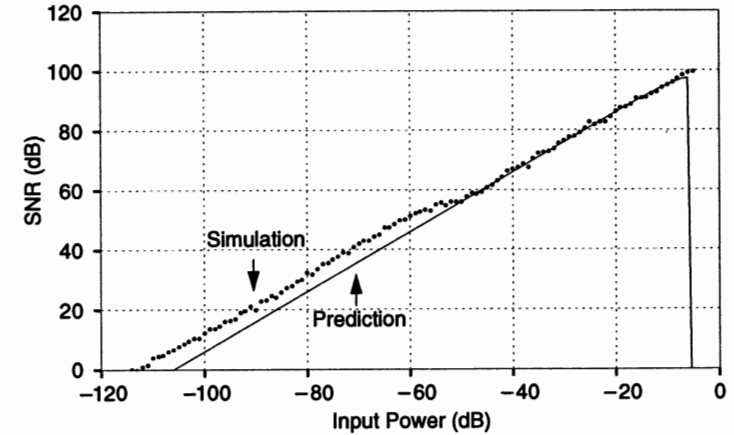


Figure 4.7 The SNR curve determined by the describing function method of Ardalan and Paulos compared against SNR measurements from simulations. The simulations used sine wave inputs where a unit-amplitude sine wave was the 0-dB reference.

The final two formulas for the parameters in the quantizer model are

$$y_0 = \sigma_1 \sqrt{2} \text{erf}^{-1}(u) \quad (4.17)$$

and

$$k_0 = \frac{1}{y_0} \text{erf}\left(\frac{y_0}{\sqrt{2}\sigma_1}\right) \quad (4.18)$$

Note that the linear model depends on the input u . In order to apply the above results, a pair of nonlinear equations [Eq. (4.14) and Eq. (4.15)] must be solved for a variety of u values and the results plotted. For high-order modulators, this can only be done using numerical methods. As an illustration, the above procedure can be applied to the modulator whose root locus was plotted in Figure 4.4, resulting in the SNR curve of Figure 4.7. Simulations using sine wave inputs were used to determine the true signal-to-noise ratio of the modulator and, as Figure 4.7 shows, the measured SNR is quite close to the predicted SNR. Even more remarkably, the prediction of the maximum stable input, $u_{\max} = 0.566$, is within 4% of $u_{\max} = 0.548$, the value determined by long simulations.

4.3 FIRST- AND SECOND-ORDER MODULATORS

In this section rigorous theoretical results on the stability of first- and second-order modulators are described. We shall see that although these modulators are fairly well understood, a few questions still remain.

4.3.1 First-Order Modulator

The first-order modulator, shown in Figure 4.8, includes both the quantizer gain k and the integrator pole β as parameters. In an ideal first-order modulator, $k = \beta = 1$, and it follows that $G(z) = 1$ and $H(z) = 1 - z^{-1}$. The time-domain equations for this modulator can then be written as

$$y(n) = u(n) - e(n-1) \quad (4.19)$$

$$v(n) = \text{sgn}[y(n)] \quad (4.20)$$

$$e(n) = v(n) - y(n) \quad (4.21)$$

Using this formulation, we will show that if $|y(0)| \leq 2$ and $|u(n)| \leq 1$ for all n , then $|y(n)| \leq 2$ for all n . The argument is an inductive one.

First, it follows directly from the quantizer characteristic that $|y(n)| \leq 2 \Rightarrow |e(n)| \leq 1$. Now,

$$y(n+1) = u(n+1) - e(n) \quad (4.22)$$

So that under the assumption $|u(n+1)| \leq 1$ and the induction hypothesis $|y(n)| \leq 2$,

$$|y(n+1)| \leq |u(n+1)| + |e(n)| \leq 2 \quad (4.23)$$

Thus the induction step is satisfied; that is, $|y(n)| \leq 2 \Rightarrow |y(n+1)| \leq 2$. Since the starting inequality $|y(0)| \leq 2$ is simply part of the premise, the desired result has been proven.

If $|y(0)| > 2$ and $|u(n)| < 1$ for all n , it is a simple matter to show that $|y(n)|$ decreases monotonically until $|y(n)| \leq 2$, whereupon the preceding conditions hold and $|y(n)|$ remains less than or equal to 2. Similarly, when $u(n) > 1$ for all n , or when $u(n) < -1$ for all n , $y(n)$ increases or decreases monotonically and $|y(n)|$ is unbounded.

Thus, the ideal first-order modulator is unstable with inputs greater than 1 in magnitude, stable with inputs less than or equal to 1 in magnitude, and furthermore is able to recover from any initial state, including one caused by overload.

More generally, $\beta \neq 1$. Choosing $k = \beta$ results in $G(z) = \beta$ and $H(z) = 1 - \beta z^{-1}$. Following a procedure similar to that described above, the bound on the integrator state is found to be $|y(n)| \leq 2/\beta$, with the proviso that $|y(0)| \leq 2/\beta$ and $|u(n)| \leq 2/\beta - 1$ for all n . If $\beta < 1$, as would be the case with finite op-amp gain, the modulator has a bounded state for *all* inputs, not just those whose magnitudes are less than $2/\beta - 1$. However, the bound is very large when $|u|$ exceeds $2/\beta - 1$. If $\beta > 1$, which would be the case for a chaotic modulator [5], the modulator is stable despite the fact that the embedded integrator is unstable. However, the maximum stable input is less than 1.

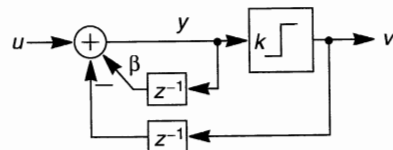


Figure 4.8 First-order modulator. For the ideal modulator, $\beta = k = 1$.

From the completeness of this discussion, it should be clear that there is very little mystery left regarding the stability of the first-order modulator.

The no-overload argument can be generalized to higher order and multibit modulators, but it only yields useful bounds for high-order modulators if the quantizer is given m levels, with $m > 2$ [6, 7]. The conclusion of the argument is that the maximum stable input, $u_{\max}$, expressed as a fraction of the outermost quantizer levels, is at least

$$u_{\max} \geq \frac{m - \sum_{n=1}^{\infty} |h(n)|}{m-1} \quad (4.24)$$

4.3.2 Second-Order Modulator

In this section, results on the stability of a class of second-order modulators are described. Although the state bounds are only valid for dc inputs, they are fairly tight and are certainly adequate for design [8–12].

Figure 4.9 shows the block diagram of a general second-order modulator. In the standard case, $\alpha = \beta = 0$, $\gamma = 1$, and it follows that $G(z) = z^{-1}$ and $H(z) = (1 - z^{-1})^2$. In practice, γ is used to adjust the poles of the NTF so that the modulator is “more stable,” α is used to move the zeros of the NTF along the unit circle for optimum noise shaping and β models the effects of finite op-amp gain. In this more general case, the signal and noise transfer functions are

$$G(z) = \frac{kz}{z^2 + (-2 + k + \alpha + \beta + k\gamma)z + (1 - \beta - k\gamma)} \quad (4.25)$$

$$H(z) = \frac{z^2 + (-2 + \alpha + \beta)z + (1 - \beta)}{z^2 + (-2 + k + \alpha + \beta + k\gamma)z + (1 - \beta - k\gamma)} \quad (4.26)$$

It should be apparent from these equations and the fact that there are four design parameters in Eq. (4.26) that any second-order NTF can be realized with the system of Figure 4.9.

For this section, we shall only deal with the case $\alpha = 0$, $\beta = 0$, and $u = \text{const}$. Under these assumptions, the NTF has two zeros at $z = 1$ and arbitrary poles:

$$H(z) = \frac{(z-1)^2}{z^2 + (-2 + k + k\gamma)z + (1 - k\gamma)} \quad (4.27)$$

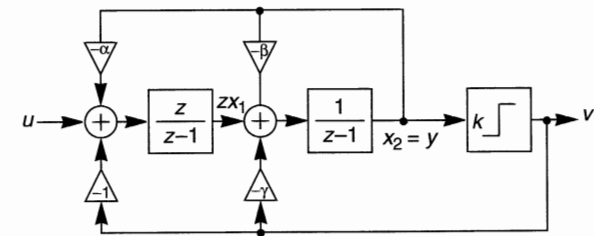


Figure 4.9 General second-order modulator.

The state equations for this system are

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix} + \begin{bmatrix} -(\gamma+1) \\ -1 \end{bmatrix} v(n) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u \quad (4.28)$$

$$v(n+1) = \text{sgn}(x_2(n+1)) \quad (4.29)$$

By considering the cases $v = 1$ and $v = -1$ separately, these equations may be solved using the z -transform if one is careful of initial conditions, or they can be solved more simply by assuming a quadratic form for x_1 and a linear form for x_2 , substituting into Eq. (4.28) and then matching coefficients. In either case, the result is that the state trajectories lie on the parabolas

$$C_p: x_1 = -\frac{(x_2 - x_{2p})^2}{2(1-u)} + (x_2 - x_{2p})\left(\frac{1}{2} + \frac{\gamma - x_{2p}}{1-u}\right) + x_{1p} \quad (v = 1) \quad (4.30)$$

$$C_n: x_1 = \frac{(x_2 - x_{2n})^2}{2(1+u)} + (x_2 - x_{2n})\left(\frac{1}{2} + \frac{\gamma + x_{2n}}{1+u}\right) + x_{1n} \quad (v = -1) \quad (4.31)$$

where (x_{1p}, x_{2p}) and (x_{1n}, x_{2n}) are the coordinates of some known points on C_p and C_n , respectively.

From these equations, it is possible to find a region of state space S , with the property that any point inside S maps to another point inside S under the nonlinear mapping [Eqs. (4.28) and (4.29)]. Such a region is called a *positively invariant set*. Figure 4.10 illustrates such a set for the case $\gamma = 1$ and $u = 0.54321$. This figure shows how points in the upper and lower half planes map under Eq. (4.28) with $v = 1$ and $v = -1$, respectively; it is apparent from this figure that both images are contained in the original set and that part of the original set is inaccessible and hence not strictly required. Figure 4.11 plots the boundary of the positively invariant set together with 10,000 samples of the modulator's state. This figure demonstrates that the state of the modulator is indeed contained in the given set. Figure 4.11 also shows that the set is only partially filled, and consequently the bounds are somewhat conservative for this case.

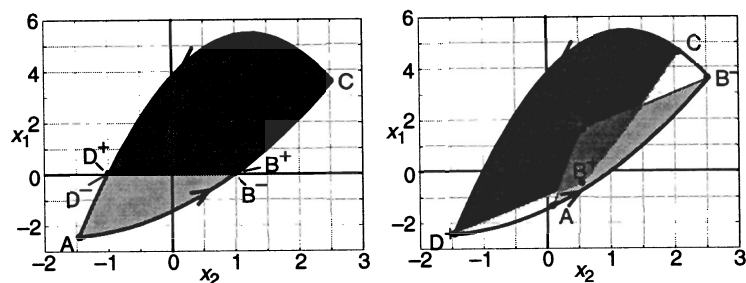


Figure 4.10 Positively invariant set for the standard ($\gamma = 1$), second-order modulator with an input $u = 0.54321$. The graph on the left is the original set and the graph on the right is its image after one iteration of Eq. (4.28). The motion of states along the bounding curves is counterclockwise, as indicated by the arrows.

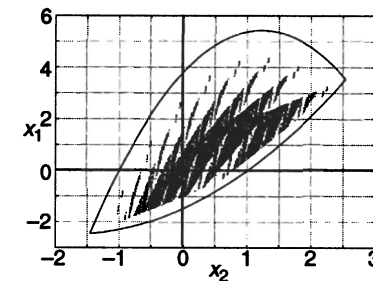


Figure 4.11 Plot of the boundary of the positively invariant set and 10,000 iterations of the state equations [Eqs. (4.28) and (4.29)] with $u = 0.54321$ and $\gamma = 1$.

By solving for the positively invariant set as a function of the input and the γ parameter, analytic bounds on the state variables can be found. In the general case, one has to be wary of the possibility that the images of B^+ or D^- may lie outside of a set bounded by two simple parabolic segments. This eventuality causes the formulas for the analytic bounds to become rather complex, as they must handle several different cases. For the sake of clarity, only the formulas for the standard case $\gamma = 1$ will be given:

$$|x_1|_{\max} = |u| + 2 \quad (4.32)$$

$$|x_2|_{\max} = \frac{(5 - |u|)^2}{8(1 - |u|)} \quad (4.33)$$

These results were first presented as part of [8] and later generalized to $\gamma \neq 1$ in [12]. In this later work, the authors use these bounds to show that a value of γ that maximizes the peak SNR is $\gamma = 1.16$.

Figure 4.12 plots the bounds given by Eqs. (4.32) and (4.33) as functions of u and compares them with simulation results. As this figure shows, the analytical bounds are fairly tight at low values of u , namely within approximately 20% of the maximum values seen in long simulations.

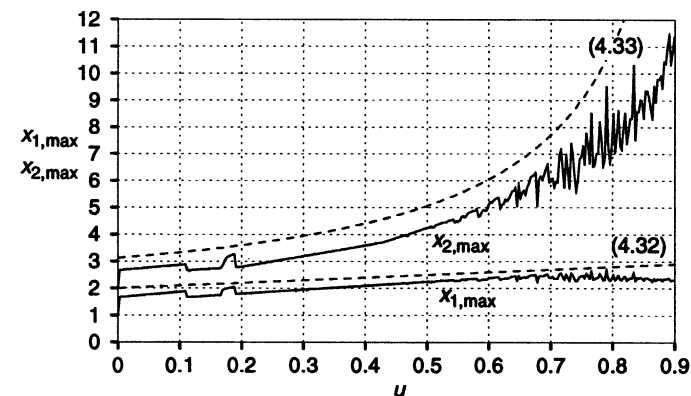


Figure 4.12 Comparison of the analytical state bounds (4.32) and (4.33) with the maximum values seen in simulations of the second-order modulator with dc inputs. Each simulation was run for 10^6 time steps.

This section has shown how direct analysis of the second-order modulator in terms of its state-space mapping can be used to find analytical bounds for the state variables. The mathematics become difficult for the general case and, as of this writing, analytical solutions have only been found for $\alpha = \beta = 0$.

Third-order modulators have also been studied using a similar approach [13, 14]. The formulae are quite complex and will not be included here for the sake of brevity.

4.4 PRACTICAL DESIGN METHODOLOGY

The previous section presented rigorous results that define the stable operating region of a modulator in terms of its state variables. Since this method has not yet been applied to modulators of order greater than 3, it does not help the design engineer who has been assigned the task of designing a stable high-order loop. This section comes to the rescue by describing an empirical method based on ordinary linear filter design that can be used to design high-order loops. Chapter 5 will cover in detail the implementation of the loop filter once a stable NTF has been designed, as well as other more subtle points that affect the detailed implementation.

4.4.1 Cookbook Design Procedure

In Section 4.2.2, it was suggested that a high-order modulator would be stable if the gain of the NTF were kept low enough. This simple intuition leads to a highly practical design methodology that has yielded numerous stable designs [15–17]. The methodology is straightforward and will be presented in the form of a step-by-step recipe.

1. Choose a modulator order and an NTF filter family.

The authors have used high-pass Butterworth, Chebyshev, and maximally flat all-pole filters. For each family, a particular filter can be specified by the filter cutoff frequency.

2. Pick a value for the filter cutoff frequency and scale the transfer function so that the first sample of the impulse response is 1.

This step is necessary since the L_1 loop filter must contain at least one unit of delay. If it did not, the modulator would represent an inconsistent, unrealizable system. To see this, note that errors introduced by the quantizer pass directly to the output without delay (and hence the impulse response of the NTF must have 1 as its first sample). Thus, only values of the quantizer error incurred in previous time steps can be allowed to form the current input to the quantizer.

In terms of transfer functions, since $h(0) = H(\infty)$, this requirement amounts to

$$H(\infty) = 1 \quad (4.34)$$

This constraint is satisfied by setting the leading coefficients (those multiplying z^n) of the numerator and denominator polynomials of $H(z)$ to 1.

3. Construct a modulator with this NTF and either simulate it or use the describing function method to determine its maximum stable input and peak SNR.

Simulations may be performed using either the general modulator structure of Figure 4.2 (after the trivial step of solving for the loop filter transfer functions) or a particular modulator structure, such as one of those presented in Section 5.6 (after the more difficult step of solving for the filter coefficients).

4. If the modulator is unstable, reduce the out-of-band gain of the NTF.

For the filter types described above, this is accomplished by lowering the filter cutoff. Lowering the filter cutoff reduces the magnitude of the first sample of the impulse response. When the filter is rescaled to make the first impulse sample equal to 1, the resulting filter passband gain will be reduced in comparison to the original filter.

A rule-of-thumb, derived from dozens of stable designs done by the authors, states that the out-of-band gain of the NTF should be about 1.5. In many cases this rule-of-thumb is accurate enough that iteration is not required. The designer would be wise to use this rule to select the filter cutoff in step 2 before proceeding with the detailed characterization of step 3. This rule applies only to filter shapes that are flat-topped, such as Butterworth, Chebyshev, and so on. Unusual filter shapes, especially ones with large resonant peaks, will not benefit from this approximation.

5. If the modulator is stable but the SNR is not adequate, then increase the out-of-band gain of the NTF.

By pushing the modulator closer to the edge of instability, an aggressive NTF yields a peak SNR that can be as much as 20 dB higher than that produced by the first-cut design. However, modulators with aggressive NTFs are easily driven into instability by large inputs or small parameter shifts.

If the desired SNR is incompatible with modulator stability, a higher modulator order is needed. However, the designer should be aware that simply increasing the modulator order does not allow an arbitrarily high SNR to be achieved with a fixed oversampling ratio. For example, a simple information-theoretic argument shows that it is impossible to get 16-bit performance with a single-bit quantizer operated at an oversampling ratio less than 16.¹ Practical limitations increase the minimum value of the oversampling ratio, as we shall see in the next section.

4.4.2 SNR Limits [17]

Figure 4.13 shows the maximum signal-to-noise ratio achievable with modulators whose NTFs have all their zeros at $z = 1$. The modulators were designed using the method described in the previous section and simulated to verify their stability. Note the tremendous decrease in performance that results from the stability constraint. In particular, Figure 4.1 indicates that a SNR of 160 dB would result from a $(1 - z^{-1})^5$ NTF at an

1. Consider a system that transforms a single-bit stream at f_s bits/sec into a Nyquist-rate N -bit data stream at f_s/R bits/sec. Since each sample in a Nyquist-rate signal is independent of all other samples, the output data rate is Nf_s/R bits/sec. In order for the information content of the output stream to not exceed that of the input stream, N must be less than or equal to R .

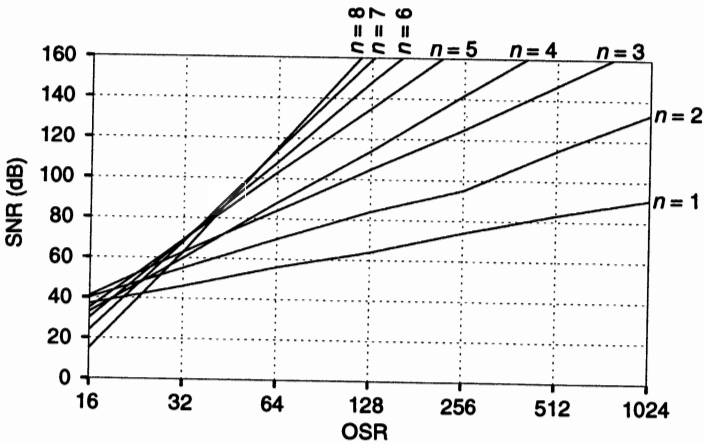


Figure 4.13 Maximum SNR achievable by modulators of order n with coincident zeros, as a function of the oversampling ratio.

oversampling ratio of 64, but Figure 4.13 shows that due to limitations imposed by stability, the achievable SNR is 60 dB lower.

Some of this loss can be recovered by spreading the NTF zeros across the band of interest in a manner that minimizes the in-band noise. Table 4.1 lists the analytically determined zero placements for minimum in-band noise and Figure 4.14 plots the maximum SNR achievable with NTFs employing such placements. This figure shows that an optimized fifth-order modulator operated at an oversampling ratio of 64 gains approximately 20 dB by using an optimized NTF.

4.4.3 Sixth-Order NTF

To make the above procedure more concrete, consider the problem of designing a sixth-order modulator with an oversampling ratio of 40. To make the signal transfer function maximally flat when a single feed-in is used in the cascade-of-resonators structure

TABLE 4.1 ZERO PLACEMENT FOR MINIMUM IN-BAND NOISE [ZEROS ARE GIVEN AS FRACTIONS OF THE PASSBAND EDGE].

n	Zero locations, normalized to ω_B	SNR improvement (dB)
1	0	0
2	± 0.57735	3.5
3	0, ± 0.77460	8
4	$\pm 0.11559, \pm 0.74156,$	13
5	0, $\pm 0.28995, \pm 0.82116$	18
6	$\pm 0.23862, \pm 0.66121, \pm 0.93247$	23
7	0, $\pm 0.40585, \pm 0.74153, \pm 0.94911$	28
8	$\pm 0.18343, \pm 0.52553, \pm 0.79667, \pm 0.96029$	34

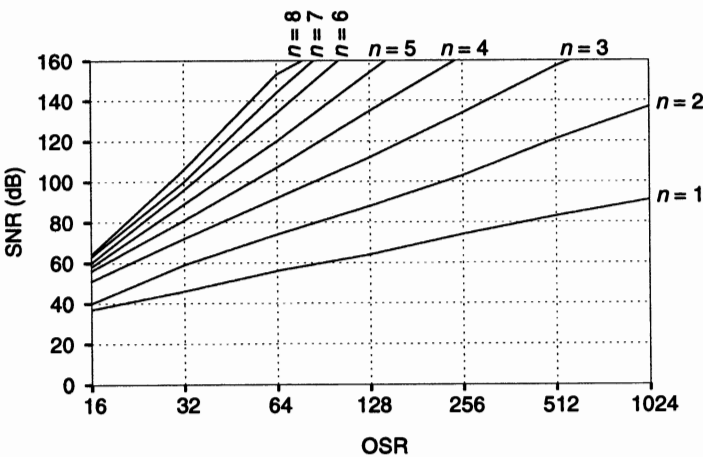


Figure 4.14 Maximum SNR achieved by modulators of order n whose zeros have been optimally spread across the band of interest.

(Section 5.6.4), choose as the filter family those filters whose denominator polynomials are those of maximally flat low-pass filters. The Butterworth family could have been selected just as well, since the passband variation of a low-pass filter with Butterworth poles is very slight.

To ensure that dc signals are faithfully reproduced and to reduce the likelihood of tones, we will require that two of the NTF zeros be located at dc. The remaining zeros are then placed such that the mean-square value of the in-band gain of the NTF is minimized. These frequencies can be found analytically and are given by $\omega_1 = 0.340\omega_B$ and $\omega_2 = 0.866\omega_B$, where $\omega_B = \pi/40$.

Filter design software is then used to find the value of the NTF high-pass corner frequency, which yields an out-of-band gain of 1.5 after scaling the transfer function according to the $H(\infty) = 1$ constraint. The poles and zeros of the resulting filter have already been plotted in Figure 4.4 but are also tabulated in Table 4.2.

Figure 4.15 plots the response of this filter. The out-of-band gain is 3.5 dB (1.5 in absolute terms) and the mean-square in-band gain is approximately -80 dB. The latter amount of attenuation leads to a predicted in-band noise power of slightly less than -100 dB.

Discrete-time simulations of this modulator, the results of which were plotted earlier in Figure 4.7 and discussed in Section 4.2.3, show that this design is stable for inputs up to 0.55 and achieves a peak SNR of approximately 100 dB, which correlates well with the SNR predicted by the simple linear model.

If this level of performance were to be judged unsatisfactory, the designer could repeat the process using a higher value for the out-of-band filter gain.

TABLE 4.2 ZEROS AND POLES OF THE NTF FOR THE EXAMPLE MODULATOR

Zeros	1, 1	$0.99874 \pm j0.05024$	$0.99734 \pm j0.07290$
Poles	$0.81483 \pm j0.04427$	$0.85180 \pm j0.12765$	$0.92705 \pm j0.19164$

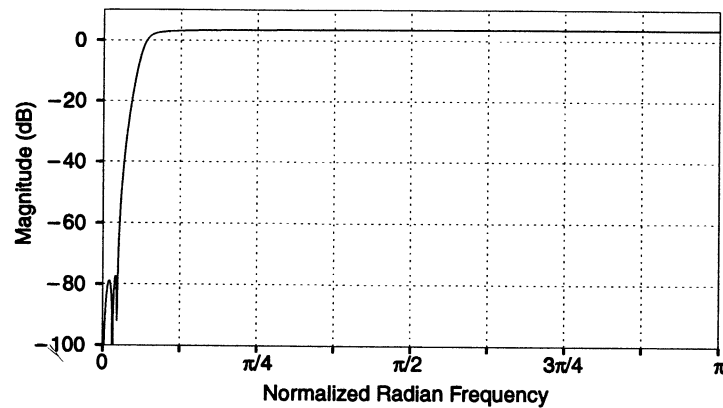


Figure 4.15 Magnitude response of the example sixth-order NTF.

This design process has been coded into a set of MATLAB functions for NTF design and simulation that are available from the second author. As of this writing, these functions are available via anonymous ftp from next242.ece.orst.edu (128.193.48.65); ftp them directly, or follow the link from <http://www.ece.orst.edu/~schreier>.

4.4.4 Design Trade-Offs

In this section, the trade-off between the NTF gain and the peak SNR and input range is discussed in the context of the example modulator. The influence of the comparator gain on the design process is also described.

As suggested in the previous section, a higher peak SNR may result if the out-of-band gain of the NTF can be set to a larger value without causing the modulator to become unstable. Figure 4.16 shows that this is indeed the case for the modulator family under

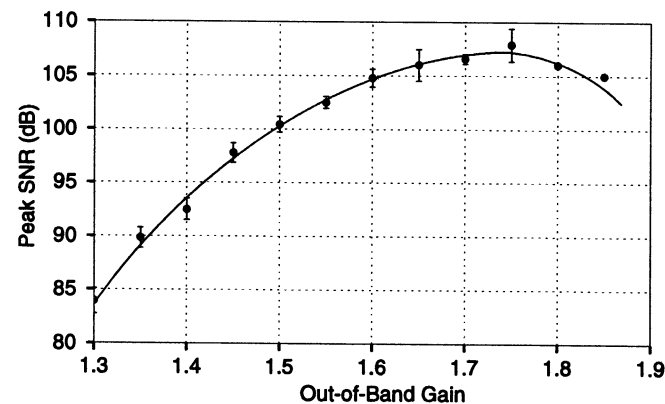


Figure 4.16 Peak SNR as a function of the out-of-band gain for sixth-order modulators with optimized zeros, operated at an oversampling ratio of 40.

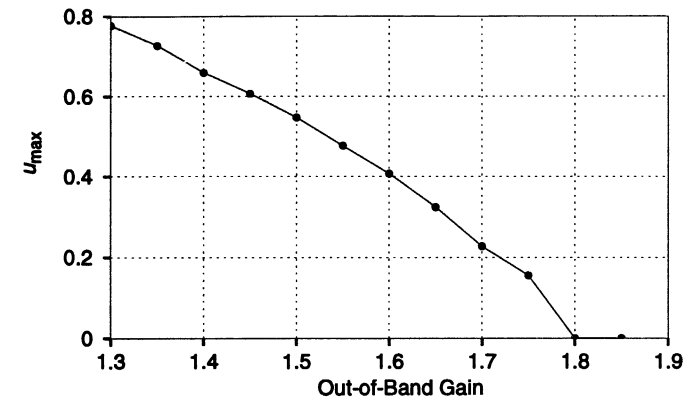


Figure 4.17 Maximum stable input as a function of the out-of-band gain for sixth-order modulators with optimized zeros.

consideration. For these modulators, an out-of-band gain of 1.75 results in a peak SNR of 107 dB. The price paid for this desirable increase in the peak SNR is an undesirable decrease in the stable input range, as illustrated in Figure 4.17. This figure shows that using an out-of-band gain of 1.75 makes the stable input range less than 0.2. Once the out-of-band gain exceeds 1.75, the loss of input signal range more than offsets the reduction in the noise level, and the peak SNR drops. Typically, commercial $\Delta\Sigma$ converters are designed such that the maximum specified input signal ranges from 50 to 80% of the effective comparator feedback signal. Anything less than 50% suggests high NTF gains and low stability; anything more than 80% indicates that the NTF is not very aggressive, and some of the benefits of using high-order loops will be lost. This discussion of stable dc input range does not address the issue of stability in the presence of dynamic or transient input signals. These issues will be discussed in Chapter 5.

The issue of the comparator gain, which is so critical to the stability of the modulator, is not explicitly addressed by the cookbook design method. As a result, “the” noise and signal transfer functions of the modulator are not identical to those designed using the given method. For the example modulator, the method of Ardaian and Paulos can be used to plot the quantizer gain as a function of the modulator input. Figure 4.18 shows that the gain of the quantizer varies from 1.6 at low input levels to approximately 1.2 at high input levels. The fact that the quantizer gain is not unity leads to transfer function errors and the variability in the quantizer gain leads to variability in the transfer functions. Fortunately, the high loop gain makes the in-band error in the STF negligibly small, on the order of 10^{-4} dB, and the out-of-band peaking is also small, less than 0.7 dB, as illustrated in Figure 4.19. This figure also shows that the NTF does not exhibit out-of-band peaking (which would otherwise have indicated a susceptibility to oscillation) and thus the nonunity comparator gain is not an especially troublesome issue.

A more advanced design procedure can be envisioned that would take the comparator gain into account at the transfer function design stage and thus avoid any surprises due to nonunity comparator gain. The previous example suggests that an NTF, which has sufficient out-of-band gain to yield a quantizer gain of 1, would result in an exceedingly small

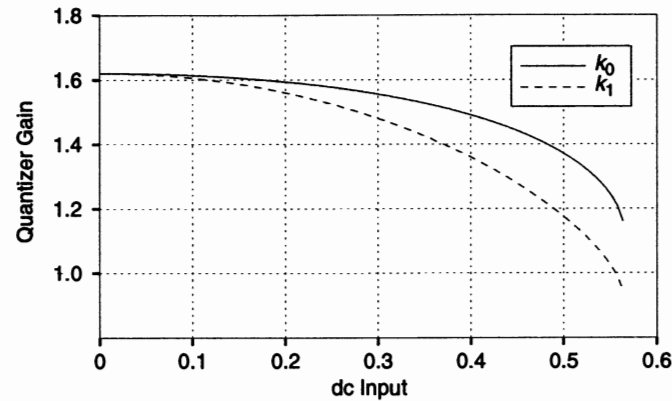


Figure 4.18 Quantizer gains as functions of the dc input for the example modulator, as determined by the method of Ardan and Paulos.

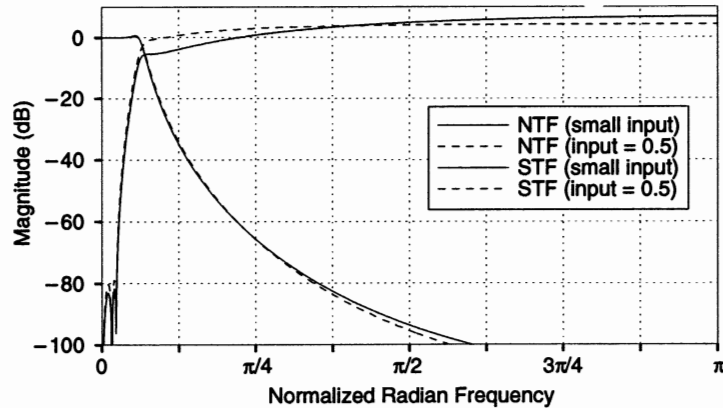


Figure 4.19 Noise and signal transfer functions change as the input level changes.

stable input range. It is still unclear how one could design a modulator that has a quantizer gain (defined by the method of Ardan and Paulos) of unity while at the same time having an acceptable stable input range. Classical NTF filter shapes such as Butterworth may not be optimal in this respect, and future researchers may discover that other, more unconventional filter shapes are required to meet both criteria.

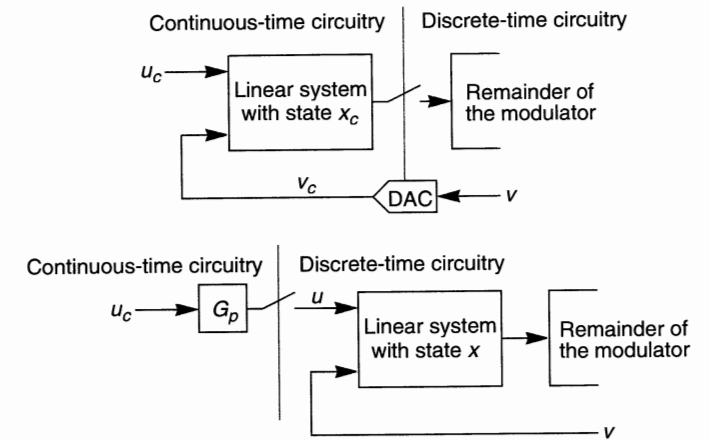


Figure 4.20 Equivalence between a continuous-time modulator and a discrete-time modulator with a prefilter on the input.

the resulting equivalent discrete-time system can be analyzed using the methods described in this chapter. Alternatively, a working discrete-time system can be mapped into a continuous-time system by the inverse transformation.

Figure 4.20 illustrates the mapping we are trying to achieve. In the continuous-time modulator, the leftmost block of the modulator is made with continuous-time circuitry while the remainder is discrete-time. In the discrete-time model, the input is filtered, sampled, and then fed to a fully discrete-time modulator. The prefiltering operation does not affect the stability of the modulator and for the purposes of this chapter can be ignored.

Without loss of generality, we shall assume that the sampling frequency is 1 Hz. The state equations for the linear parts of the continuous and discrete modulators are, respectively,

$$\dot{x}_c = A_c x_c + B_c \begin{bmatrix} u_c \\ v_c \end{bmatrix} \quad (4.35)$$

and

$$x(n+1) = A x(n) + B \begin{bmatrix} u(n) \\ v(n) \end{bmatrix} \quad (4.36)$$

Equation (4.35) may be solved to yield the following equation:

$$x_c(t) = e^{A_c t} x_c(0) + e^{A_c t} \int_0^t e^{-A_c \tau} B_c \begin{bmatrix} u_c(\tau) \\ v_c(\tau) \end{bmatrix} d\tau \quad (4.37)$$

A sample of x_c may be found from the previous sample and the linear system's inputs via

$$\begin{aligned}
 x_c(n+1) &= e^{A_c(n+1)} x_c(0) + e^{A_c(n+1)} \int_0^{n+1} e^{-A_c\tau} B_c \begin{bmatrix} u_c(\tau) \\ v_c(\tau) \end{bmatrix} d\tau \\
 &= e^{A_c} \left(e^{A_c n} x_c(0) + e^{A_c n} \int_0^n e^{-A_c\tau} B_c \begin{bmatrix} u_c(\tau) \\ v_c(\tau) \end{bmatrix} d\tau \right) \\
 &\quad + e^{A_c(n+1)} \int_n^{n+1} e^{-A_c\tau} B_c \begin{bmatrix} u_c(\tau) \\ v_c(\tau) \end{bmatrix} d\tau \\
 &= e^{A_c} x_c(n) + \int_0^1 e^{A_c\tau} B_c \begin{bmatrix} u_c(n+1-\tau) \\ v_c(n+1-\tau) \end{bmatrix} d\tau \\
 &= e^{A_c} x_c(n) + \int_0^1 e^{A_c\tau} B_{c1} u_c(n+1-\tau) d\tau + \int_0^1 e^{A_c\tau} B_{c2} v_c(n+1-\tau) d\tau
 \end{aligned} \tag{4.38}$$

where

$$B_c = \begin{bmatrix} B_{c1} & B_{c2} \end{bmatrix} \tag{4.39}$$

The first integral in Eq. (4.38) represents the filtering operation on u_c , which precedes the sampling operation. As stated at the outset, this filtering does not impact the stability of the modulator and for our present purposes can be neglected. For the second integral, one must know the waveshape of the v_c signal. If the waveform is perfectly rectangular, as shown in Figure 4.21(a), $v_c(t) = v(n)$ for $n < t \leq n+1$, where n is an integer. Thus

$$\int_0^1 e^{A_c\tau} B_{c2} v_c(n+1-\tau) d\tau = A_c^{-1} (e^{A_c} - I) B_{c2} v(n) \tag{4.40}$$

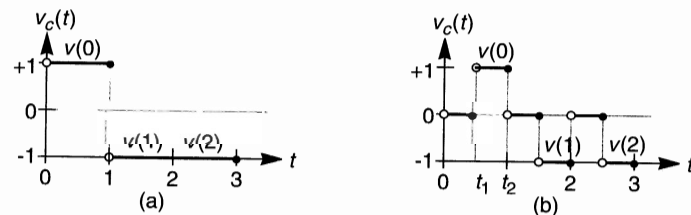


Figure 4.21 Example DAC feedback waveforms: (a) a nonreturn-to-zero (NRZ) waveform and (b) a return-to-zero (RZ) waveform.

As a result, Eqs. (4.35) and (4.36) describe systems whose samples will be identical provided

$$A = e^{A_c} \tag{4.41}$$

and

$$B_2 = A_c^{-1} (A - I) B_{c2} \tag{4.42}$$

In a similar manner, the inverse transformation

$$A_c = \log A \tag{4.43}$$

$$B_{c2} = (A - I)^{-1} A_c B_2 \tag{4.44}$$

converts a discrete-time modulator to its continuous-time equivalent, thereby facilitating both analysis and synthesis of continuous-time modulators.

In the context of $\Delta\Sigma$ modulators, degenerate cases (such as $A - I$ or A_c singular) are commonplace, but a discrete-continuous transformation is nonetheless possible. For example, the MATLAB [18] functions `d2c`, `d2cm`, `c2d`, and `c2dm` convert between a discrete-time system and its continuous-time counterpart in all cases, provided the input waveforms are of the “zero-order hold” variety illustrated in Figure 4.21(a). For example, in the first-order modulator the loop filter is

$$L(z) = \frac{H(z) - 1}{H(z)} = \frac{-z^{-1}}{1 - z^{-1}} \tag{4.45}$$

The MATLAB command `[Lcnum, Lcden] = d2cm([0, -1], [1, -1], 1)` then yields that the continuous-time loop filter must be

$$L_c(s) = -\frac{1}{s} \tag{4.46}$$

Note that the `d2cm` command is able to operate directly on the loop transfer function (the conversion to a state-space formulation is done internally) and that the command also allows time scaling by supplying the sampling period as the last argument.

The second-order modulator is likewise easy to convert to continuous time. The loop filter for the discrete version of the modulator is

$$L(z) = \frac{H(z) - 1}{H(z)} = \frac{-2z^{-1} + z^{-2}}{1 - 2z^{-1} + z^{-2}} \tag{4.47}$$

and the command `[Lcnum, Lcden] = d2cm([0, -2, 1], [1, -2, 1], 1)` yields that the continuous-time loop filter is

$$L_c(s) = -\frac{1.5s + 1}{s^2} \tag{4.48}$$

If the DAC waveform is of the form shown in Figure 4.21(b), with $0 \leq t_1 < t_2 \leq 1$, Then a procedure identical to that described earlier yields that the continuous and discrete systems are equivalent provided.

$$A = e^{A_c} \quad \text{and} \quad B_2 = A_c^{-1} (e^{A_c(1-t_1)} - e^{A_c(1-t_2)}) B_{c2} \quad (4.49)$$

This value of B_2 is different from Eq.(4.42) by a factor of

$$(e^{A_c(1-t_1)} - e^{A_c(1-t_2)})(A - I)^{-1} = (e^{-A_c t_2} - e^{-A_c t_1})(e^{-A_c} - I)^{-1} \quad (4.50)$$

which suggests that the `d2cm` function could be used to do the bulk of the work, with this correction factor applied at the end. Note that the rearrangement of factors implied by this operation is possible since the various matrices commute.

4.6 NONLINEAR STABILIZATION TECHNIQUES

All high-order single-loop modulators are conditionally stable. In the context of traditional servo-control systems, conditional stability means that if a variable gain block is inserted into the loop and the gain is varied, stability is achieved only over a limited range of gains. Intuitively, one would expect that high gains would be more troublesome than low gains, but in fact the opposite is true. As demonstrated by the root-locus plot of Figure 4.4, the poles may venture outside of the unit circle if the gain becomes too low.

There are two conditions under which a conditionally stable modulator will oscillate; signal overload and power-on. We have already seen how the effective in-circuit gain of the comparator is signal dependent, and how large input signals may trigger sustained oscillations. It is often difficult in practice to limit the amplitude of the incoming signal. This is especially true of audio signals, which are highly unpredictable. Analog clipping circuits may be used to limit the input range, but it is difficult to design a circuit that does not cause an increase in distortion for “legal” input signals.

The second condition that causes oscillation, power-on, is best explained by a state-space view of stability. The state of an n^{th} -order modulator may be completely defined by n state variables. It is usually most convenient to define the state variables to be the integrator output voltages. If these state variables are set to a particular set of voltages and then released, the system will either return to a normal idling pattern or produce very large voltages indicative of oscillation. Therefore the entire multidimensional state space may be mapped into stable and unstable regions, in a manner similar to Figure 4.11. If the initial conditions of the integrators on power-up correspond to an unstable region of the state space, then oscillations will begin when the system is turned on.

This state-space view of stability leads to an effective method for ensuring global stability. The solution is simply to bound the state space by placing nonlinear limiting elements in parallel with each integrating capacitor. By limiting the value of each state variable independently, the system is constrained to operate in a stable region of the state space.

Another method of ensuring global stability is somewhat less elegant but very simple. Upon detection of oscillation, all integrator states may be reset to zero for a brief period and then released to resume normal operation. Sustained overloads will cause

repeated reset events, which sometimes lead to ugly waveforms at the output of the digital filter/decimator. This can be mitigated somewhat by resetting only the last few integrators in the forward loop.

There are two common ways of sensing instability. One is to look for integrator states above a certain value using a comparator and use this to trigger the reset circuitry. The other method is to look for long strings of 1's or 0's in the digital bit stream. A normally operating modulator usually has a maximum “run length” anywhere from 6 to 10 bits in a row, whereas oscillations often produce run lengths of up to 100 cycles. A threshold of 12–32 same-valued bits is usually sufficient to detect oscillation.

Practical tips for designing effective stabilization circuits will be discussed in Chapter 5.

4.7 CONCLUSION

The first- and second-order modulators are fairly well understood at the theoretical level and the third-order modulator has also recently received attention. The describing function method predicts the stability of high-order modulators very well. However, no guarantees can be made since the method is approximate. Computer simulations of the nonlinear difference equations are still required, and even those may blow up after millions of cycles have passed. Until an iron-clad, but not too conservative, theory for modulator stability is found, the approximation methods presented here, coupled with the judgment of the designer, will have to suffice.

ACKNOWLEDGMENTS

The timely help of Bo Zhang and Montgomery Goodson with numerous equations and figures is much appreciated. Both the numerical power of *Matlab* [18] and the analytical power of *Mathematica* [19] have likewise been indispensable.

REFERENCES

- [1] R. W. Adams, “Design and implementation of an audio 18-bit analog-to-digital converter using oversampling techniques,” *J. Audio Eng. Soc.*, vol. 34, pp. 153–166, March 1986.
- [2] R. W. Adams, P. F. Ferguson, A. Ganesan, S. Vincelette, A. Volpe, and R. Libert, “Theory and practical implementation of a fifth-order sigma-delta A/D converter,” *J. Audio Eng. Soc.*, vol. 39, pp. 515–528, July 1991.
- [3] T. Ritoniemi, T. Karema, and H. Tenhunen, “The design of stable high order 1-bit sigma-delta modulators,” *Proc. 1990 IEEE Int. Symp. Circuits Sys.*, vol. 4, pp. 3267–3270, May 1990.
- [4] S. H. Ardalan and J. J. Paulos, “Analysis of nonlinear behavior in delta-sigma modulators,” *IEEE Trans. Circuits Sys.*, vol. 34, pp. 593–603, June 1987.
- [5] R. Schreier, “Destabilizing limit cycles in delta-sigma modulators with chaos,” *Proc. IEEE Int. Symp. Circuits Sys.*, vol. 2, pp. 1369–1372, May 1993.

- [6] N. He, F. Kuhlman, and A. Buzo, "Multiloop sigma-delta quantization," *IEEE Trans. Inf. Theory*, vol. 38, no. 3, pp. 1015–1028, May 1992.
- [7] R. Schreier and Y. Yang, "Stability tests for single-bit sigma-delta modulators with second-order FIR noise transfer functions," *Proc. IEEE Int. Symp. Circuits Sys.*, vol. 3, pp. 1316–1319, May 1992.
- [8] S. Hein and A. Zakhhor, "On the stability of interpolative sigma delta modulators," *Proc. 1991 IEEE Int. Symp. Circuits Sys.*, vol. 3, pp. 1621–1624, June 1991.
- [9] S. Hein, K. M. Ibrahim, and A. Zakhhor, "New properties of sigma-delta modulators with DC inputs," *IEEE Trans. Commun.*, vol. 40, no. 8, pp. 1375–1387, Aug. 1992.
- [10] H. Wang, " $\Sigma\Delta$ modulation from the perspective of nonlinear dynamics," *Proc. 1992 IEEE Int. Symp. Circuits Sys.*, vol. 3, pp. 1296–1299, May 1992.
- [11] H. Wang, "A geometric view of $\Sigma\Delta$ modulations," *IEEE Trans. Circuits Sys.-II: Analog Digital Signal Proc.*, vol. 39, no. 2, pp. 402–405, June 1992.
- [12] S. Hein and A. Zakhhor, "On the stability of sigma delta modulators," *IEEE Trans. Signal Proc.*, vol. 41, no. 7, pp. 2322–2348, July 1993.
- [13] H. Wang, "On the stability of third-order sigma-delta modulation," *Proc. 1993 IEEE Int. Symp. Circuits Sys.*, vol. 2, pp. 1377–1380, May 1993.
- [14] H. Wang, "A study of sigma-delta modulations as dynamical systems," Ph.D. Dissertation, Columbia University, June 1993.
- [15] W. L. Lee, "A novel higher order interpolative modulator topology for high resolution oversampling A/D converters," Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1987.
- [16] W. L. Lee and C. G. Sodini, "A topology for higher order interpolative coders," *Proc. 1987 IEEE Int. Symp. Circuits Sys.*, vol. 4, pp. 459–462, May 1987.
- [17] R. Schreier, "An empirical study of high-order single-bit delta-sigma modulators," *IEEE Trans. Circuits Sys. II*, vol. 40, no. 8, pp. 461–466, Aug. 1993.
- [18] The MathWorks, Inc., *Matlab*, Version 3.5, The MathWorks, Inc., Natick, MA, 1992.
- [19] Wolfram Research, Inc., *Mathematica*, Version 2.0, Wolfram Research, Inc., Champaign, IL, 1992.
- [20] B. P. Agrawal and K. Shenoi, "Design methodology of $\Sigma\Delta$ M," *IEEE Trans. Commun.*, vol. 31, pp. 360–370, March 1983.
- [21] O. Feely and L. O. Chua, "Nonlinear dynamics of a class of analog-to-digital converters," *Int. J. Bifurcation Chaos*, vol. 2, no. 2, pp. 325–340, June 1992.
- [22] O. Feely and L. O. Chua, "The effect of integrator leak in Σ - Δ modulation," *IEEE Trans. Circuits Sys.*, vol. 38, no. 11, pp. 1293–1305, Nov. 1991.
- [23] A. Rachid, "Positively invariant polyhedral sets for uncertain discrete time systems," *Control Theory Adv. Technol.*, vol. 7, no. 1, pp. 191–200, March 1991.

Robert W. Adams

The Design of High-Order Single-Bit $\Delta\Sigma$ ADCs

5.1 INTRODUCTION

Until 1987 [1], it was widely believed that single-bit $\Delta\Sigma$ modulators with loop filters of order greater than 2 were impossible to stabilize. Indeed, any casual attempt to build such a modulator will result in impressive oscillations, which tend to discourage the faint-hearted from further experimentation. This belief that high-order modulators could not be stabilized continued even after several commercial examples of such modulators were in production and appeared in everyday equipment built for the consumer audio market.

Chapter 4 described how to design a loop filter that results in a stable single-loop high-order modulator. Low-order and cascaded modulators are inherently stable, although under near-overload conditions they may be perilously close to oscillation.

In this chapter we will examine many practical aspects of high-order $\Delta\Sigma$ designs, including loop filter topologies, maximum safe input range, prevention of idling tones, and common commercial design practices. Since most commercial $\Delta\Sigma$ ADCs use switched-capacitor circuitry, we will focus on those topologies that can be implemented using switched-capacitor circuits. A design example of a fourth-order modulator and a description of a commercially available fifth-order modulator will be given.

It is interesting to speculate on the history of high-order modulators. To the author's knowledge, the first working breadboard of a single-bit high-order modulator was produced at Philips research labs in the late 1970s. In private conversations with the researchers, it is evident that stability was achieved somewhat accidentally by the way that the integrator op-amps behaved under overload conditions. As will be shown later, limiting the range of the state variables in the loop filter is one way to ensure global stability in a high-order loop. The Harris patent [2] is perhaps the earliest written disclosure of a single-loop higher order modulator stabilized using state-variable clamping techniques.

5.2 MOTIVATION FOR USING HIGH-ORDER SINGLE-BIT LOOPS

The world of noise-shaping converters can be roughly divided into the following camps: single-bit single-loop low-order designs, single-bit single-loop high-order designs, multi-loop cascaded designs with feedforward error cancellation, and multibit noise shapers [3] (both low and high order). The advantages and disadvantages of these topologies are summarized in Table 5.1.

Single-bit high-order designs are most attractive whenever high SNR, simple circuit design, and good idle-tone performance are important. The major obstacle to overcome is the issue of stability, but once a methodology has been established, designing a new loop

TABLE 5.1 COMPARISON OF MODULATOR ARCHITECTURES

Modulator type	Advantages	Disadvantages
Low-order single-loop single-bit	• Guaranteed stability. • Simple loop filter design. • Simple circuit design. • Input range may use almost the full range of 1's densities.	• Low SNR (except for high oversampling ratios). • More prone to idling tones (dither may help).
High-order single-loop single-bit	• High SNR for modest oversampling ratios. • Less prone to idling tones. • Simple circuit design.	• Difficult loop filter design. • Stability is signal dependent. • Maximum input range must be restricted to ensure stability.
Multiloop cascade	• High SNR for modest oversampling ratios. • Stability guaranteed. • Maximum input range almost equal to the full range of 1's densities.	• Requires near-perfect matching between analog integrator and digital differentiator. Complex switched-capacitor circuits are required to ensure matching. • Imperfect matching may result in leakage of tones into baseband. • Decimation filter must allow for multibit inputs.
Multibit	• High SNR for fairly low oversampling ratios. • Stability much easier to achieve for high-order loops. • Extra quantization levels allow for large dither signals at the quantizer input, eliminating idling tones.	• Imperfect matching of levels (in D/A half of quantizer) results in imperfect dc transfer function (integral nonlinearity errors). Autocalibration or dynamic element matching may help. • Decimation filter must allow for multibit inputs. • More complex circuit design.

filter is a simple matter of running a few programs and checking the result with a discrete-time simulator.

For a given oversampling ratio, a high-order modulator is capable of much greater signal-to-noise ratios than a simple second-order loop. This improvement is a function of the oversampling ratio; high oversampling ratios benefit the most from high-order loop filters, whereas low oversampling ratios (e.g., <16) do not show as much improvement when high-order loop filters are used. (See Figure 4.14.)

If the bandwidth requirements of the ADC are modest (say, less than 5 kHz) and master clocks are available in the 3–10-MHz range, then it is usually possible to use a conventional second-order loop in the design. A second-order $\Delta\Sigma$ converter running with a high oversampling ratio (e.g., 256 or more) can achieve theoretical SNR figures in excess of 100 dB, although the idle tones produced by an undithered second-order modulator may prove unacceptable in some applications. High-order loops become attractive when a high SNR is required for modest oversampling ratios (e.g., 64). The ADCs designed for audio signals usually fall into this category. With a decimated sample rate of 48 kHz, an oversampling ratio of 64 calls for a master clock of 3.072 MHz. The same converter implemented with a second-order loop might require a master clock frequency of 12.288 MHz (256 times oversampling) to achieve the same SNR. Op-amps that achieve full settling to a high degree of accuracy are much harder to design at 12 MHz than at 3 MHz.

In the literature it is common to find graphs of SNR versus the oversampling ratio for various loop filter orders. These graphs should be used with great caution. In many cases, these results are derived from the linear model with a noise-shaping transfer function of

$$H(z) = (1 - z^{-1})^n \quad (5.1)$$

where n is the order of the differentiation.

It is important to realize that the use of this noise-shaping transfer function does not result in stable systems for loop filters of order greater than 2, and therefore the results are meaningless unless multibit quantizers are used. If we allow any arbitrary n^{th} -order noise-shaping transfer function, then the number of different stable noise-shaping transfer functions that can be designed is infinite. Therefore, it is unwise to trust the results of another designer unless the coefficients of the loop filter are exactly given and the results come from a discrete-time simulation and not a linear model. The only modulator for which there are no degrees of freedom is the lowly first-order modulator.

5.3 DESIGN CHOICES: SC OR ACTIVE-RC?

Perhaps the first choice that must be made in designing any $\Delta\Sigma$ modulator is a choice between a switched-capacitor (SC) implementation and a conventional active-RC (continuous-time) design. In general, most integrated circuit (IC) implementations of $\Delta\Sigma$ ADCs use SC circuits, whereas most system-level or hybrid implementations use active-RC circuits. The reasons for this are summarized in Table 5.2.

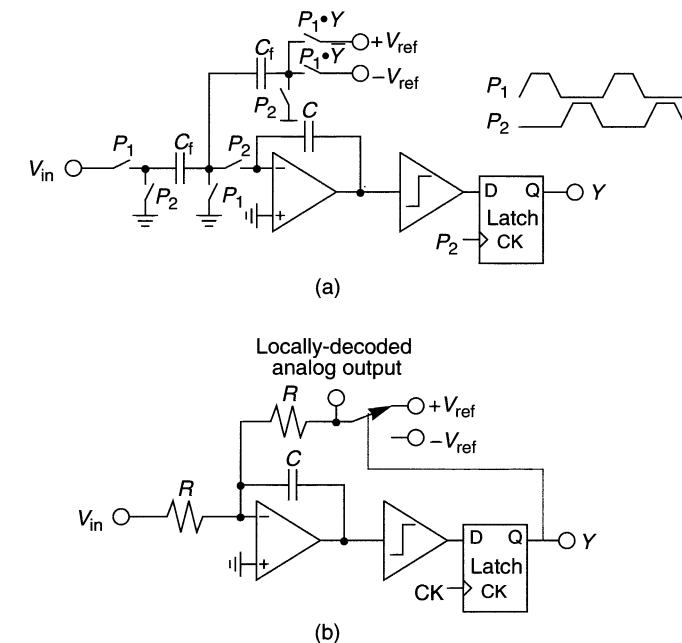
Figure 5.1 shows a comparison between the two design styles. In Figure 5.1(a), a simple SC circuit is shown with a two-phase clock. On phase 1, the input is sampled, and the comparator decision is made. On phase 2, the input is delivered to the integrator summing junction, along with the result of the comparator decision. A “1” comparator decision results in a charge of $C_f V_{\text{ref}}$ delivered to the integrator, whereas a “0” decision results

TABLE 5.2 COMPARISON OF SC AND RC MODULATOR REALIZATIONS

Circuit style	Advantages	Disadvantages
Switched capacitor	- Easily simulated. - Compatible with VLSI CMOS process (extra poly layer desirable to make small-area linear caps). - Insensitive to clock jitter as long as full settling occurs. - Insensitive to exact shape of op-amp settling waveform as long as full settling occurs. - Pole-zero locations are set by capacitor ratios, which are highly accurate.	- Large capacitors required for high SNR (kT/C noise limit). - Switched-capacitor circuits are true samplers, potentially causing aliasing of out-of-band noise. They are thus more prone to picking up digital noise. - Large spike currents drawn by capacitors are hard to drive from external sources (RC isolation circuits required). - Very difficult to prototype (typical capacitor values are less than 1 pF and are easily swamped by parasitics on a breadboard).
Continuous time	- Easy to breadboard. - Less prone to pick up digital noise (no true input samplers are used). - Easy to drive from external sources; no switched-capacitor current pulses. - SNR is not limited by cap size.	- Not as compatible with a simple complementary metal–oxide–semiconductor (CMOS) process. Needs large capacitors, linear high-value resistors, low-noise op-amps. - Accurate RC time constants not possible for monolithic designs without laser trimming. - SNR degraded by nonideal comparator feedback signal. Sensitive to jitter, noise, and switching characteristics of 1-bit feedback waveform. - Loop filter does not scale with clock frequency. - Op-amps must remain linear at all times. It is <i>not</i> just the settled value that counts. - Discrete-time simulation more difficult.

in a charge of $-C_f V_{\text{ref}}$ being delivered. In many cases, the required inversion of the reference signal can be accomplished by appropriate phasing of the switches using only a single reference.

Figure 5.1(b) shows a simple continuous-time design. The output of the comparator is latched on every clock pulse, and the latched comparator decision is used to apply a $+V_{\text{ref}}$ or $-V_{\text{ref}}$ voltage to a resistor connected to the integrator summing junction. This


Figure 5.1 First-order switched-capacitor (a) and continuous-time (b) $\Delta\Sigma$ modulators.

voltage is continuously applied to the integrator during the entire clock period. Note that the amount of charge delivered to the integrator is a function of the width of the clock pulse, the rise and fall time of the voltage feedback waveform, and, in the case of unequal rise and fall times, the previous bit decision. The use of a “return-to-zero” scheme can in theory eliminate the effects of previous bit decisions and rise–fall mismatches, but it increases the sensitivity to jitter [4].

The sensitivity to these effects is quite severe; a typical audio converter with a 20-kHz bandwidth and an oversampling ratio of 128:1 might require jitter less than 20 ps peak to peak to achieve > 90 dB of SNR! In one commercial example of a continuous-time loop, a multibit approach was used to add dynamic range “on top” of the jitter-limited noise floor [4]. This allowed an SNR of greater than 105 dB to be achieved, even though the theoretical SNR of the loop was much higher than this.

One common mistake made in continuous-time design is to assume that the V_{ref} feedback signal at the “locally decoded” analog output point can be measured to determine the SNR of the loop. However, in the presence of nonideal effects such as jitter, the effective value of a 1 or 0 comparator decision is not constant. Since the loop is continuous time, the feedback will ensure that the analog version of the 1-bit feedback signal is a faithful representation of the input, regardless of nonideal clocking effects. Unfortunately, the 1-bit digital signal that is passed to the decimator assumes the value of a 1 or 0 is invariant. It is very common to achieve high SNR when a measurement at the locally decoded output point is made (e.g., with an analog spectrum analyzer), while the same measurement made at the decimated digital output exhibits excess noise.

5.4 STABILITY

Chapter 4 gives a complete procedure for the design of stable noise transfer functions based on establishing a relationship between the statistics of the comparator input signal and the comparator “gain” as seen by the loop at low frequencies. We will assume throughout this chapter that a stable NTF has been designed and now needs to be implemented.

One of the most challenging aspects of stability analysis lies in establishing a “safe” input range [5, 6]. The relationship between the stable dc input range and the desired NTF was established in Chapter 4, but dynamic signals often behave very differently from dc signals when it comes to stability.

5.4.1 Stability and the Uncontrolled Input Signal: A Practical Guide to Safe Operation

The stability analysis given in Chapter 4 shows that stable operation of a high-order loop is possible only over a restricted range of dc input signals. At large dc input levels, the effective comparator gain falls, causing instability to occur. The stable range is related to the choice of the NTF; NTFs that are not very aggressive in the context of noise shaping have larger usable input ranges (e.g., see Figure 4.17). For systems that use Butterworth pole alignments (a good compromise between aggressive noise-shaping and stable operation), the stable input range is typically between 60 and 80% of the comparator feedback voltage. It is prudent to define the full-scale input to be less than the value at which instability actually occurs, as it is impossible to exhaustively simulate all the possible inputs that may be applied. Typically, a converter that is dc stable over an 80% bit density range will have a defined maximum input range of 65–70% of the comparator feedback voltage. This “safety factor” depends to a large extent on the Q of the poles in the NTF as well as on how well-behaved the input is.

5.4.2 Transient Input Signals and Stability: The Case for Mild Prefiltering

The stability analysis based on the comparator gain argument is approximately valid for dc input signals and for sinusoids that are low in frequency compared with the poles of the NTF. For example, a typical audio converter with a fourth-order loop filter might use a Butterworth NTF with a high-pass frequency of about 150 kHz and a clock frequency of 3 MHz, compared with a signal bandwidth of 20 kHz.

In real life, it is impossible to ensure that the input signal is completely band limited, so one must consider what happens when transient or other wide-band signals are applied to the input. From our previous analysis of stability, we know that large comparator input signals cause reduced comparator gain and hence result in possible instability. We can then ask the question, “what input signal will cause the largest input to the comparator?” It is well known that the peak value at the output of a filter with a given impulse response is equal to the L_1 norm (sum of absolute values) [7] of the impulse response of the filter, assuming that the input is limited in amplitude to ± 1 . The input signal, $u(n)_{\text{worst-case}}$, that produces this peak output is simply a “sliced” version of the loop filter’s impulse

response, $l(n)$:

$$u(n)_{\text{worst-case}} = \text{sgn}[l(n)] \quad (5.2)$$

where

$$\text{sgn}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$$

If we use the impulse response of the signal transfer function (using the linear model), then we can obtain the input signal that causes the largest peak value at the comparator input. This input signal is simply a signal that alternates between the positive and negative full-scale input of the converter depending on the sign of the impulse response. Note that the term “full scale” here means the full-scale range defined at the input terminals, which results in less than 100% 1’s density at the modulator output.

The comparator gain argument is a statistical argument that applies over the long term, and therefore it does not adequately address the situation where a single large comparator input occurs. Chapter 4 derived a more rigorous bound on the loop filter state space for stable operation. In any case, it is apparent that repeated large values at the comparator input are more dangerous than single-cycle overloads. In the experience of the author, a good worst-case steady-state input signal is a square wave with amplitudes $\pm V_{\text{max}}$ and a frequency equal to that of the highest- Q pole pair in the NTF. This tends to be roughly the same as the worst-case L_1 norm input signal derived above; only it continues forever. Note that this signal is always well outside of the normal signal bandwidth.

With no input band limiting, this worst-case input signal will result in instability occurring at much lower amplitude levels than for the case of normal in-band signals. Fortunately there is a simple and effective solution to this problem: prefilter the input using a simple first- or second-order low-pass filter. Since the frequency of the highest- Q pole is normally 5–10 times that of the useful signal bandwidth, a simple filter can effectively reduce the amplitude of out-of-band signals at the pole frequency without significantly affecting the in-band frequency response. Exhaustive simulations with various proposed worst-case input signals have shown that the stable range of a prefiltered $\Delta\Sigma$ loop is very close to the stable range of the same loop for dc signals.

If the NTF uses very high Q poles resulting in a large response peak at the pole frequency, then more aggressive prefiltering may be required to prevent out-of-band signals from causing instability.

If the input is highly band limited, simulations show that only a small “safety factor” needs to be observed. For example, if a particular modulator becomes unstable for dc values that exceed 1 V, it is unlikely that a band-limited real-life input signal will cause instability at values less than 1 V. If, however, the input signal is not strongly band limited, the system may become unstable for signals considerably lower than 1 V. Bear in mind that these simulations need to be run for millions of modulator clock cycles; it is not at all uncommon to find a modulator that becomes unstable with a large dc input signal after many thousands of clock cycles.

5.5 CHOICES FOR THE NTF

Section 4 of Chapter 4 described a procedure for designing high-order NTFs. In this section, we will elaborate on how common choices for the NTF family impact the operation of a single-loop $\Delta\Sigma$ modulator.

5.5.1 n^{th} -Order Pure Differentiation

Let us start with the simplest of noise-shaping functions, pure n^{th} -order differentiators:

$$H(z) = (1 - z^{-1})^n \quad (5.3)$$

These functions meet the causality requirement, established in Chapter 4, that the first value of the impulse response must be 1. The magnitude response of $H(z)$ for various values of n is shown in Figure 5.2. Note that at high values of n , we obtain more effective suppression of quantization noise at low frequencies but have more gain at high frequencies.

Figure 5.3 shows a histogram of the quantization levels (used for idling only) for various loop orders, assuming an infinite quantizer [8]. These plots are obtained by running a loop for thousands of cycles and counting the number of occurrences of each quantization level in the loop quantizer. Note that the number of levels used increases at a rate proportional to 2^n , which makes sense since the high-frequency noise-shaping gain also increases at this rate. From our previous discussion of comparator gain versus assumed linear model gain, it appears that this noise-shaping function is not a good candidate for loops with 1-bit quantizers. As a rule of thumb, any high-order loop that uses many levels for idling with an infinite quantizer will oscillate when the extra levels it needs are taken away and a 1-bit quantizer is used. The reason for this can be explained using the comparator gain argument outlined in Section 4.2.2. The fact that a loop uses many levels for idling implies that the random idling-waveform input to the quantizer is large. If we use only a 1-bit quantizer, the input to the comparator will be large, causing its in-circuit gain to be very small. This small gain causes oscillations to occur.

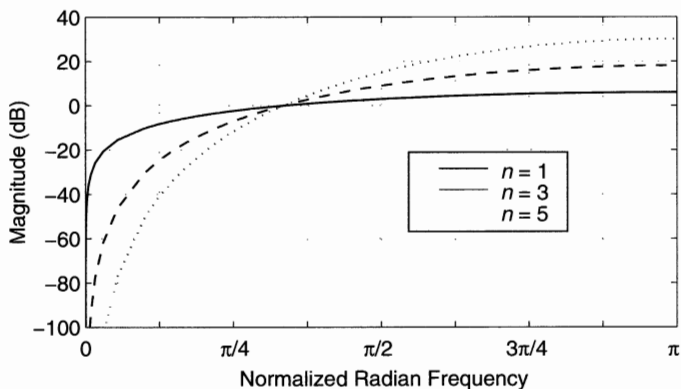


Figure 5.2 Noise transfer functions of the form $H(z) = (1 - z^{-1})^n$.

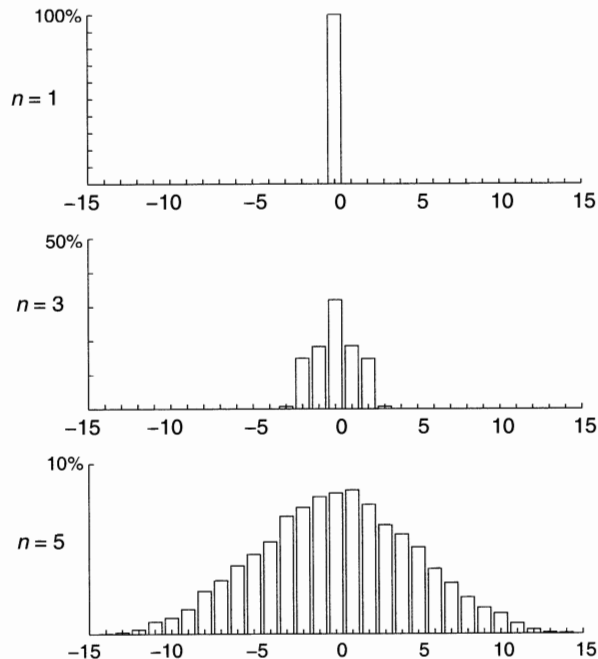


Figure 5.3 Histograms for n^{th} -order differentiation NTF loops, assuming an infinite quantizer.

5.5.2 Butterworth High-Pass Response

The problem encountered in the previous noise-shaping function was the large high-frequency noise-shaping gain for large n . The idling waveform at the comparator input becomes very large, resulting in low comparator gain and hence instability. We can modify the pure differentiating response by introducing poles into $H(z)$, as shown below:

$$H(z) = \frac{(z-1)^n}{D(z)} \quad (5.4)$$

where $H(z)$ is an n th-order polynomial with a leading coefficient of 1.

The purpose of adding $D(z)$ is to flatten the high-frequency portion of $H(z)$, as shown in Figure 5.4. Now $H(z)$ is simply an n th-order high-pass function. With the correct choice of $D(z)$, a Butterworth alignment of the poles can be obtained, resulting in a maximally flat high-frequency region of $H(z)$. With a correct choice of cutoff frequency, it is possible to meet both the 3 dB-gain rule and the causality rule, as outlined earlier. Note that to meet both requirements, there are no degrees of freedom; once a Butterworth filter is chosen, there is one and only one choice of cutoff frequency that meets both conditions.

The Butterworth alignment of the NTF is often a good choice and is commonly used in commercial products. One reason for this is that the poles are relatively low Q , and therefore the Butterworth alignment tends to be less susceptible to oscillations caused by input signals that are at the same frequency as the poles.

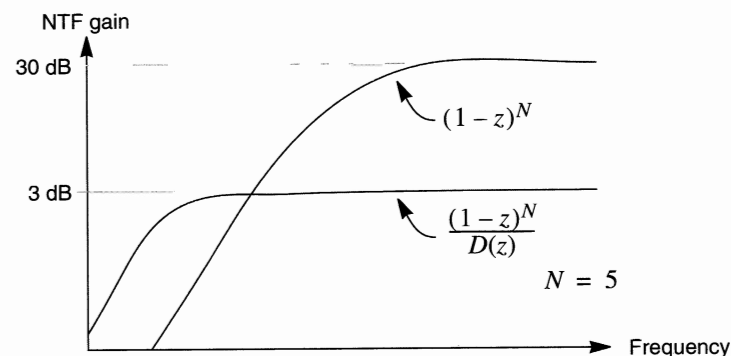


Figure 5.4 Modifying the NTF to reduce high-frequency gain.

5.5.3 Complex Zeros on the Unit Circle (Inverse Chebyshev)

The Butterworth high-pass response of the previous section can be modified to move the real stopband zeros at the $(1, j0)$ point out on the unit circle to produce nulls in the NTF at frequencies other than dc. Compared with the Butterworth alignment, improved signal-to-noise ratio can be obtained, as the filter exhibits greater attenuation over the desired signal passband range.

For a given order, the designer must decide how many complex zero pairs will be moved to nonzero frequencies on the unit circle. For example, a fifth-order system could have three real zeros and one complex pair or it could have one real zero and two complex pairs. Generally, using the largest number of complex pairs possible for a given order results in the best noise-shaping characteristic, although there may be circuit-related reasons for choosing otherwise.

The same iterative design methodology outlined previously can be used here as well, the only differences being that the commercial design program should be instructed to design an inverse Chebyshev filter instead of a Butterworth filter and that the stopband edge should be entered as the edge of the desired signal passband [9]. Note that the design program will result in a filter that is equiripple in the stopband. This is not the optimum solution in terms of the total integrated noise over the audio band, and adjusting the complex zero locations to those listed in Table 4.1 may yield several decibels of improvement in the SNR. It is also advisable to do this optimization in conjunction with the known frequency response of the decimation filter, as the noise from the noise shaper rises at a very fast rate at frequencies just beyond the last complex pair frequency, which is typically quite close to the decimator cutoff frequency.

5.6 COMPARISON OF LOOP TOPOLOGIES

There are as many architectures for building $\Delta\Sigma$ modulators as there are ways to build a low-pass filter. In this section we will focus on the most common loop topologies.

All $\Delta\Sigma$ feedback topologies may be characterized by two transfer functions: the NTF and the STF (signal transfer function). The NTF determines to what extent the quantiza-

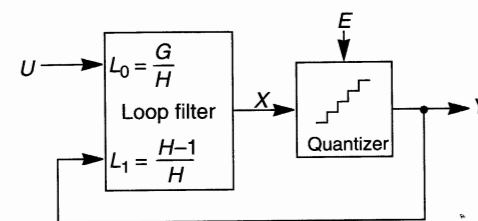


Figure 5.5 A universal $\Delta\Sigma$ structure.

tion noise is reduced in a given bandwidth and hence determines the overall SNR of the converter. Depending on the chosen architecture, it may or may not be possible to independently specify the STF. Normally the desired STF is flat over the band of interest.

From a circuit design standpoint, it is desirable to use integrators as the fundamental active building block. This allows switched-capacitor circuits to be designed in a parasitic-insensitive manner, with the integrator virtual ground receiving the charge stored on various capacitors in the circuit. Moreover, as we shall show shortly, loop filters built with integrators result in high-quality NTF notches.

In general, it is possible to describe all single-quantizer loops with the universal architecture shown in Figure 5.5. From this equivalent topology we easily see that the NTF and STF are

$$H(z) = \frac{1}{1 + L_1(z)} \quad (5.5)$$

and

$$G(z) = \frac{L_0(z)}{1 + L_1(z)} \quad (5.6)$$

respectively.

Given a particular topology, the loop filters $L_0(z)$ and $L_1(z)$ can be expressed as functions of the loop parameters. Matching these expressions to Eqs. (5.5) and (5.6) yields the desired parameters. For loops of high order, the algebra may become very messy. In some cases nonlinear equation solvers may be used to find the coefficients without having to solve the equations manually. Alternatively, symbolic equation solvers may be used.

In practice, $L_1(z)$ has high gain in the band of interest and is thus responsible for attenuating the quantization noise. Specifically, the poles of $L_1(z)$ are the zeros of the NTF. Also note that both the NTF and the STF generally share the same poles [the roots of $1 + L_1(z) = 0$], unless pole-zero cancellation occurs by judicious choice of $L_0(z)$.

In the previous chapter the signal-dependent gain of the loop quantizer was analyzed, and from this result we can observe that Eqs. (5.5) and (5.6) should be taken as approximations only. Any attempt to use high- Q poles (poles very close to the unit circle) for either the NTF or the STF will result in potential instability, or at the very least large variations in response shape as a function of the input signal.

The following circuit topologies have been successfully used in $\Delta\Sigma$ modulator design. This list is by no means exhaustive, and many other loops are possible, especially if one allows the use of continuous-time circuits:

1. Chain of integrators with weighted feedforward summation [5–6, 9].
2. Chain of integrators with feedforward summation and local resonator feedbacks [6].

3. Chain of integrators with distributed feedback [5, 10].
4. Chain of integrators with distributed feedback and distributed feedforward input paths [5, 9].
5. Chain of integrators with distributed feedback, distributed feedforward input paths, and local resonator feedbacks [5].
6. Error feedback only (normally only used for digital implementations of $\Delta\Sigma$ modulators) [11].

We will now give the basic design equations for each type of loop as well as the advantages and disadvantages for each type.

5.6.1 Chain of Integrators with Weighted Feedforward Summation

For the topology shown in Figure 5.6, the loop filters are essentially identical:

$$L_0(z) = -L_1(z) = \frac{a_1}{z-1} + \frac{a_2}{(z-1)^2} + \frac{a_3}{(z-1)^3} \dots \quad (5.7)$$

Note that as a result of this equivalence, once the loop filter is set for optimum noise shaping, the STF is fixed. Specifically,

$$G(z) = 1 - H(z) \quad (5.8)$$

The poles of $L_1(z)$ are restricted to dc ($z = 1, j0$) by the nature of the filter topology. Since the poles of $L_1(z)$ translate to the zeros of the NTF, the NTF must have all its zeros at dc. Inverse Chebyshev NTFs, which have stopband zeros at nonzero frequencies, cannot be

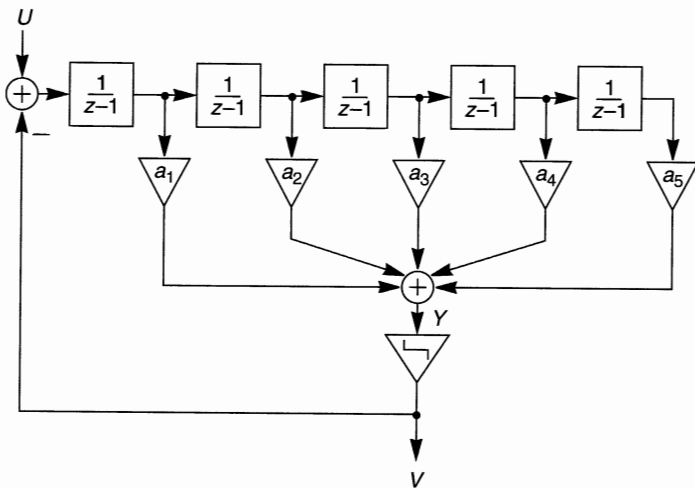


Figure 5.6 Chain of integrators with weighted feedforward summation.

implemented with this architecture. Butterworth high-pass filters can be implemented and are often used in practice. A drawback of a Butterworth NTF is that it produces a STF that invariably contains peaking at high frequencies. Thus, it is wise to either add a prefilter to the input to prevent inputs in this range from overloading the modulator or modify the NTF family such that flat STF result from Eq. (5.8).

5.6.2 Chain of Integrators with Feedforward Summation and Local Resonator Feedbacks

By adding a small negative-feedback term around pairs of integrators in the loop filter as shown in Figure 5.7, it is possible to move the open-loop poles (which become the NTF zeros when the loop is closed) away from dc along the unit circle. This causes the frequencies of infinite loop gain (and hence infinite noise attenuation) to be shifted away from dc to finite positive frequencies. The equation for a pair of integrators with feedback is

$$R(z) = \frac{z}{z^2 - (2 - \gamma)z + 1} \quad (5.9)$$

The poles have a radius of 1 and a frequency ω given by

$$\omega = a \cos\left(1 - \frac{\gamma}{2}\right) \approx \sqrt{\gamma} \text{ for } \gamma \ll 1 \quad (5.10)$$

Equations (5.9) and (5.10) were derived assuming that one of the integrators has a one-sample delay while the other does not, as shown in Figure 5.7. This can be accomplished in switched-capacitor circuits by correctly phasing the switches that form the negative-feedback network.

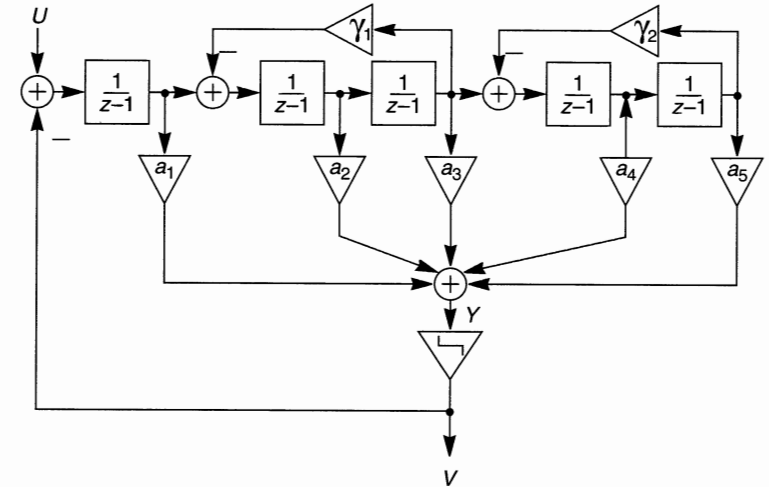


Figure 5.7 Chain of integrators with feedforward summation and local resonator feedbacks.

A slightly less effective resonator can be built by using feedback around pairs of integrators, while allowing both integrators to have a z^{-1} delay term in the numerator. In this case, the poles move on a vertical line from the $(1, j0)$ point away from the real axis and therefore do not exhibit infinite gain at the resonance frequency. For small displacements, however, the gain at resonance is quite large, and in most cases it is equally effective at removing quantization noise as the single-delay pair of integrators. The switched-capacitor implementation for the dual-delay integrator pair is somewhat simpler and does not require double op-amp settling. By allowing complex NTF zeros, it is possible to implement the inverse Chebyshev NTF described earlier. To solve for the coefficients, the transfer function of the loop filter $L_1(z)$ is determined, and then the NTF [derived from Eq. (5.5)] is equated to the desired NTF. This equation is then solved for the coefficients. This process assumes that the designer has chosen the number of resonators in $L_1(z)$ to equal the number of unit-circle zero pairs in the NTF.

5.6.3 Chain of Integrators with Distributed Feedback

The topology shown in Figure 5.8 is an “inverted” form of the previous topologies and may in fact be graphically derived from the feedforward topology by invoking the flow-graph inversion rule. The loop filters for this topology are

$$L_0(z) = \frac{b_1}{(z-1)^n} \quad (5.11)$$

$$-L_1(z) = \frac{a_1}{(z-1)^n} + \frac{a_2}{(z-1)^{n-1}} + \frac{a_3}{(z-1)^{n-2}} + \dots \quad (5.12)$$

Like the topology of Section 5.6.1, all NTF zeros are required to be at dc. Also, as before, the NTF determines the $L_1(z)$ loop filter uniquely and this in turn specifies the STF. In particular, if the NTF is of the form

$$H(z) = \frac{(z-1)^n}{D(z)} \quad (5.13)$$

then the STF is

$$G(z) = H(z)L_0(z) = \frac{b_1}{D(z)} \quad (5.14)$$

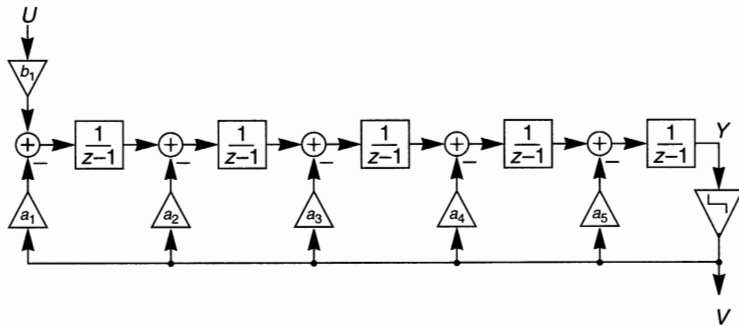


Figure 5.8 Chain of integrators with distributed feedback.

Therefore, if we design the NTF to be a classical high-pass filter such as a Butterworth filter, then the STF will be a low-pass filter with Butterworth poles. [The zeros at $z = (-1, j0)$ needed to make a true Butterworth filter are absent.] For example, a modulator designed with a Butterworth NTF with a high-pass corner frequency of 200 kHz would have a signal frequency response of a low-pass Butterworth filter with a 200-kHz cutoff. We conclude that it is possible to achieve nearly flat passband response with this topology by using a classical NTF filter shape such as a Butterworth. This STF flatness yields an advantage over the previous loops, as the STF does not contain significant peaking. In fact, the STF itself is a low-pass filter, which should result in improved stability when driven by large transient signals with significant out-of-band energy.

One drawback of this topology is that the integrator outputs contain significant amounts of the input signal as well as filtered quantization noise. This fact can be seen by considering what happens when a dc signal is applied to the input. Since each integrator has infinite gain at dc, the sum of the two input paths into each integrator must be zero to prevent any dc content from appearing at the integrator input. One of these paths is the 1-bit feedback signal multiplied by the feed-in coefficient. The other integrator input path is the output of the previous integrator in the loop. The previous integrator output must therefore contain a dc component to counteract the weighted 1-bit feedback.

Each integrator output thus contains a combination of filtered quantization noise and a low-frequency component equal to the input signal. In simulations it is apparent that the signal component is larger than the quantization noise component by a considerable amount. When designing switched-capacitor circuits, the output swings must be scaled by adjusting capacitor ratios to stay within the available supply range. Larger integrator swings therefore result in larger integrator feedback capacitors to keep the swings within the allowable range. Distributed-feedback circuits thus tend to be larger and more power hungry than circuits that use the feedforward topology.

5.6.4 Chain of Integrators with Distributed Feedback and Distributed Feedforward Inputs

The previous topology can be modified to allow a path from the input node to each integrator's summing junction, as shown in Figure 5.9. The addition of input feedforward paths allows a certain degree of independence in specifying the NTF and STF. The loop filters for this topology are given by

$$L_0(z) = \frac{b_1}{(z-1)^n} + \frac{b_2}{(z-1)^{n-1}} + \frac{b_3}{(z-1)^{n-2}} + \dots \quad (5.15)$$

and Eq. (5.12). Thus, if the NTF is given by Eq. (5.13), the STF is

$$G(z) = \frac{b_1 + b_2(z-1) + b_3(z-1)^2 + \dots}{D(z)} \quad (5.16)$$

The numerator of $G(z)$ is arbitrary, except that it has an order that is 1 less than the denominator order. The zeros of $G(z)$ may be placed in such a way as to cancel some of the poles, thus allowing the STF to have a lower roll-off rate than could otherwise be obtained. For example, if the NTF was designed to be a fifth-order Butterworth high-pass

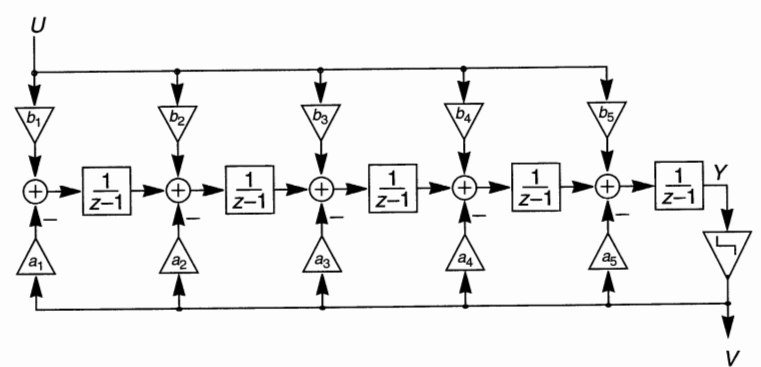


Figure 5.9 Chain of integrators with distributed feedback, distributed feedforward inputs.

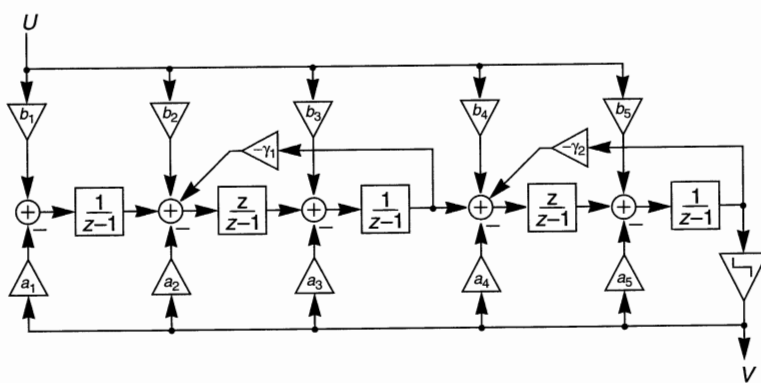


Figure 5.10 Chain of integrators with distributed feedback, distributed feedforward input paths and local resonator feedbacks.

function, the STF with no prefilter would be constrained to be essentially a fifth-order Butterworth low pass. With the addition of the feedforward coefficients, one might choose to cancel all the poles except for the real pole, resulting in an STF with a single-pole roll-off characteristic.

This topology may also be modified to include local resonator feedback paths around pairs of integrators, as shown in Figure 5.10. Again, this allows the placement of zeros on the unit circle at finite positive frequencies, allowing the NTF magnitude response to exhibit one or more notches in its frequency response.

5.6.5 Error Feedback Only

The topology of Figure 5.11 is often used in all-digital implementations of $\Delta\Sigma$ loops. It contains a single FIR (all-zero) feedback filter $F(z)$. While it may seem at first that this

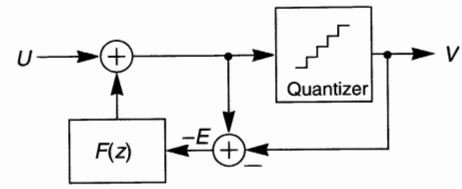


Figure 5.11 Error feedback only.

topology is fundamentally different than those we have discussed so far, it is in fact just another special case. The loop filters for this topology are given by

$$L_0(z) = \frac{1}{1 - F(z)} \tag{5.17}$$

$$L_1(z) = \frac{-F(z)}{1 - F(z)} \tag{5.18}$$

with the result that the NTF and STF are simply $H(z) = 1 - F(z)$ and $G(z) = 1$.

Equation (5.18) indicates that a first-order modulator may be obtained with only a single delay [$F(z) = z^{-1}$]. A second-order modulator may be obtained with a simple two-tap filter with weightings of 2 and -1 for the first and second tap, respectively. It is interesting to note that with the error feedback topology, the STF is flat and delay free, regardless of the loop filter used. This topology is often used when the NTF is an all-zero design, as this causes $F(z)$ to be a simple FIR (finite impulse response) filter. If the desired NTF contains poles (as all stable designs of order greater than 2 must), then the required feedback filter becomes an IIR (infinite impulse response) filter, and the advantage of simplicity is lost.

This topology is never used in ADC because analog FIR loop filters require accurate sample-and-hold circuits, which are more difficult to design than integrators. Also, any mismatches that occur in generating the error signal by subtracting quantizer input and output will seriously affect the performance of the loop. For this reason, only digital implementations of $\Delta\Sigma$ converters use this topology. In a digital implementation, this topology has some definite advantages. The difference between quantizer input and output can be done by simply taking the LSBs of a digital number; no hardware is required! Also, the feedback FIR filter is very easily accomplished, especially when the coefficients are simple powers of 2. An equivalent loop filter designed with conventional integrators requires several three-input adders, as each integrator must sum its input signal as well as its own feedback signal to generate a new output sample.

A more thorough discussion of all-digital noise-shaping loops is given in Chapter 10.

EXAMPLE: LOOP FILTER DESIGN

In this section a simple fourth-order example will be given, with the following design parameters.

1. NTF filter type: fourth-order Butterworth high pass.
2. Clock rate: 1 MHz.

The “cookbook” design procedure given in Chapter 4 was used to design this modulator. A commercial IIR filter design package (FDAS) was used to find the cutoff frequency of a fourth-order Butterworth filter whose impulse response begins with a 1 and whose passband gain is 3 dB. Since filter design packages normally design filters with unity-gain passbands, a filter with an initial impulse value

of 0.707 was found by experimenting with various cutoff frequencies of a fourth-order Butterworth filter. The resulting polynomial was scaled by 1/0.707, resulting in a 3-dB passband gain and an initial impulse value of 1. Note that this search for the correct high-pass corner is not completely random, as there exists a simple inverse relationship between the high-pass cutoff frequency and the value of the initial impulse response. If the initial impulse value for a given cutoff frequency is too low, for example, the high-pass corner frequency needs to be reduced to increase it. A successive-approximation type of search pattern usually gives the correct result after only a few iterations. The resulting NTF is

$$H(z) = \frac{(z-1)^4}{(z^2 - 1.5526z + 0.609)(z^2 - 1.753z + 0.817)} \quad (5.19)$$

Since this is a Butterworth and not a Chebyshev design, all the NTF zeros are placed at $z = 1, j0$ (dc). The cutoff frequency (−3-dB point) required to achieve the correct 3 dB of passband gain after impulse scaling was 50 kHz.

If we assume a single loop filter $L(z)$, we can solve for $L(z)$ in terms of the desired NTF using

$$L(z) = \frac{1 - H(z)}{H(z)} \quad (5.20)$$

Substituting Eq. (5.19) into Eq. (5.20), we find

$$L(z) = \frac{(z^2 - 1.5526z + 0.609)(z^2 - 1.753z + 0.817) - (z-1)^4}{(z-1)^4} \quad (5.21)$$

Note that the $(z-1)^4$ term in the denominator implies a structure that uses four cascaded integrators for the loop filter so the structure shown in Figure 5.6 could be used.

To derive the tap weights from each integrator, we must do a partial-fraction expansion on Eq. (5.21). To avoid messy algebra, we can use a nonlinear equation solver and equate Eq. (5.21) to the expression

$$L(z) = \frac{a_1}{z-1} + \frac{a_2}{(z-1)^2} + \frac{a_3}{(z-1)^3} + \frac{a_4}{(z-1)^4} \quad (5.22)$$

Alternatively, one can observe that Eq. (5.22) is linear in $a_1, \dots, a_4$ and thus form a system of linear equations by evaluating Eq. (5.22) at four points in the z -plane and equating these to the values of Eq. (5.21) evaluated at those same four points. The solution can then be found by invoking a numerical linear equation solver.

As a third alternative, one could derive the symbolic equations by hand or by using a symbolic equation solver.

The parameter values that result are given in Table 5.3. The rather large ratio of these coefficients can be reduced by adding gain to each integrator stage, as is commonly done in switched-capacitor circuits to avoid large capacitor ratios.

It was noted in Chapter 4 that an arbitrary gain placed in front of the comparator does not change the operation of the actual circuit, since the sign of the comparator input voltage is not

TABLE 5.3 LOOP FILTER COEFFICIENTS

Coefficient	a_1	a_2	a_3	a_4
Value	0.694025	0.230824	0.042539	0.0036151

affected by gain scaling. This implies that $a_1, \dots, a_4$ may be scaled by an arbitrary factor with no effect on the operation of the actual circuit, and therefore it is only necessary to observe the correct ratios between the coefficients.

At this point in the design, it is necessary to use a nonlinear difference equation simulator to establish the SNR, stable dc range, and other important characteristics of the loop. In addition to the loop itself, it is usually desirable to simulate the decimation filter at the same time. It is then possible to simulate the SNR of the loop in the time domain, using a running rms sum of the filtered bit stream. Commercial software simulators such as MATLAB may be used for this purpose, with the advantage that a graphical description of the loop using simple building blocks such as integrators and comparators may be used. Otherwise, custom C programs may easily be written to simulate the loop.

The loop designed here was simulated with the following results:

SNR: 85 dB over a 5-kHz bandwidth.

Stable dc range: −0.7 to +0.7 of full scale.

5.7 NONLINEAR GLOBAL STABILIZATION TECHNIQUES

In Chapter 4, two strategies for ensuring global stability were discussed: the state-variable clamping technique and the integrator reset technique. In this section we will delve more deeply into the practical implications of these techniques.

All high-order modulators become unstable for inputs that exceed certain bounds. This may also occur after power-on since op-amp integrators with arbitrary initial states may place the modulator in an unstable region of its state space.

The state-variable clamping technique is shown in Figure 5.12. A set of nonlinear elements is connected across each integrating capacitor to prevent large values from appearing at the integrator outputs. Note that placing a voltage limiter on the op-amp output rather than around the capacitor will not work, as the inverting input of the op-amp will cease to be a virtual ground, and extra charge from the previous stage can still accumulate on the capacitor.

One question that arises using this technique is, “how close do the limits have to be placed to the normal peak-to-peak operating range of the integrators?” The answer depends entirely on the NTF; high- Q poles are troublesome. For typical fourth- or fifth-order circuits, experiments suggest that the nonlinear elements should turn on at about

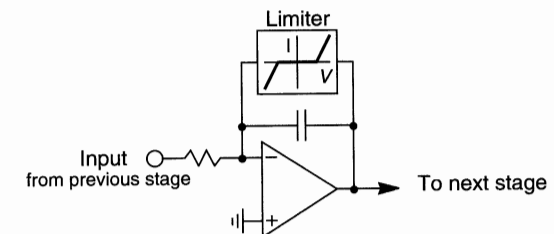


Figure 5.12 State variable stabilization method.

20–50% higher than the peak-to-peak integrator swings. In a multibit modulator, the effective step size is much smaller, and nonlinear limits may therefore be much larger than the normal integrator swing while still ensuring stability. This difference between multibit and 1-bit stabilization may be explained by recalling that instability is triggered by overloading the quantizer. In a multibit modulator the integrator swings can become large compared to the step size of the quantizer while still staying within the quantizer's saturation limits. In many cases the op-amp supply rails themselves may serve to limit the state variables to the stable region.

When building switched-capacitor modulators in a CMOS process, the integrator limiter approach is less attractive. A simple MOS device does not have a sharp enough voltage–current characteristic to make a good limiter, especially if the integrator swings are only ± 1 V or so. This results in the requirement for complex active limiters around each op-amp. For this reason, state-variable limiting is used more often in multibit $\Delta\Sigma$ converters, where the limiting characteristic need not be so sharp.

In single-bit designs, the integrator reset approach is the preferred solution to this problem. Usually the reset circuitry simply takes the form of a MOS switch connected across the integrating capacitor of each op-amp. The duration of the reset pulse is not critical as long as the integrating capacitor is fully discharged.

As discussed in Chapter 4, there are two common ways of sensing instability. One is to look for integrator states above a certain value using a comparator and use this to trigger the reset circuitry. The other method is to look for long strings of 1's or 0's in the digital bit stream. A normally operating modulator usually has a maximum “run length” anywhere from 6 to 10 bits in a row, whereas oscillations often produce run lengths of up to 100 cycles. A threshold of 12–32 same-valued bits is usually sufficient to detect oscillation.

In many cases the overloading signal will persist at the input for many thousands of clock cycles. A sequence of events then occurs as follows:

1. Modulator overload is detected, and the integrators are reset with a short pulse.
2. The modulator starts running again, and as the overloading signal is still present, the output pulse density ramps toward a high value with a time constant determined by the loop filter. At a certain point, oscillations begin once again. Loops that are designed with high-bandwidth STF's will begin to oscillate very quickly after coming out of the reset state, whereas loops with a “slow” STF will take some time to achieve an output pulse density that is high enough to trigger the start of oscillations.
3. The oscillation detection circuitry goes into action and once again resets the integrators.

This sequence of events happens repeatedly for the entire duration of the overloading input event. The frequency at which these repeated reset events occur is determined by the roll-off characteristic of the STF. If the frequency of reset events is much higher than the cutoff frequency of the decimation filter, then the noise introduced by the reset sequence will be partially filtered by the decimation filter. Simulations show that the dc value of the bit stream during these repeated reset events is larger than the normal full-scale bit-stream density. Therefore, a decimator that is designed with saturation logic to prevent the output signal from exceeding full scale may completely hide the increase in noise that occurs during signal overload and repeated reset events. This results in “well-behaved” decimator outputs during modulator overload. On the other hand, modulators that are designed to

have low-pass STF characteristics may have a frequency of reset events that is lower than the cutoff frequency of the decimation filter, causing large amounts of noise to appear at the decimator output. Such a modulator may also exhibit a reduction in average bit-stream densities during sustained overload, causing an inverted “cusp” to appear at the peaks of the decimator output signal.

5.8 PRACTICAL MEASURES FOR PREVENTING IDLE TONES

Another strong motivation for using a high-order modulator is that the idling patterns produced by such a modulator tend to be more stochastic and less likely to produce large idling tones in the baseband. Chapter 3 discusses this effect in detail. One cannot, however, assume that any high-order modulator will be free of idle tones. In fact, this is one of the most common areas of first-silicon disappointment, and many an engineer has spent long hours in the lab trying to find why a circuit does not work as well in reality as in simulation. Here we will only discuss practical measures for minimizing idle tones; the reader is referred to Chapter 3 for a more thorough treatment.

Most commercial designs of high-order loops avoid the problem of nonstationary quantization noise by a simple method: they make the theoretical quantization noise lower than the circuit noise (caused by switched-capacitor circuits, op-amps, reference noise, etc.) by 15 dB or more. This makes sense in light of the fact that the actual area and power consumed by an integrated modulator has more to do with analog circuit noise than with the theoretical quantization noise, and therefore it makes sense for the analog circuit noise to dominate over the quantization noise. This noise “perfume” goes a long way toward covering up any strange tones or noise modulation produced by the modulator. If the intended application is audio, it is wise to recall that the ear can be roughly modeled as a constant- Q spectrum analyzer with fractional bandwidths as small as one-fifth of an octave at midfrequencies. This unfortunately means that the ear can detect tones that are buried in a white-noise floor by as much as 20 dB.

Most practical $\Delta\Sigma$ converters have idle tones that are far in excess of what is predicted by computer simulation using difference equations. This is almost always due to tones that occur at much higher frequencies folding down into the baseband as a result of nonlinearities or other practical circuit effects. These high-frequency tones can be found in simulation by “zooming in” on the region around one-half of the sample frequency. For small dc inputs, spectral sticks of considerable magnitude may be found in this frequency region. There are two common ways that these tones can fold into the baseband:

1. *Interfering clock signals on the reference voltage, especially at odd multiples of $f_s/2$.* The reference pin is a multiplicative input to the modulator and hence directly translates signals from the region around $f_s/2$ to the region around dc. The reference pin is extremely sensitive to this effect. Often only a few microvolts of interference is enough to degrade idle-tone performance.
2. *Nonlinearity in the input op-amp.* Since there are multiple large tones at high frequencies, the difference frequency between pairs of tones may fold down into the baseband. If the requirement for low signal distortion is not very severe, the unwary designer will often use an op-amp with poor linearity. This design approach may lead to serious idle-tone problems. Most successful designs insist on op-amp linearity that

is at least the equal of the desired SNR of the converter, even if this figure is overkill as far as signal distortion is concerned.

To minimize problem (1), an external large reference bypass capacitor is often used to eliminate noise on the reference pin. Problem (2) can only be addressed by careful op-amp design.

These high-frequency idling patterns may be largely eliminated by adding a dither signal. The benefits and penalties associated with dither in high-order modulators are described in Chapter 3.

5.9 PRACTICAL IMPLEMENTATION OF A STEREO 18-BIT $\Delta\Sigma$ ADC IC

Detailed circuit design techniques for single-bit high-order loops is not markedly different than those used for second-order loops, which are covered in detail elsewhere in this book. The only additional hardware is normally the integrator reset or nonlinear state-variable limit circuit to ensure global stability. Since most high-order loops are used in designs where high SNRs are desired, individual building blocks such as op-amps must normally achieve very high performance in areas such as noise, distortion, slew rate, and bandwidth.

In this section we will describe the design of a commercially available dual, 20-kHz-bandwidth, 18-bit $\Delta\Sigma$ ADC with an oversampling ratio of 64 and a noise floor that is 104 dB below full scale [9, 11]. The converter is implemented on two chips: a 10-V, 3- μm CMOS modulator chip and a 5-V CMOS decimator chip. Only the modulator design will be discussed here.

5.9.1 Noise-Shaping Modulator IC

The techniques described in Chapter 4 were used to design a fifth-order noise-shaping filter with an inverse Chebyshev characteristic for the NTF. The loop filter employs two resonator sections and one integrator, resulting in the NTF shown in Figure 5.13. This plot does not include the normalization of the transfer function to make the first value of the impulse response equal to 1, which would increase the magnitude response by 3 dB at all frequencies. The corresponding pole-zero diagram is shown in Figure 5.14.

A block diagram of the circuit is shown in Figure 5.15. The modulator chip contains a dual-output 3.0-V reference, clocking and interface circuitry, and two fifth-order 1-bit fully-differential switched-capacitor $\Delta\Sigma$ modulators. Each modulator contains five amplifiers, a comparator, switches, capacitors, and associated nonlinear stabilization circuitry.

5.9.2 Switched-Capacitor Loop Filter Design

The architecture chosen for this design corresponds to the inverted architecture shown in Figure 5.8. This design was chosen to avoid having to use an extra active summing amplifier at the comparator input (a requirement of the noninverted topologies of Figure 5.6 and Figure 5.7).

Figure 5.16 shows the detailed switching arrangement used for both the signal input paths and the reference feedback path. A fully differential architecture was used to

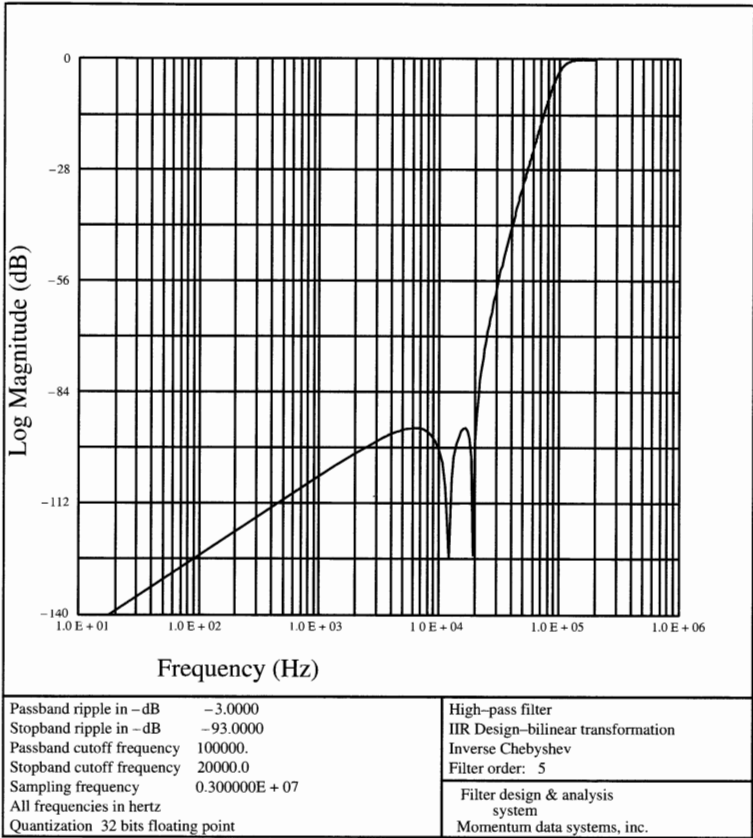


Figure 5.13 Frequency response of the NTF of a fifth-order modulator.

increase signal swing and power supply rejection while reducing the effects of substrate coupling and digital feedthrough. Since switched-capacitor circuits are true sampling devices, even small amounts of high-frequency noise coupling through the supplies or any other path will fold down and cause noise and/or distortion in the audio band. The switches are sized to meet the requirement of 18-bit settling in one-half the clock period, and the RC bandwidths of the switch/capacitor combination may be 30 MHz or more. There is therefore little intrinsic filtering of high-frequency noise that may cause foldovers.

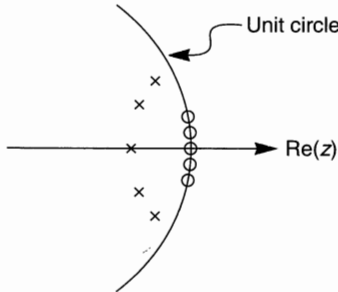


Figure 5.14 Blow-up of pole-zero plot.

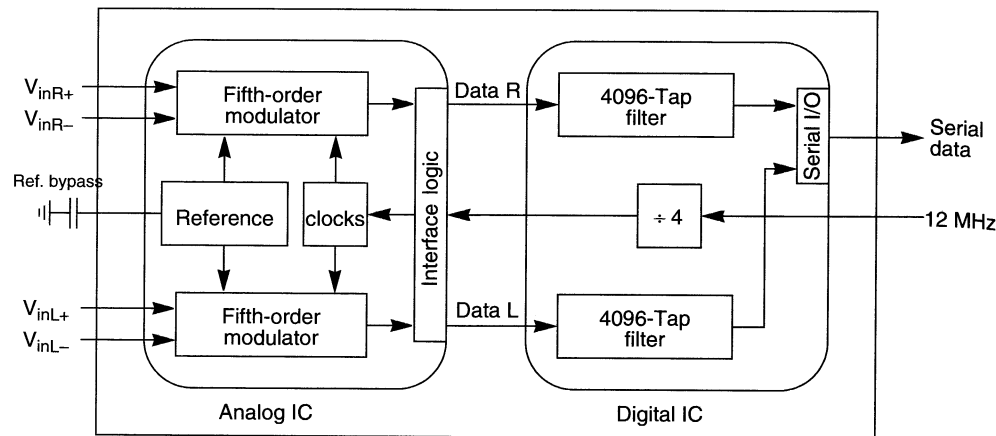


Figure 5.15 Block diagram of the ADC.

On phase 2 of the nonoverlapping clock, the differential input is sampled onto capacitors C_1 and C_2 . To avoid distortion due to signal-dependent charge injection, the grounded phase 2 switch is opened slightly before the input phase 2 switch. This prevents the signal-dependent channel charge of the input switch from appearing on the input capacitor. On phase 1, the input capacitors are connected to the op-amp input terminals, and the stored charge is transferred. This charge is summed with the reference feedback charge, the polarity of which is determined by the comparator decision.

The reference switching scheme is also shown in Figure 5.16. The signals V and $\bar{V}$ are the true and complemented outputs of the comparator. Instead of using both a positive

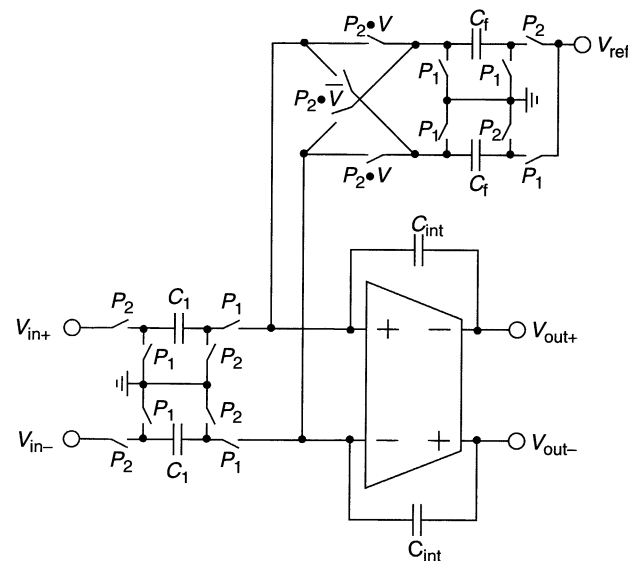


Figure 5.16 A switched-capacitor implementation of an integrator block showing the 1-bit DAC input.

and a negative reference, the feedback sense of the reference is inverted by swapping phase 1 and phase 2 clocks on the reference input sampling switches. In one case, the capacitor is shorted in phase 2 and in phase 1 it is connected between V_{ref} and the integrator input, delivering a positive packet of charge to the integrator. In the other case, the capacitor is connected between V_{ref} and ground on phase 2 and between ground and the integrator input on phase 1, delivering a negative charge packet to the integrator. Depending on the decision of the comparator, these charge packets are steered on phase 1 into either one side or the other side of the differential integrator. This scheme has the important advantage that the current drawn from the reference does not depend on the signal. Were this not true, some nonlinear version of the signal would appear on the reference and cause distortion, as the reference is a multiplicative input to the system. An external large capacitor is used to reduce noise on the reference voltage.

The differential architecture is fully pipelined, with each integrator having an effective delay of one sample. This avoids the double-settling problem, where two op-amps are connected in series during the settling phase of the circuit. Double settling would impose more stringent settling requirements on each op-amp.

The notches in the NTF are implemented by feeding back around pairs of integrators, as shown in Figure 5.10. As described previously, this feedback moves the open-loop poles of the feedback filter along the unit circle to frequencies in the audio band. It is interesting to note that in the open-loop configuration, these sections would in theory oscillate with constant amplitude at their resonant frequency. Fortunately, the application of negative feedback to the system turns these open-loop poles into zeros on the unit circle, where they serve to reduce in-band quantization noise.

5.9.3 Circuit Noise Considerations

Careful attention was paid to all noise sources in the circuit. The theoretical quantization noise of the loop (as simulated using a difference-equation discrete-time simulator) is at the -112 -dB level. One noise source that is frequently overlooked is the digital truncation noise that occurs at the decimator output when the accumulator is truncated to 18 bits. This noise is by definition at the 18-bit level (110 dB). This truncation is properly dithered by the analog circuit noise so as to remove any distortion effects that might otherwise occur.

The capacitors were sized from kT/C noise considerations, with the first-stage capacitors being the dominant noise source and hence the largest capacitors. These capacitors must be quite large to give 18-bit noise performance, and this results in large op-amps with high bias currents to meet the 18-bit settling requirement. The total noise of the circuit is the power sum of the theoretical loop quantization noise, the kT/C noise, the digital truncation noise, and the op-amp thermal noise.

The theoretical performance of the loop with no analog noise sources was -112 dB. It is quite typical to overdesign the loop in this way, because it is surprising how fast all of the noise sources that result from practical circuit effects can add up. Since many high-resolution $\Delta\Sigma$ converter ICs are dominated by capacitor area, it makes sense to ensure that kT/C noise dominates all the rest of the noise sources by a large margin. It is often cheaper in terms of area to increase the order of the loop filter than it is to decrease the kT/C noise by 3 dB, which doubles the capacitor area.

5.9.4 Stabilization Using Integrator Reset

Higher order modulators are stable over a limited range of one's density at the output [1, 5, 6]. Instability in this modulator is sensed digitally, by counting the number of consecutive 1's or 0's in the modulator bit stream. A sufficiently long string of either 1's or 0's indicates modulator instability and triggers circuitry that resets the state in the integrators to put the modulator into a stable operating condition. When a signal continuously overloads the modulator for an extended period of time (e.g., dc), the modulator goes into a mode where it detects overload, resets, comes out of reset, and after some time goes into overload again. As long as the average value of the 1-bit stream remains high enough, the digital clipper in the decimator will prevent any of this noisy behavior from appearing at the output.

5.9.5 Op-amp Design

The performance of the input amplifier in higher order loops determines overall converter performance. Therefore, to achieve the stated performance, the following features were designed into our front-end amplifier:

- Fully differential, including a common-mode loop.
- Settling to 18-bit accuracy (23 μV) in one-half clock cycle (120 ns).
- Gain linearity of 0.001%, or 10 μV peak nonlinearity over the entire 3-V peak output range.
- Input the referred thermal noise contribution below the modulator noise floor of -105 dB full scale.

The question of whether or not the op-amp must settle to the full resolution of the converter is still undecided. In this design, a conservative approach was taken to avoid risk, and full settling was a design goal of the op-amp (at least according to SPICE; measurements to confirm this settling performance are not easily done!). Arguments can be made that less than full settling can be tolerated if the “nonlinear” portion of the settling waveform is less than the resolution of the converter. This nonlinear settling behavior is a result of op-amp slewing at the beginning of the settling period, the effects of which are still present at the end of the settling period.

A simplified circuit diagram of the amplifier is shown in Figure 5.17. A single-stage folded-cascode architecture was chosen for both the differential and common-mode paths. This topology has an inherently high bandwidth that enables the amplifier to meet the settling time specifications. Compensation for both the common-mode and differential-mode paths is provided using the switched-capacitor circuit capacitances surrounding the amplifier.

The common-mode output voltage is sensed using PMOS source followers on each output driving a resistive and capacitive summing network at the output. The summed voltage is compared to ground through an NMOS differential pair, and the difference current is fed back to the amplifier to control the common-mode current.

The amplifier linearity is limited by impact ionization current in the NMOS output devices. This impact ionization current causes an effective nonlinear resistive load on the amplifier, and this will cause distortion when reflected through the finite gain of the

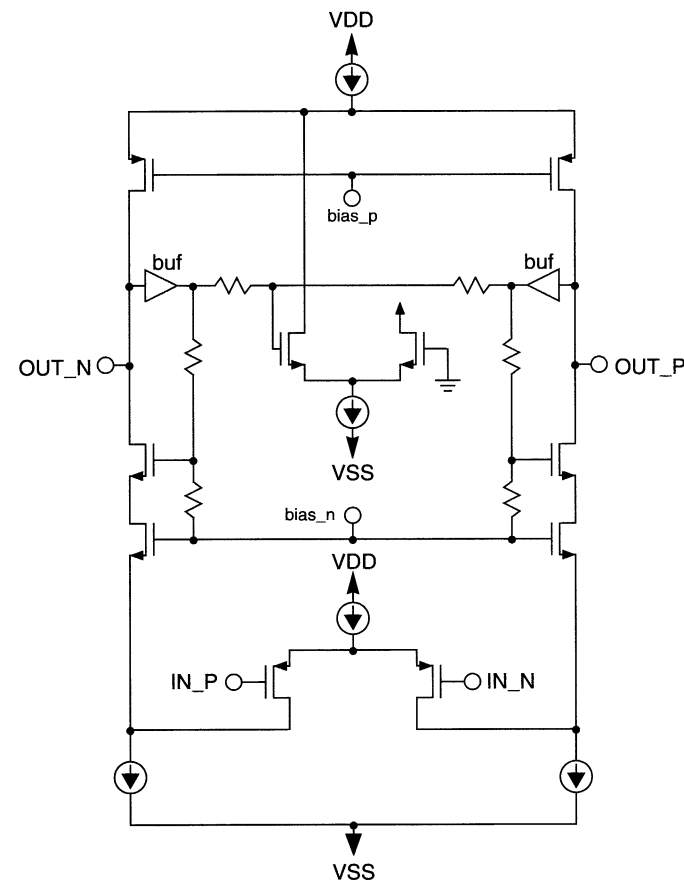


Figure 5.17 A single-stage folded-cascode amplifier with improved linearity.

amplifier to the input terminals. This problem is corrected by placing an NMOS transistor between the output and the NMOS current source and driving its gate with half the difference between the output and the minus analog supply. This has the effect of maintaining a low drain-source voltage across all NMOS signal transistors. Since impact ionization is a strong function of the drain-to-source voltage, this circuit dramatically reduces impact ionization. Without the correction circuitry, the third harmonic distortion products were only 88 dB below the fundamental signal. The correction circuitry improves the distortion performance by almost 20 dB.

5.9.6 Results and Comments

The actual silicon results were within about 2 dB of the original noise analysis (103 dB achieved vs. 105 dB expected). Most of this difference can be traced to D/A interference; when the decimator and modulator are physically separated, performance improves. One must recall that the RC bandwidth of the sampling switches is large, and therefore it is very difficult to achieve very high resolution in a noisy digital environment.

Distortion was measured at 1 kHz and found to be approximately 98 dB below the fundamental. Frequency response was flat within a few hundredths of a decibel.

Considerable time was spent on PC-board layout and decoupling techniques to achieve this performance. In particular, the reference pin is quite sensitive to pickup. Any signal on the reference has the potential to fold down shaped noise, as the reference pin is a multiplicative input to the system. Even the small amount of signal picked up on the bond wire to the internal reference pad was found to be significant. One must be prepared to spend considerable amount of laboratory time tracking down such subtle effects if the highest performance levels are to be reached.

REFERENCES

- [1] W. L. Lee and C. G. Sodini, "A topology for higher order interpolative coders," *Proc. 1987 IEEE Int. Symp. Circuits Sys.*, vol. 4, pp. 459–462, May 1987.
- [2] R. W. Harris, "Enhanced delta modulation encoder," U.S. Patent 4,509,03, filed Dec. 1, 1982, assigned to Gould Inc.
- [3] G. C. Temes, R. H. Walden, and T. Catalepe, "Architectures for high-order multibit sigma-delta modulators," *Proc. IEEE Int. Symp. Circuits Sys.*, vol. 2, pp. 895–898, May 1990.
- [4] R. W. Adams, "Design and Implementation of an audio 18-bit analog-to-digital converter using oversampling techniques," *J. Audio Eng. Soc.*, vol. 34, pp. 153–166, March 1986.
- [5] R. W. Adams, P. F. Ferguson, A. Ganesan, S. Vincelette, A. Volpe, and R. Libert, "Theory and practical implementation of a fifth-order sigma-delta A/D converter," *J. Audio Eng. Soc.*, vol. 39, pp. 515–528, July 1991.
- [6] D. R. Welland, B. P. Del Signore, E. J. Swanson, T. Tanaka, K. Hamashita, S. Hara, and K. Takasuka, "Stereo 16-bit delta-sigma A/D converter for digital audio," *J. Audio Eng. Soc.*, vol. 37, pp. 476–486, June 1989.
- [7] R. Schreier, "Noise-shaped coding," Ph.D. Dissertation, University of Toronto, 1991.
- [8] M. O. J. Hawksford, "Chaos, oversampling, and noise-shaping in digital-to-analog conversion," *J. Audio Eng. Soc.*, vol. 37, no. 12, Dec. 1989.
- [9] P. F. Ferguson, Jr., A. Ganesan, and R. W. Adams, "One bit higher order sigma-delta A/D converters," *Proc. IEEE Int. Symp. Circuits Sys.*, vol. 2, pp. 890–893, May 1990.
- [10] S. K. Tewksbury and R. W. Hallock, "Oversampled, linear-predictive and noise-shaping coders of order > 1 ," *IEEE Trans Circuits Sys.*, vol. 25, pp. 436–447, July 1978.
- [11] C. C. Cutler, "Transmission systems employing quantization," U.S. Patent 2,927,962, filed 1954, issued 1960.

Mike Rebeschini

The Design of Cascaded $\Delta\Sigma$ ADCs

6.1 INTRODUCTION

The early structures used to implement oversampled $\Delta\Sigma$ ADCs closely resembled their predecessor, the delta modulator (Δ modulator). In fact, a first-order $\Delta\Sigma$ modulator is functionally equivalent to a Δ modulator preceded by an integrator. In actual implementations, the integrator is in the forward path between the subtractor and the quantizer, for practical reasons. The first-order $\Delta\Sigma$ modulator is then seen to be a Δ modulator with the integrator in the forward path as opposed to the feedback path [1].

Significant performance improvements in terms of dynamic range and reduction of idle pattern tones were achieved with the second-order $\Delta\Sigma$ modulator. Second-order noise shaping is achieved by placing two integrators in the forward path plus a zero for stability. Both the first-order modulator and the second-order modulators described in [1] and [2] are examples of what have been referred to as single-loop single-stage, or classic, modulators.

A block diagram of a generalized single-loop modulator is shown in Figure 6.1. The distinguishing characteristic of these modulators is the presence of only one quantizer. The digitized output of the quantizer is input to a DAC. The output of the DAC is subtracted from the analog input. Therefore, the entire system consists of a single loop. If the quantizer is a 1-bit quantizer, or comparator, only first- and second-order modulators of this type are unconditionally stable. Higher order single-loop modulators can be stabilized by correct choice of filter coefficients and by restricting the region of operation. Examples of higher order single-loop modulators can be found in [3–5].

Another approach for realizing higher order modulators is shown in Figure 6.2. In this structure, the overall $\Delta\Sigma$ modulator is not a single loop, but consists of a cascade of

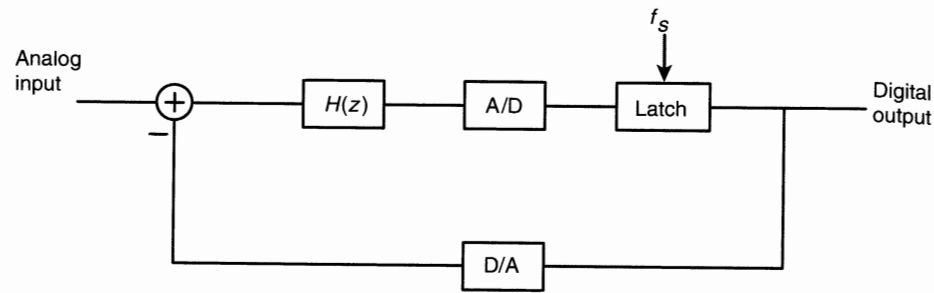


Figure 6.1 Block diagram of single-loop modulator.

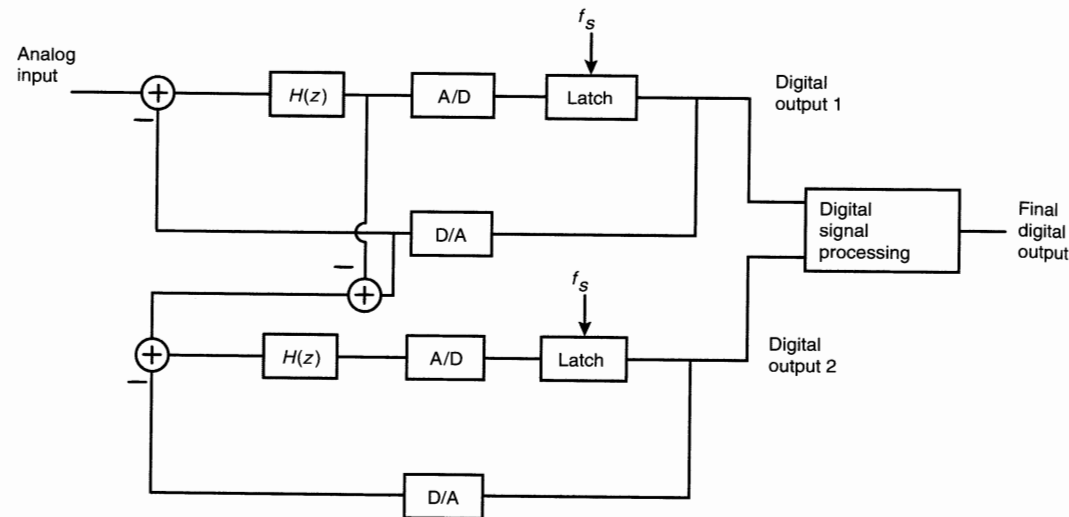


Figure 6.2 Block diagram of cascaded modulator.

several lower order single-loop modulators, each with its own quantizer. Each single-loop modulator in the cascade converts the quantization error from the preceding modulator. The errors of all but the last single-loop modulator are then digitally canceled. Examples of these types of converters, which have been termed cascaded, or MASH, $\Delta\Sigma$ modulators, can be found in [6–8].

This chapter will deal with the design of cascaded $\Delta\Sigma$ modulators. A comparison will be made between single-loop and cascaded designs, describing the advantages and disadvantages of each approach. A linearized model will be derived and used to analyze specific third-order modulator structures. Circuit topologies used for the implementation of the single-loop modulators in a third-order (1–1–1) cascaded design are described. Using the linearized model, the effects of circuit nonidealities on overall modulator performance are calculated. Some experimental results from the third-order modulator described in [8] are presented. The chapter concludes with the presentation of some previously unpublished material regarding continuous-time implementations of cascaded modulators.

With the exception of the material presented in Section 6.7, the circuits in this chapter are assumed to be switched-capacitor implementations. Therefore discrete-time z -domain transfer functions are used throughout. Circuit implementations are shown as single ended for simplicity, although they may be implemented as fully differential circuits in the actual design. The designs presented here assume a low-pass system. Therefore, $H(z)$ will be an integrator, or a low-pass filter. The material in this chapter can be extended in a straightforward manner to bandpass $\Delta\Sigma$ modulators by making $H(z)$ a bandpass transfer function.

6.2 SYSTEM DESIGN

6.2.1 Comparison of Single-Loop and Cascaded Designs

6.2.1.1 Single-Loop Designs Referring to Figure 6.1, it can be seen that the digital output of the modulator is fed into a DAC. The output of the DAC is compared to the analog input. The difference between the analog input and the DAC output is fed through the forward path of the loop, which consists of $H(z)$ and the quantizer. Here, $H(z)$ is chosen such that there is a large amount of gain in the forward path at the lower, in-band frequencies. At these frequencies, the input-to-output transfer function will be determined mostly by the feedback path, consisting of the DAC. This implies that the overall modulator performance will be relatively insensitive to the tolerance of the analog components making up $H(z)$, or the linearity of the quantizer. This robustness is one of the major advantages of single-loop modulators. However, since the DAC is in the feedback path, the linearity of the overall converter will be no better than the linearity of the DAC. Nonlinearity in the DAC will cause harmonic distortion and an increase in baseband noise due to intermodulation of high-frequency noise.

By far the most popular choice for a quantizer in a $\Delta\Sigma$ modulator is a 1-bit quantizer, or comparator. The reason for this is that it allows the DAC in the feedback path to be a 1-bit converter also, so there are no linearity problems. In fact, this choice is so common that in the remainder of this chapter the quantizers will be assumed to be comparators unless explicitly stated otherwise.

When the quantizer is a comparator, the gain of the quantizer is somewhat ambiguous, and the effective gain of the quantizer depends on the region of operation of the system. This presents a problem for higher order single-loop modulators, since the gain of the quantizer will affect the placement of the closed-loop poles. If the poles enter the right-half plane, the system becomes unstable. Because of this, only first- and second-order single-loop modulators are unconditionally stable. The stability of $\Delta\Sigma$ modulators was covered in Chapter 4.

The quantization noise in the output signal of a $\Delta\Sigma$ converter will not consist entirely of shaped white noise but will also contain discrete spectral lines, or tones. These tones can present a problem in digital audio even if their power is less than the total power of the noise floor. Determining the exact structure of the quantization noise is a formidable mathematical problem and can only be solved for certain cases (see Chapter 2). It has been observed experimentally that the higher the order of a single-loop modulator, the lower the power of the in-band idle tones. Therefore, higher order single-loop modulators have less need for a dither signal.

6.2.1.2 Cascaded Designs Figure 6.2 shows a generalized cascade of two modulators. The second modulator converts the error from the first modulator, which is then digitally canceled. It can be seen that this is equivalent to a two-step converter, with both the coarse and fine converters being $\Delta\Sigma$ modulators. It is not necessary that both converters be of the same type. The first converter can be a $\Delta\Sigma$ modulator and the second converter a pipeline, flash, or algorithmic converter. It is probably desirable that the first converter be a $\Delta\Sigma$ modulator for linearity reasons. In this chapter, all converters in a cascade are assumed to be $\Delta\Sigma$ modulators.

The first thing to note is that the cascaded modulator is not a single-loop system. The error from the first modulator is converted by the second modulator. The output of the second modulator is digitally recombined with the first modulator output to cancel the first modulator error. The degree of cancellation depends on how well the analog implementation of $H(z)$ matches the inverse of the digital implementation of $H^{-1}(z)$. The performance of the cascaded modulator is more sensitive to the imperfections of the analog components than that of the single-loop modulator. However, the accuracy requirements on the analog components are nowhere near as severe as for an equivalent successive approximation converter, because the in-band noise has been greatly reduced by noise shaping prior to the digital cancellation.

It is interesting to consider the idle-tone performance of a cascaded $\Delta\Sigma$ modulator. In theory, cascades of three or more first-order modulators yield a final quantization noise that corresponds to what would be predicted using the white-noise model for the quantizer (Chapter 2). Therefore, the idle-tone performance for a cascade of first-order modulators should be better than the idle-tone performance of any single-loop, single-bit modulator. However, in practice, this is not the case. Because of imperfect cancellation, the idle-tone performance of a cascade of modulators is determined by the uncanceled noise from the first modulator. If the first modulator in the cascade is a first-order modulator, then the uncanceled noise will exhibit the idle-tone characteristics of a first-order modulator. Since a first-order modulator has very poor idle-tone performance, the overall idle-tone performance of a cascaded modulator can be worse than that of a single-loop modulator, and a dither signal may be required. An improvement can be made by using a second-order modulator as the first modulator in the cascade, as will be discussed later.

Since a cascaded modulator structure contains only feedforward paths and no feedback between the individual modulators, it will be stable if the modulators it is composed of are stable. This is one of the advantages of a cascaded modulator. It can be made inherently stable to any order over all regions of operation. Arbitrarily low quantization noise cannot be obtained in practice by constructing very high order modulators. At some point, the performance will be limited by uncanceled noise from the first modulator. When this point is reached, no gain in performance will be realized by adding more stages to increase the overall modulator order.

6.2.2 Analytical Linearized Modeling

Since the $\Delta\Sigma$ modulator is a nonlinear system, an exact mathematical analysis is extremely difficult. A linearized model of the system, while not entirely correct, can be solved analytically. Such a model is commonly used to gain a semiquantitative understanding of the system and to predict how various parameters will affect the overall

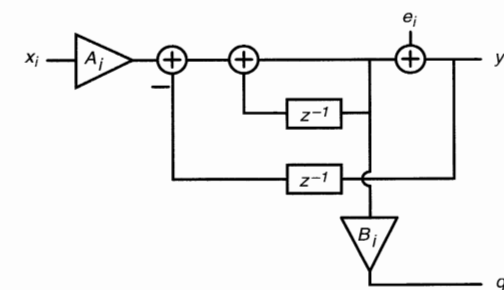


Figure 6.3 Extended linearized model of a first-order modulator.

modulator characteristics. As long as the limitations of the model are kept in mind, it can prove to be extremely useful. In a typical design, the linearized model will be used initially, so that the designer can perform hand calculations and gain an intuitive feel for how the system operates. Before finalizing the design, software simulations will be performed, since simulation is the only way to observe the true theoretical behavior of the modulator.

The linearization is usually accomplished by modeling the clocked quantizer as a gain stage followed by a summing point where white quantization noise is added in. If the quantizer is a 1-bit quantizer or comparator, the concept of gain becomes somewhat confusing. The comparator will act as an AGC over most of the input range, maintaining the loop gain at unity. For single-loop modulators of order greater than 1, the signal range at the input to the comparator will increase greatly at large input signals, and the effective comparator gain will decrease. The magnitude of the quantization noise increases, resulting in a decrease in SNR. The modulator exhibits the overload characteristic common to single-loop modulators.

In order to make the analysis of cascaded $\Delta\Sigma$ modulators a more tractable problem, an extended linearized model is proposed. The comparator is modeled as a block, providing an AGC function to make the loop gain equal to 1, followed by a summing point where white quantization noise is added in, as in the linearized model. The additional assumption is made that the final digital output of a single-loop $\Delta\Sigma$ modulator is relatively insensitive to variations in the analog components that are inside the loop. This is borne out by simulation. A complete linearized model for a first-order modulator is depicted in Figure 6.3. The overall modulator gain is represented by A_i . In a switched-capacitor implementation this is set by the ratio between the switched input capacitor and the switched feedback capacitor. The term $B_i(z)$ takes into account the nonideal transfer function of the integrator, and q_i is the output signal of the integrator.

6.2.3 Software Simulations

As mentioned previously, the only accurate method for observing the theoretical behavior of a $\Delta\Sigma$ modulator is through simulation. Writing the code for simulating $\Delta\Sigma$ modulators is a fairly simple task, particularly for discrete-time implementations. A block diagram showing the partitioning and flow of the system simulator is depicted in Figure 6.4. The inclusion of a comb-decimating filter allows for finer frequency resolution in the

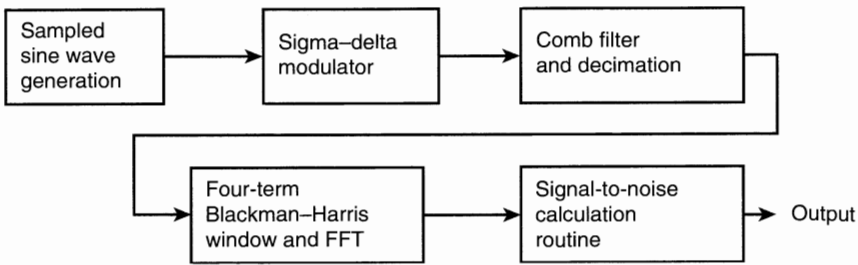


Figure 6.4 Block diagram of the system simulator.

passband without having to resort to an FFT with an extremely large number of points. The decimated data is windowed prior to performing the FFT. The purpose of the window is to prevent spectral leakage from the fundamental when the sampling frequency is not an exact multiple of the signal frequency.

The SNR is calculated by dividing the signal power, as taken from the FFT, by the sum of the powers of all the other bins. The ratio is multiplied by a correction factor to take into account the difference between the coherent and the incoherent gain of the window. This is one of several methods of determining the SNR that will be discussed in greater detail in Chapter 14. Several bins on either side of the signal bin should be thrown out before calculating the noise power. This is due to the fact that the main lobe of the four-term Blackman-Harris window [9] is two bins wide and that the first sidelobes are only 92 dB down, which may cause significant error in the evaluation of high-performance converters.

As mentioned previously, a comb filter in software provides a convenient means for filtering out the noise that would otherwise be aliased into the passband during the decimation process. A comb filter of order n , with decimation ratio D , has a transfer function given by

$$H(z) = \left(\frac{1 - z^{-D}}{1 - z^{-1}} \right)^n \frac{1}{D^n}$$

(6.1)

When using a comb decimation filter in a simulation to evaluate the theoretical performance of a particular modulator structure, the differences between a comb filter with a decimated frequency of twice the passband edge and a brick-wall filter with a cutoff frequency at the passband edge should be kept in mind. The comb filter will have significant droop in the passband, and the filtering of the quantization noise differs somewhat

between a comb filter and a brick-wall filter. Table 6.1 addresses this issue. The attenuation of quantization noise due to noise shaping and filtering for an oversampling ratio of 64 is shown for both a comb filter of given order and a brick-wall filter.

The comb filter will consistently predict a lower level of quantization noise than the idealized brick-wall one. This discrepancy will increase with increasing comb filter and modulator order. The reason for the lower noise results obtained from the comb filter is that the effect of quantization noise attenuation due to the drop in the passband outweighs the effect of additional quantization noise aliasing back into the passband due to the finite attenuation in the comb filter stopband.

6.3 ANALYSIS OF SPECIFIC CASCADED ARCHITECTURES

6.3.1 Third-Order (1–1–1) Modulator

The system diagram for a third-order modulator constructed from a cascade of three first-order modulators is depicted in Figure 6.5. This is a (1–1–1) modulator, where the numbers in parentheses refer to the orders of the individual loops. This architecture was first proposed in [6].

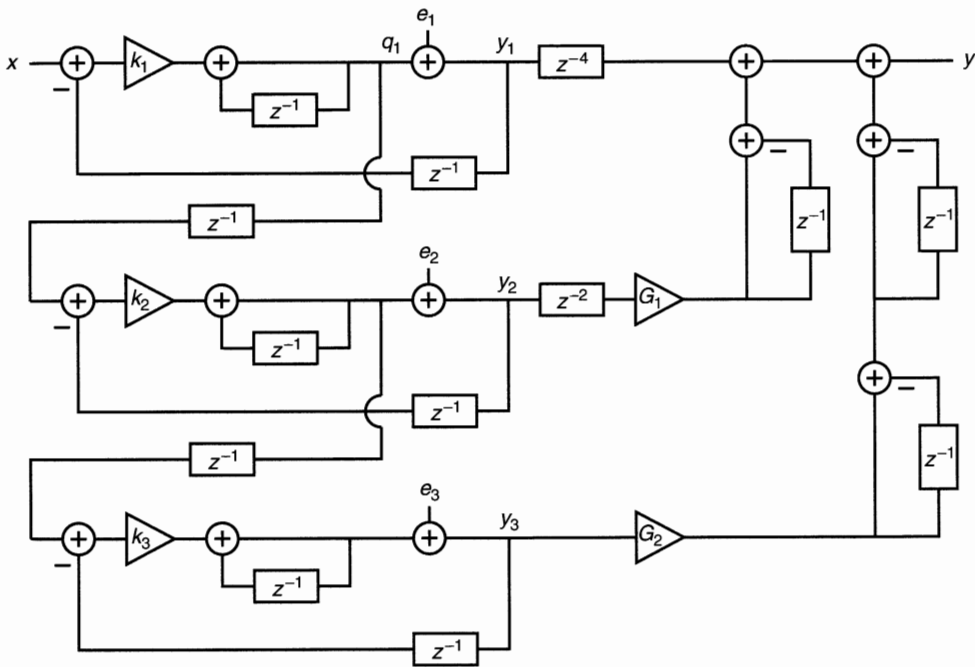


Figure 6.5 System diagram for a third-order (1–1–1) cascaded modulator.

TABLE 6.1 ATTENUATION OF QUANTIZATION NOISE (IN DB) DUE TO NOISE SHAPING AND FILTERING. OVERSAMPLING RATIO = 64

Order of noise shaping	0	1	2	3
Brick-wall	–18.1	–49.0	–77.4	–105.0
Third-order comb	–20.7	–54.2	–82.5	
Fourth-order comb	–21.3	–55.9	–86.0	–113.4

In the particular example shown in Figure 6.5, the input to each modulator, with the exception of the first, is the output of the integrator of the modulator that precedes it. The output of the integrator of an ideal first-order modulator is equal to

$$q_1 = x - e_1 z^{-1} \quad (6.2)$$

Thus, the second modulator is converting, not just the quantization noise from the first modulator, but also a signal that contains the quantization noise and the input signal. In general, any signal (or linear combination of signals) that contains the quantization noise could have been fed from the first modulator to the second. The digital recombination of the individual modulator outputs will be different, depending on the signal that is fed from one modulator to the next. The choice of signal to be fed from each modulator to the succeeding modulator will depend on practical considerations, such as ease of implementation, sensitivity of noise cancellation to analog circuit component nonidealities, and dynamic range.

In the diagram shown in Figure 6.5, K_1 , K_2 , and K_3 are gains associated with the analog integrators in the first, second, and third modulators, respectively. The terms G_1 and G_2 are digital gain blocks with values $G_1 = 1/K_1$ and $G_2 = 1/K_1 K_2$. These values are chosen such that the quantization noise from the first and second modulators in the cascade is canceled. The extra delay in the path between the output of a modulator and the input to the next modulator results from the switched-capacitor circuit used to implement the modulator. The ideal transfer function for the overall third-order modulator is

$$y = z^{-2}(1 - z^{-1} + z^{-2})x + G_2(1 - z^{-1})^3 e_3 \quad (6.3)$$

For this design, the integrator gains were set to $K_1 = K_2 = \frac{1}{2}$, because of dynamic range considerations. The maximum signal swing at each node in a first-order modulator is well defined under normal operating conditions. If the output of the modulator is assigned the values +1 or -1, then the input to the modulator must be in the range -1 to +1 to avoid overload. The output of the integrator will then lie in the range -2 to +2 under no-overload conditions. A gain of $\frac{1}{2}$ needs to be introduced into each integrator so that the integrator output does not exceed the allowable input range of the next modulator. This will result in a value of $G_2 = 4$, which causes the third-order shaped quantization noise to be amplified by 12 dB.

The extended linearized model (Figure 6.3) can be used to obtain a modulator transfer function with nonideal components taken into account. Using the notation x_i for the input, q_i for the integrator output, and y_i for the quantizer output of the i th modulator, the following expressions are derived:

$$y_i = A_i x_i + e_i(1 - z^{-1}) \quad (6.4)$$

$$q_i = B_i(A_i x_i - e_i z^{-1}) \quad (6.5)$$

$$x_{i+1} = z^{-1} q_i \quad (6.6)$$

Making use of Eqs. (6.4), (6.5), and (6.6), the somewhat unwieldy transfer function of the system including nonidealities is given below:

$$\begin{aligned} y = & x A_1 z^{-2} [A_2 A_3 B_1 B_2 G_2 + (1 + A_2 B_1 G_1 - 2A_2 A_3 B_1 B_2 G_2) z^{-1} \\ & - (A_2 B_1 G_1 - A_2 A_3 B_1 B_2 G_2) z^{-2}] \\ & + e_1 z^{-3} (1 - z^{-1}) [(1 - A_2 B_1 G_1) + (A_2 B_1 G_1 - A_2 A_3 B_1 B_2 G_2)(1 - z^{-1})] \\ & + e_2 G_1 z^{-2} (1 - z^{-1})^2 \left(1 - A_3 B_2 \frac{G_2}{G_1}\right) + e_3 G_2 (1 - z^{-1})^3 \end{aligned} \quad (6.7)$$

For an oversampling ratio of 64, the in-band first-order shaped noise will be approximately 51 dB down from the peak signal. This implies that 1% of the quantization noise from the first modulator can leak through, and a SNR of 90 dB can still be obtained. In contrast, a conventional successive approximation converter sampling at the Nyquist rate would require components accurate to 15 bits to achieve the same performance. Furthermore, the noise from each succeeding modulator is shaped to a higher degree than that from the previous modulator. The higher the order of noise shaping, the greater the attenuation of the noise in the passband. For example, with an oversampling ratio of 64, the second-order shaped noise in the passband will be 79 dB down from the peak signal. Therefore, the requirements on cancellation of noise from the second modulator are much less than the requirements on cancellation of noise from the first modulator. Because of the noise shaping, it is generally true that only the first modulator in a cascade of modulators requires special design considerations.

The dominant source of quantization noise in the (1-1-1) cascade shown in Figure 6.5 is due to uncanceled first-order-shaped noise from the first modulator. This error term is

$$\varepsilon = e_1 z^{-3} (1 - z^{-1}) (1 - A_2 B_1(z) G_1) \quad (6.8)$$

Complete noise cancellation is obtained by setting the product $A_2 B_1 G_1$ equal to unity. In practice, however, this is not possible due to nonideal characteristics in the analog components. This is the major drawback of the cascaded $\Delta\Sigma$ technique; the cancellation of quantization noise of the first modulator is dependent on the accuracy of the analog components that determine $A_2 B_1$. Since G_1 is a digital gain, it is exactly 2. Here, A_2 is determined by the matching between the switched input and switched feedback capacitors of the second modulator. Ideally A_2 has a value of 1.

Now consider the error due to the $B_1(z)$ term. The transfer function of the imperfect integrator in the first loop is given by

$$H_{\text{int}}(z) = \frac{1}{2} \frac{1 - \alpha}{1 - z^{-1}(1 - \beta)} \quad (6.9)$$

where α is the gain error and β is the pole error. This yields the following expression for $B_1(z)$:

$$B_1(z) = H_{\text{int}}(z)(1 - z^{-1}) = \frac{1}{2} \frac{1 - \alpha}{1 - z^{-1}(1 - \beta)} (1 - z^{-1}) \quad (6.10)$$

This equation gives

$$1 - A_2 B_1(z) G_1 \approx \beta \left(\frac{z^{-1}}{1 - z^{-1}} \right) + \alpha \quad \text{for } \beta \ll \frac{2\pi f_c}{f_s} \quad (6.11)$$

Substituting Eq. (6.11) into Eq. (6.8) and ignoring delays, the following expressions are obtained for the pole error ε_p and the gain error ε_g :

$$\varepsilon_p = \beta e_1 \quad \text{and} \quad \varepsilon_g = \alpha(1 - z^{-1})e_1 \quad (6.12)$$

The nonzero pole error β causes a leakage of unshaped quantization noise to the output of the converter. A nonzero gain error α causes a leakage of first-order shaped noise. Later in this chapter the physical causes of gain and pole error will be discussed.

It should be noted that the linearized model provides an exact description of the system as long as the error terms e_i represent the difference between the input and the output of the quantizer. It is only when assumptions are made about the properties of the quantization error, such as white spectral density or uniform probability distribution, that error is introduced into the analysis. Equation (6.12) correctly expresses the fact that a pole error will cause leakage of e_1 to the output of the modulator and a gain error will cause leakage of e_1 first-order shaped to the output of the modulator. Later in this chapter, the assumptions that the quantization noise is white and uniformly distributed will be made in order to obtain quantitative results. Once these assumptions are made, the analysis is no longer exact.

Another method for implementing a third-order cascaded (1–1–1) modulator is shown in Figure 6.6. In this scheme, the output of the integrator is subtracted from the modulator output (in fact, from the DAC output), resulting in just the error being fed from each modulator to the succeeding modulator. If the output of the modulator is assigned the values ± 1 , then the quantization error of the modulator will be in the range -1 to $+1$. This allows $K_1 = K_2 = 1$ without overloading the second and third modulators. Thus, $G_1 = G_2 = 1$, and the third-order shaped noise is not amplified. If the third-order shaped quantization noise

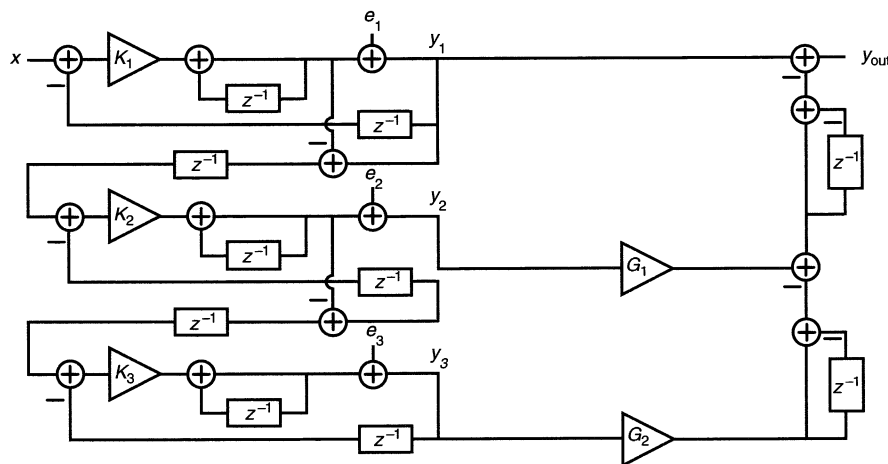


Figure 6.6 System diagram for a (1–1–1) modulator that feeds forward just the error.

is the dominant noise source in the system, this architecture will have lower in-band noise than the system shown in Figure 6.5, in exchange for a slight increase in complexity.

6.3.2 Third-Order (2–1) Modulator

In general, modulators of any order can be cascaded. The major advantage of the cascaded technique is the ability to implement higher order modulators that are unconditionally stable. It would seem reasonable, then, to cascade second-order modulators, since second-order modulators are the highest order single-loop modulators that are unconditionally stable when using a 1-bit quantizer. This section describes a third-order modulator that consists of a second-order modulator cascaded with a first-order modulator [7].

The system diagram for a third-order (2–1) modulator is shown in Figure 6.7. In this example the quantization error from the first modulator is fed forward to be converted by the second modulator. There are a number of advantages to having a second-order modulator as the first modulator in the cascade. The noise that needs to be canceled is now second-order shaped instead of first order, so there is less in-band noise to be canceled. Also, any noise that leaks through from the first modulator will exhibit the characteristics of second-order modulator noise. Therefore, one would expect a cascaded modulator with a second-order modulator as the first modulator in the cascade to have better tone performance than a cascade with a first-order modulator for the same degree of noise cancellation.

Refer to Figure 6.7 for the error analysis of the cascaded (2–1) modulator. Using the linearized model shown in Figure 6.3, the following equations can be derived:

$$\begin{aligned} y_1 &= A_1 x + e_1 (1 - z^{-1})^2 \\ q_{12} &= [(x - y_1 z^{-1}) H_1 - z^{-1} y_1] H_2 \\ y_2 &= A_2 (y_1 - q_{12}) + e_2 (1 - z^{-1}) \end{aligned} \quad (6.13)$$

Assuming the errors are small, the primary error is due to an incomplete cancellation of e_1 . This can be shown to be a second-order shaped noise due to the gain error,

$$\varepsilon_g = e_1 (1 - z^{-1})^2 [\alpha_1 + \alpha_2 + (A_2 - 1)] \quad (6.14)$$

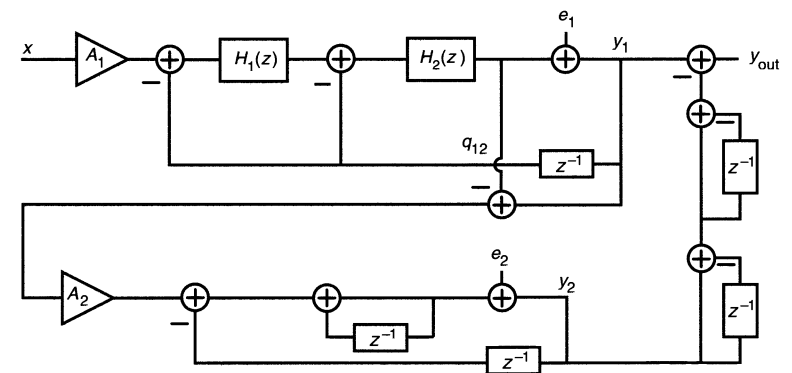


Figure 6.7 Diagram used in the analysis of a (2–1) cascaded modulator.

and a first-order shaped noise due to the pole error,

$$\varepsilon_p = e_1(1 - z^{-1})(\beta_1 + \beta_2) \quad (6.15)$$

Here, A_2 is the conversion gain of the second modulator, while α and β are the gain and pole errors of the nonideal integrators H_1 and H_2 , defined by

$$H_i(z) = \frac{1 - \alpha_i}{1 - z^{-1}(1 - \beta_i)} \quad (6.16)$$

When a first-order modulator is used as the first modulator in a cascade, the noise due to the pole error is unshaped and the noise due to the gain error is first-order shaped. Thus, it can be seen that using a second-order modulator as the first modulator greatly eases the requirements on the precision of the integrators. It should be noted that the nonidealities of both integrators in the first modulator are of equal importance in determining the leakage of noise from the first modulator to the overall system output.

Unlike a first-order modulator, the quantization noise of a second-order modulator is not bounded under normal operating conditions. As the input signal approaches full scale, the quantization noise of the first modulator will increase greatly. If it becomes too large, it will exceed the allowable input range of the second modulator. The range of the input signal can be restricted to avoid large excursions at the output of the second integrator in the first modulator. Also, the quantization noise can be scaled down before entering the second modulator. This will, however, cause the quantization noise from the second modulator to be multiplied by the inverse of the scale factor. Due to these dynamic range considerations, the maximum attainable performance of a (2–1) cascade will fall short of what would be predicted for an ideal third-order modulator.

6.4 CIRCUIT TOPOLOGIES FOR THIRD-ORDER (1–1–1) CASCADE

6.4.1 Autozeroed Integrator

The basic building block in a $\Delta\Sigma$ ADC is the analog integrator. The overall converter performance is largely determined by the characteristics of the integrator, particularly in the case of cascaded $\Delta\Sigma$ modulators. This section will describe the effect of finite op-amp gain on a switched-capacitor integrator transfer function. A special autozeroed integrator topology is presented that results in a more ideal integrator transfer function [10]. In the case of the third-order (1–1–1) modulator that is the subject of Sections 6.4, 6.5, and 6.6, only the characteristics of the integrator in the first modulator are important because of noise shaping.

A single-ended switched-capacitor integrator is shown in Figure 6.8. The ideal transfer function for this integrator is given by

$$\frac{V_o^o}{V_i^e} = \frac{C_1}{C_2} \frac{z^{-1/2}}{1 - z^{-1}} \quad (6.17)$$

The actual integrator transfer function will differ from the ideal due to nonzero switch resistance, finite op-amp bandwidth, and finite op-amp gain. Of these effects, only finite

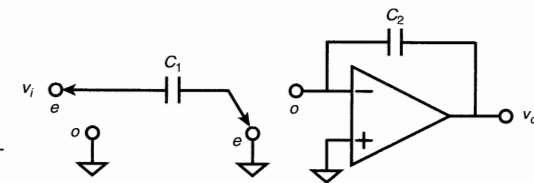


Figure 6.8 Single-ended switched-capacitor integrator.

op-amp gain causes a pole error. From Eq. (6.12), it can be seen that a pole error in the first integrator of a (1–1–1) cascade results in the leakage of unshaped quantization noise to the converter output. This noise leakage may be the limiting performance factor for cascaded $\Delta\Sigma$ modulators that employ a first-order loop as the first modulator, particularly if high-speed operation is desired. The transfer function of the integrator taking finite op-amp gain into account is

$$H(z) = \frac{(C_1/C_2)z^{-1/2}[1 - 1/A - C_1/(C_2A)]}{1 - [1 - C_1/(C_2A)]z^{-1}} \quad (6.18)$$

The gain-compensated integrator is shown in Figure 6.9. This integrator response is much less sensitive to the effects of finite op-amp gain. The term C_p represents any fixed parasitic capacitance on the summing junction.

The odd phase is the integrating phase, where charge is transferred onto integrating capacitor C_4 . During the odd phase, C_2 is charged to v_i and C_3 is charged to v_o . The even phase is the autozeroing phase. During the even phase, one terminal of C_4 floats, so the charge on it is not affected. Also, C_3 , which was previously charged to the output voltage, is switched in across the op-amp. The input side of C_1 is returned to v_i , but the charge injected onto the summing junction from C_1 is canceled by the switching of C_2 . If $C_1 = C_2$, then v_o during the autozeroing phase will be the same as it was during the previous integrating phase. The voltage at the summing junction due to finite op-amp gain is stored on C_1 . For frequency components of the input signal that are much lower than the sampling frequency, this is approximately the voltage needed at the summing junction to

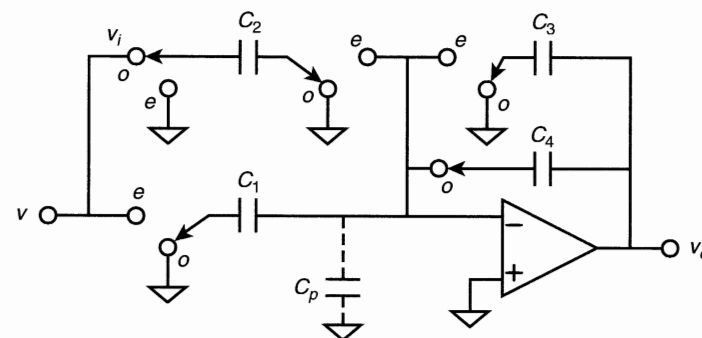


Figure 6.9 Gain-insensitive switch-capacitor integrator.

create the next output voltage. The gain of the op-amp is thus made to appear infinite, to a first-order approximation. The actual transfer function is

$$H(z) = \frac{(C_1/C_4)z^{-1/2}}{1 - z^{-1}} \frac{[1 + (1/A)(-1 - C_1'/C_4)]}{1 + z^{-1}/(1 - z^{-1})[C_1'/(C_4A^2)](1 + C_2'/C_3)} \quad (6.19)$$

where C_1' is the sum of all input capacitances connected to the summing junction including C_1 and C_p , and C_2' is the sum of all switched input capacitors including C_2 . The transfer function is complex because when finite op-amp gain is assumed, the summing junction is no longer a virtual ground. This causes the pole location to depend on the input capacitors.

The autozeroing operation cancels, not only the effect of finite op-amp gain, but also the $1/f$ noise and offset voltage. By effectively boosting the op-amp gain, this integrator configuration also improves the distortion performance of the integrator if the distortion is due to the nonlinear transfer function of the op-amp.

6.4.2 First Modulator of Third-Order (1–1–1) Cascade

The schematic of the modulator that was used as the first modulator in the (1–1–1) cascade is shown in Figure 6.10. The modulator makes use of the autozeroed integrator described in Section 6.4.1.

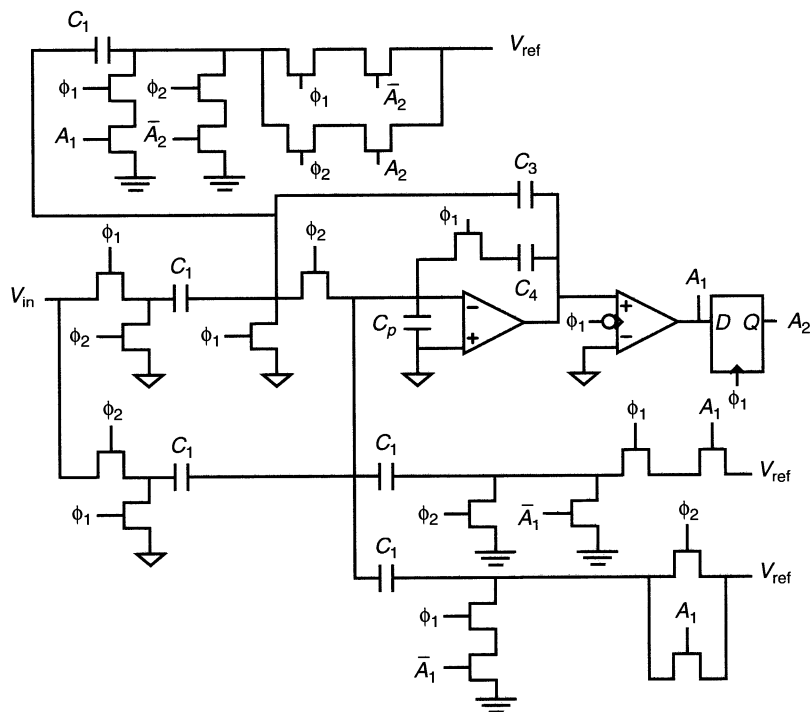


Figure 6.10 First modulator in third-order (1–1–1) cascade.

Although the gain-insensitive integrator does not entirely cancel out the effects of the finite op-amp gain A , an analysis of the circuit yields the gain and pole errors

$$\alpha = \frac{1}{A} \left(1 + \frac{3C_1 + C_p}{C_4} \right) \quad (6.20)$$

$$\beta = \frac{3C_1 + C_p}{C_4A^2} \left(1 + \frac{2C_1}{C_3} \right)$$

Note that the pole error has decreased by a factor of approximately A , the op-amp gain, when compared to the transfer function for the standard integrator given in Eq. (6.18).

A qualitative explanation for the operation of the gain-insensitive integrator is that the op-amp is autozeroed, with the output of the op-amp remaining unchanged from its value during the previous integrating phase. The voltage on the summing junction of the op-amp due to finite gain is then stored on the input capacitor(s). If the input signal varies slowly compared to the sampling frequency, then the voltage stored on the input capacitor(s) is approximately equal to the voltage required on the summing junction to create the output voltage during the next integrating phase. This will greatly increase the effective gain of the op-amp. It is often pointed out that the input signal to the integrator is not slowly varying, since it contains the feedback signal from the comparator output. Since the autozeroed integrator is a linear system, even when finite op-amp gain is taken into account, the presence of high-frequency signals will have no effect on the ability of the autozeroing scheme to reduce integrator errors at the lower, in-band frequencies. Using the autozeroed integrator in a $\Delta\Sigma$ modulator will reduce the in-band quantization noise leakage due to pole error.

6.4.3 Second and Third Modulators of Third-Order (1–1–1) Cascade

In the (1–1–1) cascade, only the design of the first modulator requires special attention. Both the circuit and quantization noise in the second and third modulators are greatly attenuated by the noise shaping, and the performance requirements of these modulators are far less stringent. Because of this, a straightforward implementation of the integrator is used in the second and third modulators of the cascade. This modulator is depicted in Figure 6.11. The gain and pole errors for this modulator are given by

$$\alpha = \frac{1}{A} \left(1 + \frac{3C_1}{C_2} \right) \quad \beta = \frac{1}{A} \frac{3C_1}{C_2} \quad (6.21)$$

The modulator shown in Figure 6.11 was used for the second and third modulators in the third-order $\Delta\Sigma$ converter of Figure 6.5. In Figure 6.6, a cascaded $\Delta\Sigma$ structure is shown in which the integrator outputs are subtracted from the modulator outputs, resulting in only the quantization error of each modulator being fed forward to the next stage. It is very convenient to accomplish this subtraction by feeding forward the integrator output, as was done in the previous case, and using the comparator output to control the ± 1 feedback circuitry of the next modulator in the cascade. Figure 6.12 is a schematic of a structure that could be used for the second and third modulators in the architecture shown in Figure 6.6. The first modulator would still use the structure shown in Figure 6.10. Figure 6.12 is

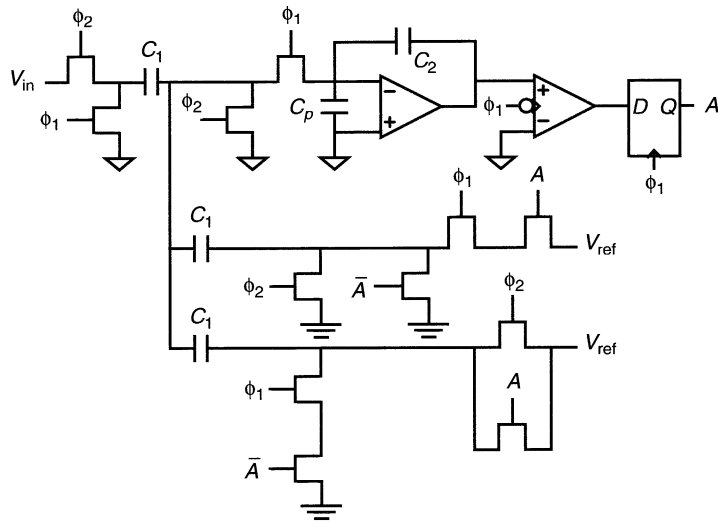


Figure 6.11 Second and third modulators for the architecture shown in Figure 6.5.

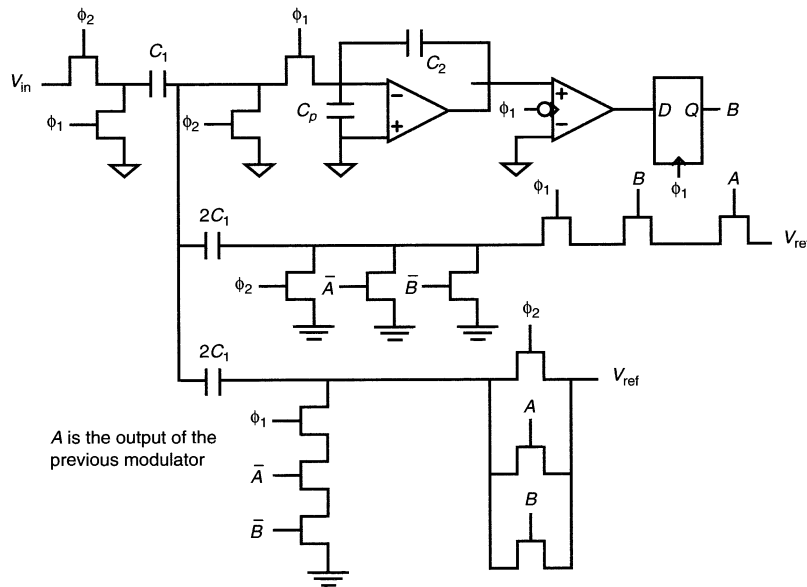


Figure 6.12 Second and third modulators for the architecture shown in Figure 6.6.

identical to Figure 6.11, except for the doubling of the size of the switched feedback capacitors and the addition of some control switches.

The input to the modulator in Figure 6.12 is actually the negative of the quantization noise. The only effect this will have is a slight modification of the digital recombination circuitry. The integrator output of the previous modulator is the input to this modulator. The output of the previous modulator is subtracted through the switched feedback capacitors. Since the output of the present modulator is also being subtracted, the feedback capacitor sizes are doubled, and three-state switching logic is used. If the present modulator and the previous modulator outputs are different, the feedback signal is zero. If they are the same, then the feedback signal is either +2 or –2, depending on the sign of the outputs.

6.5 SOURCES OF ERROR FOR THE THIRD-ORDER (1–1–1) CASCADE

In this section, the sources of error for the converter reported in [8] are calculated. This converter consists of a third-order (1–1–1) modulator, using the architecture shown in Figure 6.5, followed by a fourth-order decimate-by-64 comb filter. The first modulator in the cascade uses the gain-insensitive structure shown in Figure 6.10. The second and third modulators use the structure shown in Figure 6.11. The peak signal is a 1-V zero-to-peak sine wave, which has a power of –3 dBv. The sources of error can be divided into two categories: intrinsic noise and component nonidealities. Intrinsic noise is inherent to the MOS devices, such as $1/f$ noise and thermal noise. Component nonidealities are effects that cause the physical circuit blocks to perform differently from what is predicted by the ideal mathematical model. These effects include capacitor mismatch and finite dc op-amp gain. In a $\Delta\Sigma$ modulator, because of noise shaping, the effect that an error source will have on the overall converter performance depends strongly on the point in the circuit where the error is introduced. Because of the gain that an integrator provides at the lower (in-band) frequencies, usually only the error sources in the first integrator of a $\Delta\Sigma$ modulator are significant.

The autozeroing feature of the gain-insensitive integrator multiplies the $1/f$ noise by $(1 - z^{-1/2})$. For an oversampling ratio of 64 and the same device sizes used in the first op-amp, the resulting $1/f$ noise contribution to the overall noise is negligible.

The thermal noise due to the switches is only important in the first integrator. In this design, there is the equivalent of two switched input capacitors, one for the input and one for the DAC feedback. A switched capacitor has a total thermal noise power given by

$$e_n^2 = \frac{2kT}{C} \quad (6.22)$$

If these capacitors have a value of 1 pF, then the total thermal noise power is equal to –77.8 dB. This white noise is further attenuated 21.2 dB by the fourth-order decimate-by-64 comb filter. Finally, there is a 3-dB improvement because the modulator was implemented as a fully differential circuit. This is due to the fact that the second set of input capacitors will only increase the noise by 3 dB because the noise sources are uncorrelated, while the fully differential arrangement will boost the signal by 6 dB. The final equivalent thermal noise due to the switches is then –102 dBv.

The thermal noise due to the op-amp is calculated in a similar manner. The compensation capacitor for the op-amp is equal to 5 pF. This will give a total noise power

$$e_n^2 = \frac{2kT}{3C_C} = -89.5 \text{ dBv} \quad (6.23)$$

This will be attenuated an additional 21.2 dB by the comb filter, and a 3-dB improvement will be gained from the fully differential circuitry. The equivalent thermal noise due to the op-amp is then -114 dBv.

Mismatch in the capacitors will result in a leakage of first-order shaped noise to the converter output. As previously derived, the error is determined by the tolerance of the product A_2B_1 . Here, A_2 is the ratio of the switched input capacitor of the second modulator to the switched feedback capacitor, and B_1 is the ratio of the switched feedback capacitor of the first modulator to the integrating capacitor. If it is assumed that the matching between two capacitors is a Gaussian distributed random variable with the 3-sigma point equal to δ , then the product A_2B_1 should have a 3-sigma point $\delta\sqrt{2}$. The first-order shaped noise is attenuated 55.9 dB by the combined effects of the noise shaping and decimation filter and decreases 9 dB per doubling of the oversampling ratio. This yields a 3-sigma capacitor mismatch noise

$$N_{ce} = -57.7 + 20 \log \delta \quad \text{dBv} \quad (6.24)$$

When matching the capacitors, the integer ratios used in the modulator are particularly convenient, because they allow each capacitor to be made up of identical plates, with no fractional plates. A common centroid layout will minimize mismatch due to variations in etching. Dummy capacitors may be used to make capacitors at the edge of an array match capacitors internal to the array well.

As mentioned previously, the finite op-amp gain in the first integrator causes the gain and pole errors given in Eq. (6.20). Also previously derived were the expressions for the leakage of quantization noise from the first modulator due to pole error,

$$\epsilon_p = \beta e_1$$

and due to gain error,

$$\epsilon_g = \alpha(1 - z^{-1})e_1$$

The nonzero pole error β causes a leakage of unshaped quantization noise to the output of the converter. Since the noise is unshaped, e_1 is attenuated 21.2 dB by the decimation filter with an oversampling ratio of 64 and decreases only 3 dB per doubling of the oversampling ratio. This yields a pole error noise

$$N_{pe} = -26.0 + 20 \log \beta \quad \text{dBv} \quad (6.25)$$

A nonzero gain error causes a leakage of first-order shaped noise, which decreases 9 dB per doubling of the oversampling ratio. After decimation filtering, the gain error noise is given by

$$N_{ge} = [-60.7 + 20 \log \alpha] \quad \text{dBv} \quad (6.26)$$

Ideally the node voltages in a switched-capacitor circuit should settle to their final values before the end of each clock phase. In practice, however, the effects of nonzero switch resistance and finite op-amp bandwidth prevent this from happening. If the settling is linear, then incomplete settling in the first integrator will show up as a gain error and result in the leakage of first-order shaped noise. A linear settling of 0.1% is more than enough to make this source of noise negligible. A more serious problem is the case of nonlinear settling due to parasitic capacitances and device characteristics varying with the operating point. This nonlinear settling can give rise to harmonic distortion and cause high-frequency noise to fold back into the baseband through intermodulation. To avoid these problems, the circuits were designed to settle to within 0.001%. The op-amp bandwidth was increased at the expense of op-amp gain by boosting the current. This trade-off could be made because of the gain-insensitive integrator used in the first modulator.

6.6 EXPERIMENTAL RESULTS FOR THE THIRD-ORDER (1-1-1) CASCADE

In this section, some experimental results for the modulator described in [8] are presented. Only the analog circuitry was actually implemented in silicon. The fourth-order comb filtering was performed in software. The circuit was fabricated in a 1.5- μm double-poly, double-metal CMOS process.

One of the most useful characterizations for ADCs is a measure of the signal-to-noise-plus-distortion ratio $[S/(N+D)]$ versus input amplitude. This type of measurement will bring out any distortion or noise components that are functions of input amplitude. A sweep of $S/(N+D)$ versus input amplitude for this converter is shown in Figure 6.13. The

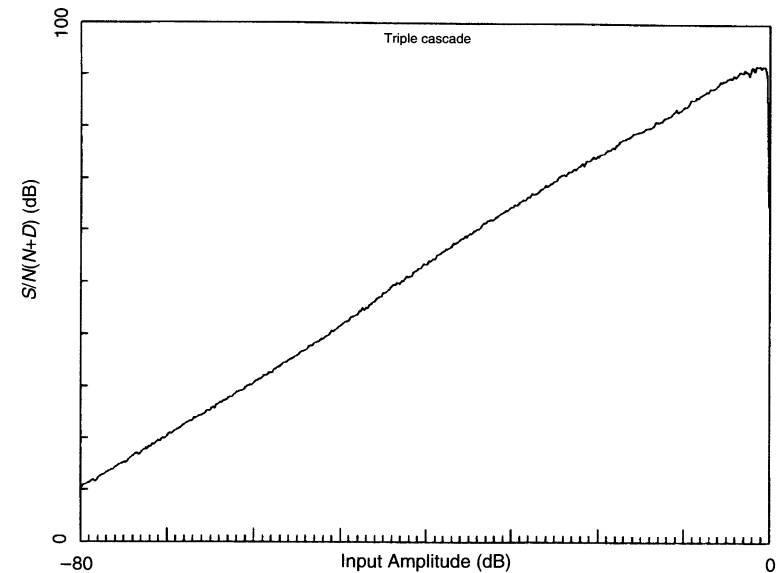


Figure 6.13 Plot of $S/(N + D)$ vs. input amplitude.

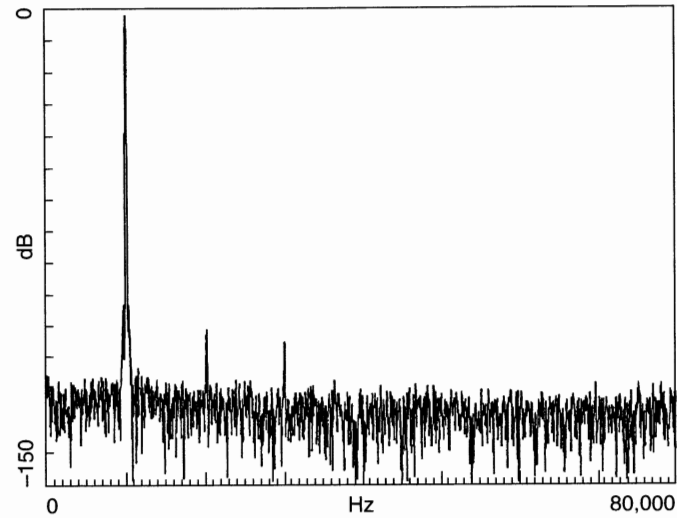


Figure 6.14 Converter output spectrum with sinusoidal input.

TABLE 6.2 CONVERTER MEASURED PERFORMANCE

Parameter	Value
Die size	3.0 mm ²
Power supply	5 V
Power	76.0 mW
Passband	80 kHz
Output rate	160 kHz
Sampling rate	10.24 MHz
Peak S/(N+D) (rms/rms)	91 dB

peak S/(N+D) is 91 dB. For this plot the modulator clock rate is 10.24 MHz, the input frequency is 10 kHz, and the decimation ratio is 64.

To demonstrate the linearity of the converter, a 4096-point FFT of the output using a Blackman–Harris window is shown in Figure 6.14. The second harmonic is 99 dB below the signal, and the third harmonic is 103 dB below the input signal. Note that there is no noticeable harmonic distortion above the third. The input signal is at 2.1 dB below full scale. The input frequency is 10 kHz, and the decimation ratio is 64.

The overall performance and characteristics of the third-order (1–1–1) cascaded modulator are summarized in Table 6.2.

6.7 CONTINUOUS-TIME CASCADED ΔΣ MODULATORS

Most present-day implementations of ΔΣ modulators are discrete-time ones using switched-capacitor circuits. Switched-capacitor circuits are compatible with digital CMOS processing since high-quality capacitors are usually available, whereas high-quality resistors are not. Switched-capacitor implementations also have the advantage of tightly controlled poles and zeros that scale with the clock frequency and a lowered sensitivity to clock jitter. However, in some cases continuous-time implementations may be more suitable. Continuous-time modulators can achieve lower thermal noise levels than switched-capacitor modulators. Achieving the same in-band noise level as a low-valued resistor may require an unrealistically large value of a switched capacitor. Also, continuous-time modulators have less aliasing problems than switched-capacitor designs.

To construct cascades of continuous-time modulators and to determine the correct digital recombination of the individual modulator outputs require an analytical method for determining the signal at various nodes in a continuous-time modulator. First, consider the continuous-time first-order modulator shown in Figure 6.15. This is nearly identical to a discrete-time first-order modulator, except that the integrator is now continuous time and has a transfer function $H(s) = a/s$. The quantizer consists of a comparator followed by a flip-flop clocked at the high-speed clock frequency f_s . It is modeled as a summing node where quantization noise e is added in. The easiest method for solving for the output y of the modulator in Figure 6.15 is to transform it into the equivalent discrete-time first-order modulator, since the solution for the discrete-time modulator is well known. The equation for the integrator output can be written as

$$q[(n+1)\tau] = q[n\tau] + a \int_{n\tau}^{(n+1)\tau} x dt - a\tau y[n\tau] \quad \tau = \frac{1}{f_s} \quad (6.27)$$

A new function $x'(t)$ can be defined such that

$$\tau x'[(n+1)\tau] = \int_{n\tau}^{(n+1)\tau} x(t) dt \quad (6.28)$$

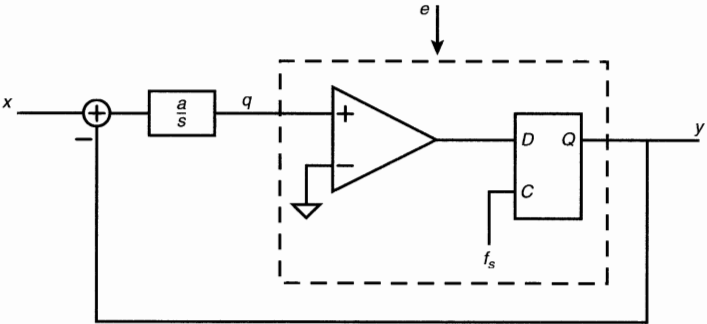


Figure 6.15 First-order continuous-time modulator.

In the frequency domain, this is equivalent to saying that passing x' through a discrete-time integrator gives the same result as passing x through a continuous-time integrator. Therefore,

$$x' \frac{1}{1-z^{-1}} = x \frac{1}{s\tau} \quad \text{and} \quad X'(\omega) = X(\omega) \frac{\sin(\omega\tau/2)}{\omega\tau/2} e^{-j\omega\tau/2} \quad (6.29)$$

Substituting x' for x and making use of the fact that the AGC function of the comparator will make $a\tau = 1$, Eq. (6.27) is seen to be identical to the equation for a first-order discrete-time modulator. The output is then given by

$$y = x' + e(1 - z^{-1}) \quad (6.30)$$

Once the modulator output has been found, the signal at any other node in the modulator can be found by simple linear analysis.

The schematic of a (1-1) cascade with a continuous-time first modulator and a discrete-time second modulator is shown in Figure 6.16. Sampling the output of the continuous-time integrator makes it equivalent to a discrete-time integrator for noise cancellation purposes. The output is given by

$$y = -x'z^{-1} + e_2z^{-1}(1 - z^{-1})^2 \quad (6.31)$$

Here, x' is defined in Eq. (6.29), and e_2 is the quantization noise from the second modulator.

A (1-1) cascade of two continuous-time first-order modulators is shown in Figure 6.17. Since the second loop is also continuous time, the continuous-time nature of the integrator in the first loop has to be taken into account. The output of the first modulator is

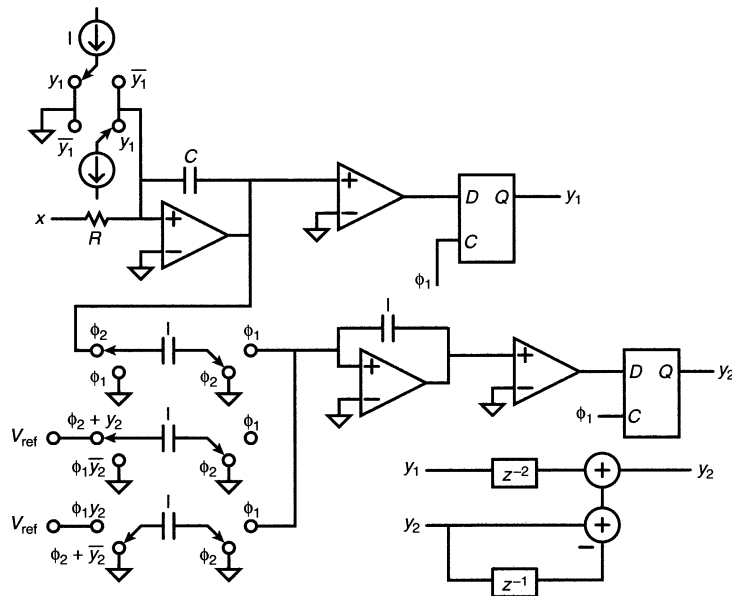


Figure 6.16 A (1-1) cascade, with continuous-time first modulator.

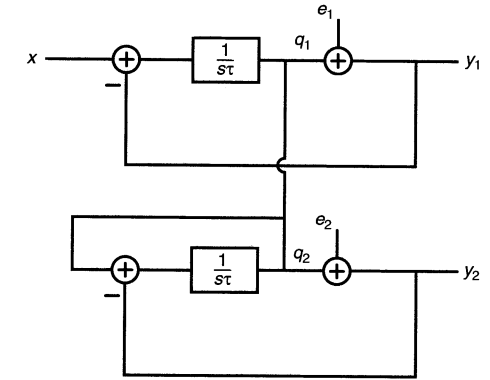


Figure 6.17 Continuous-time (1-1) cascade.

$$y_1 = x' + e_1(1 - z^{-1}) \quad (6.32)$$

Finding the output of the second modulator is a bit trickier. The equivalent input to the second modulator due to the input signal is

$$\frac{x''}{1 - z^{-1}} \quad (6.33)$$

where the Fourier transform of x'' is given by

$$X''(\omega) = X(\omega) \frac{\sin^2(\omega\tau/2)}{(\omega\tau/2)^2} e^{-j\omega\tau} \quad (6.34)$$

The equivalent input to the second modulator due to the first integrator output, neglecting the input signal, is

$$\left[\frac{-x'z^{-1}}{1 - z^{-1}} - e_1z^{-1} \right] \frac{1}{2}(1 + z^{-1}) \quad (6.35)$$

The output of the second modulator is then given by

$$y_2 = \frac{x''}{1 - z^{-1}} - \frac{x'z^{-1}}{1 - z^{-1}} \frac{1}{2}(1 + z^{-1}) - \frac{1}{2}e_1z^{-1}(1 + z^{-1}) + e_2(1 - z^{-1}) \quad (6.36)$$

The output after digital recombination is

$$y = \frac{1}{2}z^{-1}(1 + z^{-1})y_1 + (1 - z^{-1})y_2 \quad (6.37)$$

giving the final result

$$y = x'' + e_2(1 - z^{-1})^2 \quad (6.38)$$

The expressions for x' and x'' are given in Eqs. (6.29), and (6.34). It can be seen that with ideal components perfect noise cancellation can be achieved with a rather simple digital

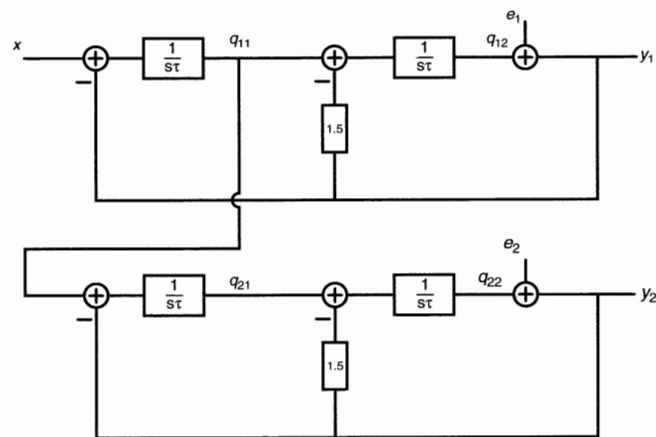


Figure 6.18 Third-order cascade composed of two second-order continuous-time modulators.

recombination. This result is somewhat surprising, since it would seem that the digital recombination would have to be an approximation to the inverse of an analog integration. The reason that such a simple digital recombination provides exact noise cancellation is that the feedback signal y_1 in the first loop is held constant during each time slot. For a constant input, an analog integration over a time slot is approximated exactly by the trapezoidal rule. The cancellation of noise is independent of the input signal x . Note that the noise cancellation depends on the exact nature of the feedback signal in the first loop. If the feedback signal is implemented as a return-to-zero pulse stream, the digital recombination necessary for noise cancellation will be different.

In concluding this section, the third-order modulator of Figure 6.18 is discussed. This configuration consists of two continuous-time second-order modulators, with the input to the second modulator coming from the output of the first integrator of the first modulator. At first glance, this configuration appears wasteful, since fourth-order noise shaping could have been obtained by taking the output of the second integrator in the first modulator as the input to the second modulator. However, this configuration does have some advantages. Even though first-order shaped noise is the input to the second modulator, it is second-order shaped noise that is being canceled. This means that a gain error in the first integrator of the first modulator results in a leakage of second-order shaped noise, and a pole error in the same integrator results in a leakage of first-order shaped noise. This is the same as for the cascaded (2–1) modulator, except that now only the gain and pole errors in the first integrator of the first modulator matter, whereas in the (2–1) cascade, errors in both integrators in the first modulator are equally important. The only parameters that are critical for noise cancellation are the ratio of the ± 1 current feedback pulses to the integrating capacitor of the first integrator in the first modulator and the input-to-output gain of the second modulator. Since the gain of a modulator can be set by matching, the only critical tuning is of the current feedback pulses to the first integrator of the first modulator. This tuning may even be accomplished with a nonlinear element, since the feedback current is restricted to two values, $+1$ and -1 .

The noise from the first modulator is only first-order shaped, but it is noise from a second-order modulator, and hence it has second-order noise characteristics. Therefore, it will have less idle tones than noise from a first-order modulator. Also, in this arrangement, the bounds on the signal being input from the first modulator are well defined, which is not the case for the cascaded (2–1) modulator.

The outputs of the individual modulators can be written as

$$y_1 = x'' + e_1(1 - z^{-1})^2 \quad (6.39)$$

$$y_2 = \frac{x'''}{1 - z^{-1}} - y_1 \frac{(\frac{1}{6}z^{-1} + \frac{2}{3}z^{-2} + \frac{1}{6}z^{-3})}{1 - z^{-1}} + e_2(1 - z^{-1})^2 \quad (6.40)$$

The correct digital recombination is

$$y = y_1(\frac{1}{6}z^{-1} + \frac{2}{3}z^{-2} + \frac{1}{6}z^{-3}) + y_2(1 - z^{-1}) \quad (6.41)$$

which gives a final output

$$y = x''' + e_2(1 - z^{-1})^3 \quad (6.42)$$

The signal x'' was previously defined in Eq. (6.34), and the Fourier transform of x''' is given by

$$X'''(\omega) = X(\omega) \frac{\sin^3(\omega\tau/2)}{(\omega\tau/2)^3} e^{-j3\omega\tau/2} \quad (6.43)$$

Notice that in this case the analog integration in the digital recombination can be implemented exactly by using Simpson's rule. This is because the feedback in the first modulator is a fully sampled and held signal.

6.8 CONCLUSION

This chapter dealt with the analysis and design of cascaded $\Delta\Sigma$ modulators. A comparison was made between single-loop modulators and cascaded modulators, pointing out the relative advantages and disadvantages of the two approaches. An extended linearized model of a $\Delta\Sigma$ modulator was presented. It was shown that such a model can be used to analyze cascaded $\Delta\Sigma$ modulators. This analytical approach provides semiquantitative results but is nonetheless a valuable design aid. As a final design check, simulations that exactly model the system should be performed in software. It was shown that nonideal components give rise to an integrator that has both gain and pole errors, resulting in incomplete noise cancellation. Analytical expressions for quantization noise leakage were derived for a (1–1–1) cascade and a (2–1) cascade, taking into account both gain and pole errors. A (1–1–1) cascaded $\Delta\Sigma$ modulator that was actually fabricated and tested is presented as a design example. Expressions for quantization noise, taking into account finite op-amp gain, capacitor mismatch, and incomplete settling, were derived, as well as expressions for intrinsic flicker and thermal device noise. Experimental results were presented.

Finally, some new material on continuous-time cascaded $\Delta\Sigma$ modulators was presented. Expressions for the transfer functions of continuous-time modulators were

derived. It was shown that continuous-time modulators can be cascaded and that, in the ideal case, complete quantization noise cancellation can be achieved with rather simple digital postprocessing. This somewhat surprising result is due to the fully sampled-and-held nature of the feedback signal.

REFERENCES

- [1] H. Inose and Y. Yasuda, "A unity bit coding method by negative feedback," *Proc. IEEE*, vol. 51, pp. 1524–1535, Nov. 1963.
- [2] J. C. Candy, "A use of double integration in sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, March 1985.
- [3] W. L. Lee and C. G. Sodini, "A topology for higher order interpolative coders," *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 4, pp. 459–462, May 1987.
- [4] D. R. Welland, B. P. Del Signore, E. J. Swanson, T. Tanaka, K. Hamashita, S. Hara, and K. Takasuka, "Stereo 16-bit delta-sigma A/D converter for digital audio," *J. Audio Eng. Soc.*, vol. 37, pp. 476–486, June 1989.
- [5] P. F. Ferguson, Jr., A. Ganesan, and R. W. Adams, "One bit higher order sigma-delta A/D converters," *IEEE Proc. ISCAS '90*, vol. 2, pp. 890–893, May 1990.
- [6] Y. Matsuya, K. Uchimura, A. Iwata, T. Kobayashi, M. Ishikawa, and T. Yoshitome, "A 16-bit oversampling A-to-D conversion technology using triple-integration noise shaping," *IEEE J. Solid-State Circuits*, vol. 22, pp. 921–929, Dec. 1987.
- [7] L. Longo and M. Copeland, "A 13 bit ISDN-band oversampled ADC using two-stage third-order noise shaping," *IEEE Proc. Custom IC Conf.*, pp. 21.2.1–21.2.4, Jan. 1988.
- [8] M. Rebeschini, N. R. van Bavel, P. Rakers, R. Greene, J. Caldwell, and J. R. Haug, "A 16-b 160 kHz CMOS A/D converter using sigma-delta modulation," *IEEE J. Solid-State Circuits*, vol. 25, pp. 431–440, April 1990.
- [9] F. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proc. IEEE*, vol. 66, pp. 51–83, Jan. 1978.
- [10] K. Haug, F. Maloberti, and G. Temes, "Switched-capacitor integrators with low finite-gain sensitivity," *Electron. Lett.*, vol. 21, no. 24, Nov. 1985.

Brian Brandt

Chapter 7

High-Speed Cascaded $\Delta\Sigma$ ADCs

7.1 INTRODUCTION

Oversampled analog-to-digital converters based on $\Delta\Sigma$ modulation have previously been used for high-resolution signal acquisition in voiceband telecommunications, digital audio, and Integrated Services Digital Network (ISDN) applications. In these applications, the use of oversampling techniques has resulted in robust implementations by exploiting the enhanced speed and circuit density of scaled VLSI technologies to overcome resolution limitations resulting from component mismatch and reduced supply voltages.

This chapter examines the application of oversampling techniques in an unconventional role, namely, A/D conversion at rates exceeding 1 MHz with a resolution of 12 bits or more. While Nyquist-rate converters are capable of achieving this level of performance in a CMOS technology using a pipelined architecture [1], there are several distinct advantages to using an oversampling approach. Oversampled ADCs can achieve high resolution without trimming or calibration because of their tolerance for component mismatch and circuit nonidealities. These converters also simplify system integration by reducing the burden on the supporting analog circuitry. Specifically, they do not require precision sample-and-hold circuitry and they relax the performance requirements on the analog antialias filter that precedes the sampling operation. Oversampled ADCs include an inherent digital filtering capability that provides a programmable trade-off between resolution and conversion rate and allows the use of the same converter in a variety of applications.

An increase in the conversion rate of an oversampled ADC may be accomplished by increasing the modulator sampling rate or decreasing the oversampling ratio R , which is the ratio of the modulator sampling rate to the conversion rate. Increasing the modulator sampling rate is facilitated through the use of continuous-time circuits implemented in

high-performance technologies. Modulators based on continuous-time integration avoid problems associated with sampling such as the aliasing of wide-band noise into the base-band and sampling uncertainty. They also significantly reduce requirements on operational amplifier bandwidth [2]. However, several practical problems complicate their implementation. In continuous-time modulators, the analog output levels of the feedback digital-to-analog converter are generally established by charging a capacitance with a constant current for a precise period of time. These modulators therefore depend on accurate pulse shaping [3] and are significantly more sensitive to clock jitter than switched-capacitor implementations. The gain of a continuous-time integrator is inversely proportional to the product of resistance and capacitance values [2]. As a consequence, modulator architectures that are relatively insensitive to gain errors must be used. Moreover, the time constants of the integrators must be scaled if the modulator sampling rate is changed. Finally, the harmonic distortion performance of continuous-time modulators depends on the linearity of the resistors used in the first integrator.

To circumvent the implementation complications of continuous-time modulators, switched-capacitor design techniques were emphasized in the work described in this chapter. An emphasis has also been placed on CMOS implementations because of the relatively low cost of this technology and its widespread use in digital signal processing applications. To achieve conversion rates above 1 MHz using CMOS switched-capacitor circuit techniques, it is advantageous to reduce the oversampling ratio below the range of 64–512 typically used in oversampling converters. The goal of the work described in the following sections of this chapter is to reduce the oversampling ratio to as low as 16 while maintaining a dynamic range of 74 dB or, equivalently, a resolution of 12 bits. After examining the limitations of existing modulator architectures for use at low oversampling ratios in the next section, a cascaded multibit modulator that substantially reduces the oversampling ratio required for 12-bit resolution is introduced in Section 7.3. Issues concerning the implementation of cascaded modulators in general and the cascaded multibit modulator in particular are addressed in Section 7.4. The design of an experimental prototype of the cascaded multibit modulator is described in Section 7.5, and measurement results characterizing its performance are presented in Section 7.6.

7.2 $\Delta\Sigma$ MODULATION AT LOW OVERSAMPLING RATIOS

The dynamic range of a $\Delta\Sigma$ modulator employing pure differentiation noise transfer functions depends on the oversampling ratio R , the order of noise shaping L , and the internal quantizer resolution N , according to [4]

$$\text{DR} = \frac{3}{2} \left(\frac{2L+1}{\pi^{2L}} \right) (2^N - 1)^2 R^{2L+1} \quad (7.1)$$

Figure 7.1 illustrates this equation by plotting the dynamic range and resolution as a function of the oversampling ratio for first-, second-, and third-order noise shaping with 1-bit quantization ($N = 1$) and for second-order noise shaping with 4-bit quantization ($N = 4$). Note that when a 4-bit quantizer is used instead of a 1-bit quantizer, the quantization level spacing is reduced by a factor of 15 for constant outermost quantization levels, resulting in a dynamic range increase of nearly 24 dB.

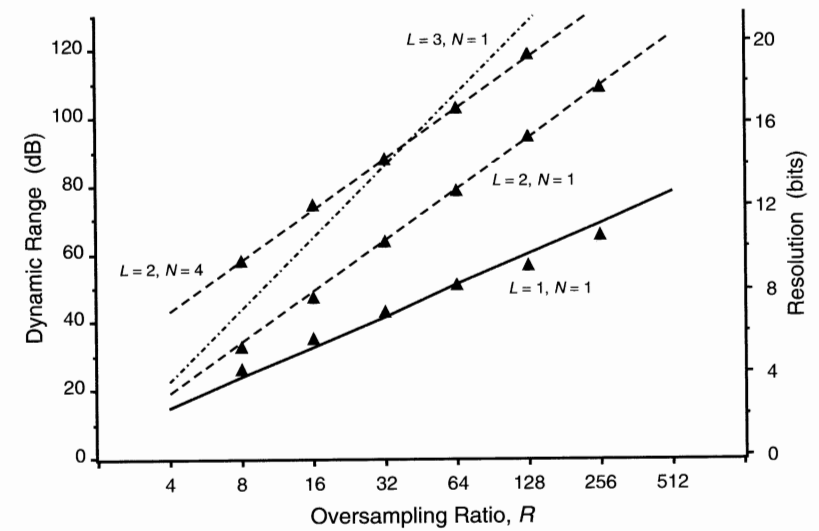


Figure 7.1 Performance of first-, second-, and third-order $\Delta\Sigma$ modulators. Diamonds denote simulated data.

According to Eq. (7.1), the dynamic range of a $\Delta\Sigma$ modulator operating at a given oversampling ratio R may be extended by increasing the order of noise shaping L , or by increasing the quantizer resolution N . Figure 7.1 indicates that the effectiveness of increasing the order of noise shaping is significantly diminished as the oversampling ratio is reduced. In contrast, the effectiveness of increasing the quantizer resolution is independent of the oversampling ratio and is therefore particularly attractive in the present application. However, while modulators based on multibit quantization are tolerant of nonlinearity in the quantizer's ADC because of noise shaping, they may impose stringent linearity requirements on the quantizer's DAC, as discussed in Chapter 6. In these modulators, the error due to DAC nonlinearity effectively enters the modulator at its input. Therefore, the modulator's linearity and resolution are limited by the precision of the multibit D/A converter.

The dependence on DAC linearity may be eliminated by reducing the quantizer's resolution to 1 bit. While such a two-level quantizer is inherently linear, its large quantization noise can cause instability in modulators of order greater than 2, as discussed in Chapters 4 and 5. Architectural modifications to extend the input range for which stable operation is maintained in higher order modulators generally reduce the dynamic range of such modulators below that predicted by Eq. (7.1).

The full dynamic range predicted by Eq. (7.1) can be achieved for $L > 2$ by cascading several first-order modulator stages, as described in Chapter 6. Furthermore, such cascaded modulators are stable and employ 1-bit quantizers to eliminate the dependence on DAC linearity. However, these modulators depend on achieving precise quantization noise shaping in the first stage. In the presence of gain error or integrator leakage, quantization noise from the first stage is not completely canceled. Moreover, discrete noise tones in the spectrum of the first-stage quantization noise can be leaked to the combined modulator output. This problem is exacerbated at low oversampling ratios because of the increased strength of the discrete noise tones [5].

In practice, the dependence on achieving precise noise shaping limits the number of cascaded first-order stages to 3 ($L = 3$). An oversampling ratio exceeding 22 would therefore be required to achieve 12-bit resolution, according to Eq. (7.1). Also, the poor performance of higher order single-bit single-stage modulators at low oversampling ratios makes them unsuitable for achieving 12-bit resolution at an oversampling ratio of 16.

Because the dynamic range improvement resulting from an increase in the quantizer resolution is independent of the oversampling ratio, multibit modulators are particularly attractive for achieving the performance goals outlined in Section 7.1. Chapter 8 describes several methods, including element swapping and digital correction, for reducing the dependence of multibit modulators on DAC linearity. The modulator described in [6] combines calibration and element randomization to significantly reduce harmonic distortion due to DAC nonlinearity. The next section of this chapter presents a cascaded multibit modulator that employs an alternative method for reducing the dependence on DAC linearity and thereby avoids the need for digital correction, calibration, or element-randomizing networks.

7.3 A CASCADED MULTIBIT $\Delta\Sigma$ MODULATOR

Shown in Figure 7.2 is a cascaded multibit $\Delta\Sigma$ modulator that avoids sensitivity to the precision of the DAC by placing the multibit quantizer in the final stage of a third-order cascaded modulator. The more critical first-stage quantizer has only two analog output levels

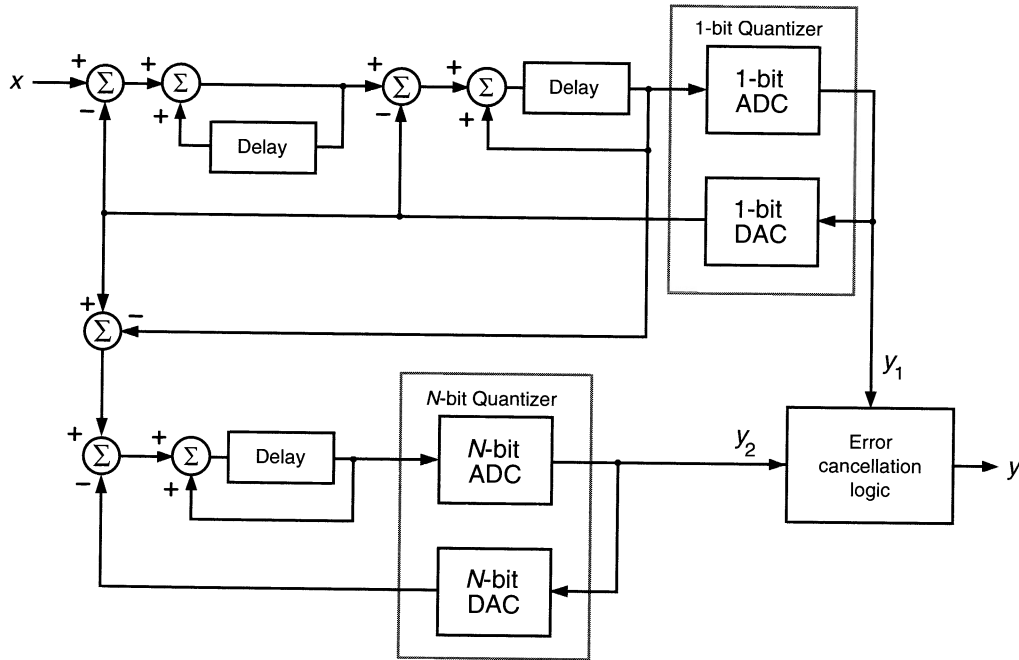


Figure 7.2 Cascaded multibit $\Delta\Sigma$ modulator.

and is therefore inherently linear. The modulator consists of a second-order stage with a cascade of two stages with 1-bit quantization proposed previously [7]. The input to the second stage is the difference between the output and the input of the first-stage quantizer. That is, the input to the second stage is the quantization error of the first stage.

Shown in Figure 7.3 is a linear approximation of the modulator depicted in Figure 7.2, wherein the quantizers are modeled by signal-independent additive error sources, while the integrators are represented by their transfer functions in the z -domain. Here, $E_1(z)$ and $E_2(z)$ model the quantization error of the first- and second-stage ADCs, respectively. The term $E_2(z)$ also contains a representation of nonlinearity in the second-stage ADC, while $E_D(z)$ models the error from nonlinearity in the multibit D/A converter in the second stage. An error source corresponding to $E_D(z)$ does not appear in the first stage because of the inherent linearity of the 1-bit D/A converter.

For the linearized model of Figure 7.3, the z -transform of the output of the first stage is

$$Y_1(z) = z^{-1}X(z) + (1 - z^{-1})^2 E_1(z) \quad (7.2)$$

Thus, the output of the first stage includes the input to the modulator delayed by one sample period plus the second-order difference of the first-stage quantization error $E_1(z)$. The input to the second stage is $E_1(z)$ and the transform of the second-stage output is

$$Y_2(z) = z^{-1}[E_1(z) - E_D(z)] + (1 - z^{-1})E_2(z) \quad (7.3)$$

The error cancellation logic combines the digital outputs from the two stages according to

$$Y(z) = z^{-1}Y_1(z) - (1 - z^{-1})^2 Y_2(z) \quad (7.4)$$

so as to cancel the quantization error $E_1(z)$ of the first stage.

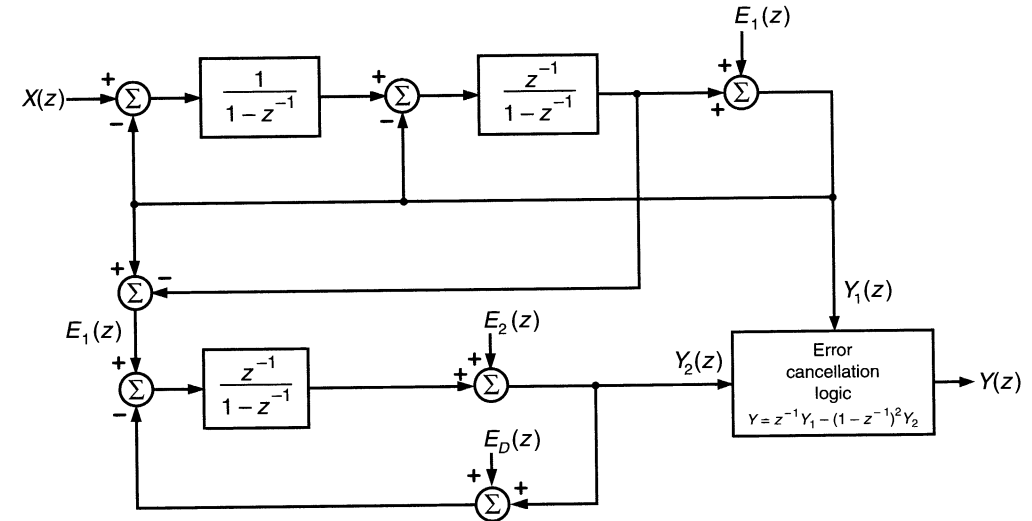


Figure 7.3 Linear model of the cascaded multibit $\Delta\Sigma$ modulator.

The resulting output of the overall modulator is obtained by substituting Eqs. (7.2) and (7.3) into Eq. (7.4),

$$Y(z) = z^{-2}X(z) + z^{-1}(1 - z^{-1})^2E_D(z) - (1 - z^{-1})^3E_2(z) \quad (7.5)$$

Thus, ideally the quantization error of the first stage is canceled and the quantization error of the second stage is attenuated in the baseband by third-order shaping. As in the cascade of three first-order stages described in Chapter 6, third-order noise shaping is achieved without instability because the constituent stages are independently stable. Also, because the quantization error of the second stage originates from a multibit quantizer, the modulator's dynamic range is improved by an increase in the quantizer resolution N according to Eq. (7.1).

In contrast to a single-stage multibit modulator, the error resulting from DAC nonlinearity, $E_D(z)$, does not enter this modulator at its input but instead is attenuated in the baseband by second-order shaping, as is apparent from Eq. (7.5). The second-order shaping makes the cascaded multibit modulator much more tolerant of DAC nonlinearity than the single-stage modulator, as indicated in Figure 7.4. This figure compares the simulated baseband spectra for the modulator shown in Figure 7.2 with that of a third-order single-stage multibit modulator under the circumstance of 5-bit DAC integral linearity. The spectrum for the single-stage modulator exhibits substantial increases in both the noise floor and harmonic distortion, while the spectrum of the cascaded modulator exhibits a noise floor only slightly higher than that resulting from an ideal DAC. More importantly, the cascaded modulator does not display any harmonic distortion, a consequence of the second-order shaping and the fact that the input to the second stage is the quantization error of the first stage, which is substantially decorrelated from the input to the modulator.

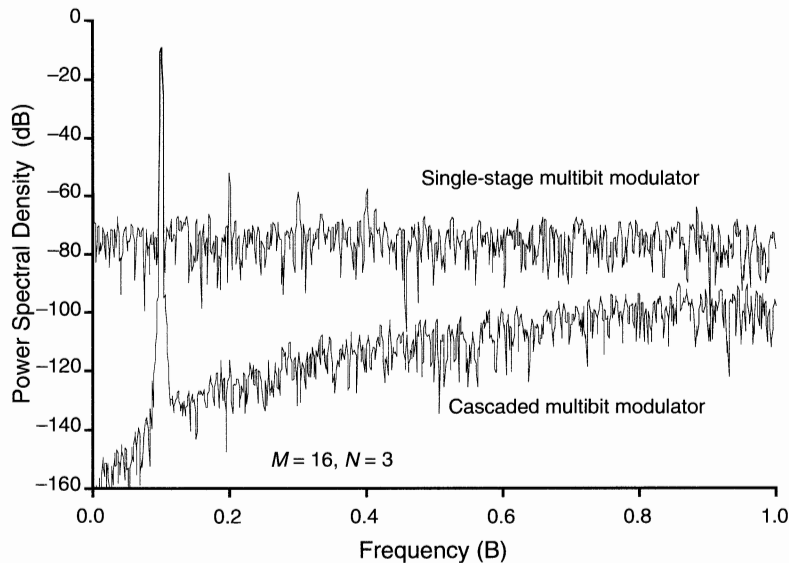


Figure 7.4 Simulated baseband spectra with 5-bit DAC integral linearity; B is the signal bandwidth.

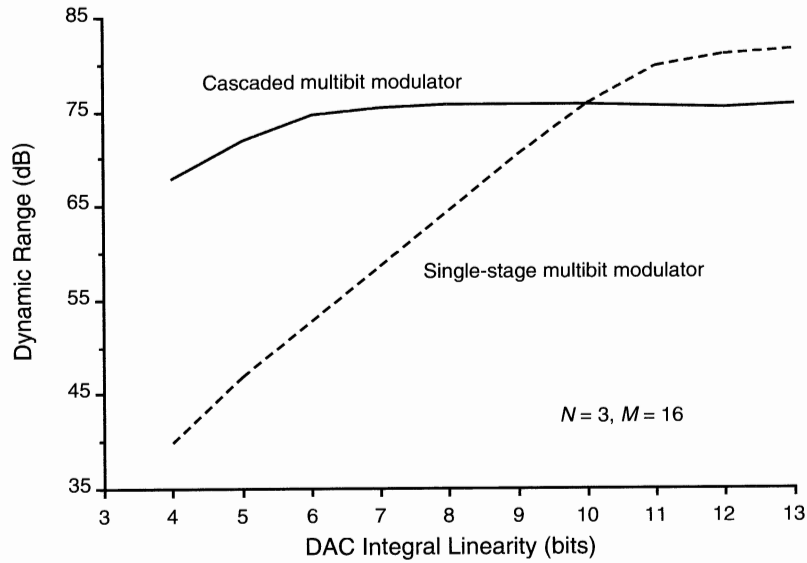


Figure 7.5 Simulated dependence of dynamic range on DAC linearity.

Figure 7.5 compares the sensitivity to DAC linearity for the cascaded and single-stage multibit modulators. To achieve 12-bit dynamic range, the cascaded modulator requires a DAC linearity of only 6 bits, while more than 10-bit linearity is needed in the single-stage modulator. Moreover, even with 10-bit DAC linearity, the single-stage modulator continues to exhibit significant harmonic distortion. The results presented in Figure 7.6 indicate that the distortion performance of the single-stage modulator is strongly dependent on the linearity of the DAC. In contrast, the harmonic power produced by the cascaded modulator is well below the power of the baseband quantization noise.

In addition to its increased tolerance for DAC nonlinearity, the cascaded multibit modulator may offer a second important advantage over a single-stage multibit modulator. Theoretical studies indicate that the quantization error spectra of single-stage modulators with multibit quantizers are free from discrete noise tones, provided that the resolution of the quantizer is sufficient to avoid overload [8–10]. However, experimental evidence of tones in a second-order modulator employing a 3-bit quantizer was recently reported [11].

The reader is referred to Chapter 3 for more information regarding tonal behavior and methods of tone elimination.

7.3.1 Interstage Coupling

In Figure 7.5 it is evident that the single-stage modulator achieves a 6-dB larger dynamic range than the cascaded modulator when the DAC linearity is greater than 12 bits. This is a consequence of a signal range reduction that must be included in the cascaded modulator. For large inputs to the cascaded modulator, the amplitude of the quantization error produced by the first stage exceeds 3.5 times the first-stage input range, which is defined by the two levels of the 1-bit DAC. Since the quantization error of the first stage is the input to the second stage, the input range of the first stage must be reduced to

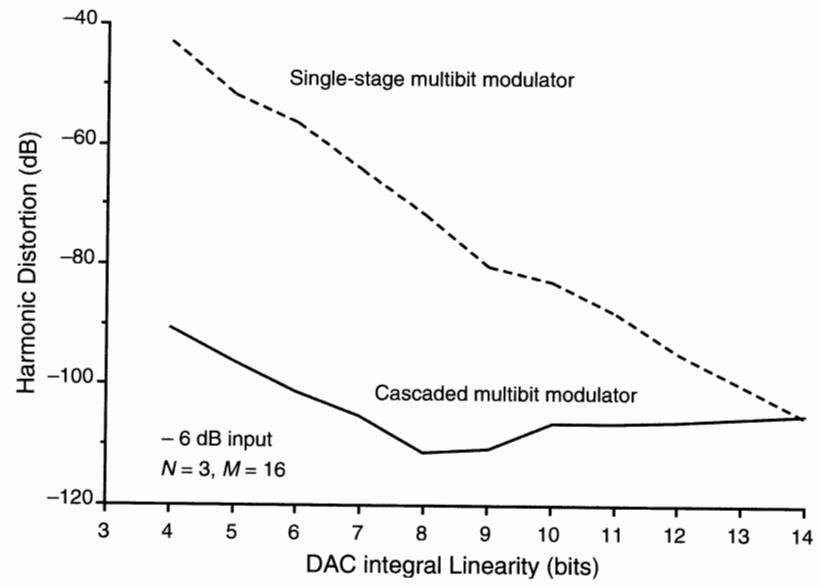


Figure 7.6 Simulated dependence of harmonic distortion on DAC linearity. Distortion power is normalized by the power of a full-scale sinusoidal input.

approximately one-fourth the input range of the second stage, which is defined by the two outermost levels of the N -bit DAC. However, such a reduction significantly increases the impact of electronic device noise in the first stage. Thus, it is preferable that both stages have the same input range in order to utilize the maximum signal swing allowed by the power supplies and also to reduce the number of required voltage references.

The two interstage coupling coefficients, α and γ , that are present in the modified cascaded multibit modulator depicted in Figure 7.7 are used to reduce the signal range at the input to the second stage so that both stages may have the same input range. When $\gamma = 0$, the input to the second stage is simply attenuated by a subunity gain factor α . To compensate for the attenuation, the digital output of the second stage must be scaled by α^{-1} before processing by the error cancellation logic. The resulting output of the modulator is then

$$Y(z) = z^{-2}X(z) + \alpha^{-1}[z^{-1}(1 - z^{-1})^2 E_D(z) - (1 - z^{-1})^3 E_2(z)] \quad (7.6)$$

Note that the effect of the error sources in the second stage [$E_D(z)$ and $E_2(z)$] are increased by α^{-1} . The performance impact of this increase is evident in Figure 7.8, which shows the simulated signal-to-(noise+distortion) ratio (SNDR) as a function of the input signal strength for three values of α . At low signal levels, a decrease in α results in a reduction in the SNDR because of the increased effect of the error sources in the second stage. However, a reduction in α also increases the input signal level at which the effects of second-stage input overload begin to degrade the SNDR. Note that it is advantageous to make α^{-1} a power of 2 so that the digital scaling of the second-stage output may be implemented with a simple shift of bit significance.

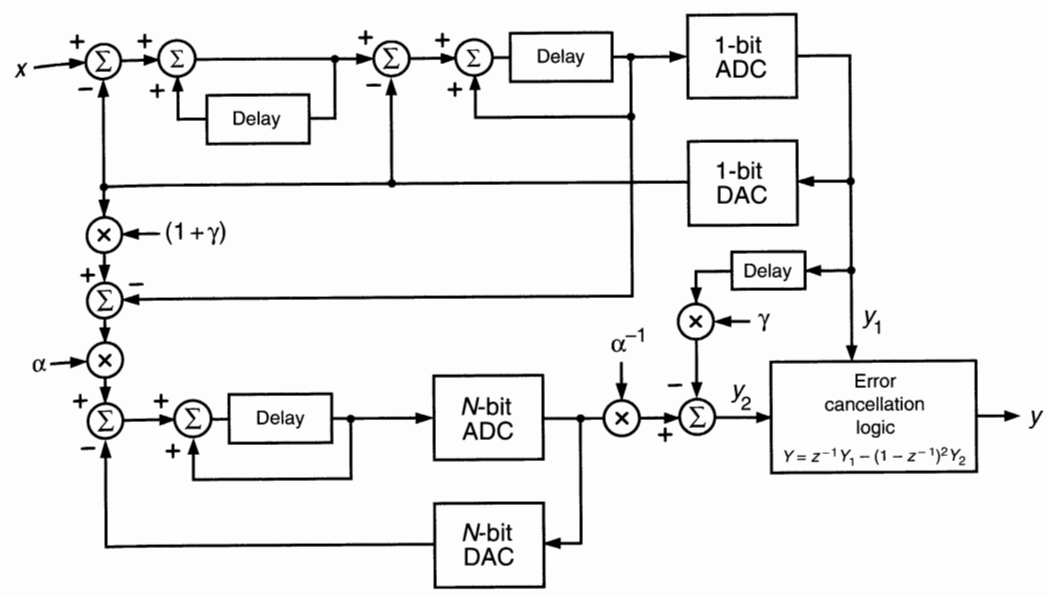


Figure 7.7 Cascaded multibit modulator with interstage coupling coefficients, α and γ .

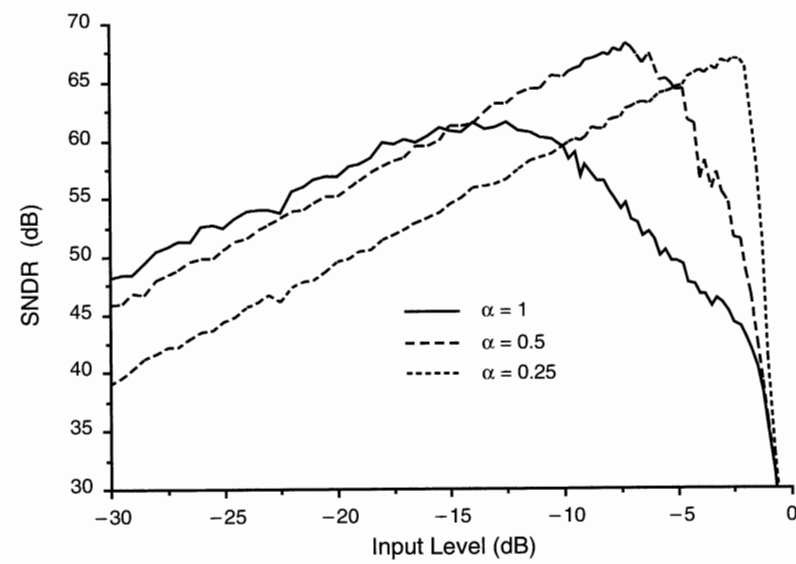


Figure 7.8 Simulated signal-to-(noise + distortion) ratio at $R = 16$.

The trade-off between small-signal and large-signal performance can be improved by adjusting γ , the second interstage coupling coefficient, to achieve a more optimal weighting of the input and output of the first-stage quantizer in the formation of the second-stage input. When γ is nonzero, a component of the analog representation of y_1 is introduced into the second-stage input. This component of y_1 must be digitally subtracted from the output of the second stage before performing the error cancellation. As such, the value of γ does not affect the combined output of the modulator, $Y(z)$, as expressed by Eq. (7.6). Therefore, γ may be optimized to constrain the signal range at the input to the second stage and thereby allow the largest possible value of α .

Figure 7.9 shows the probability density functions at the input to the second stage for a -5 -dB modulator input and four pairs of interstage coupling coefficients, α and γ . The input ranges of the first and second stages both equal $\pm 0.5\Delta_1$, where Δ_1 is the step size of the 1-bit DAC in the first stage. While all four pairs of coupling coefficients sufficiently constrain the second-stage input, setting γ to unity produces a fairly even distribution and allows the largest value of α . As a result, the $\alpha = 0.5$, $\gamma = 1$ coupling produces the best small-signal performance, as indicated in Figure 7.10. This figure shows the SNDR as a function of the input signal level for the same four pairs of coupling coefficients considered in Figure 7.9.

For input signals smaller than -4 dB, the SNDR is independent of γ , consistent with its absence in Eq. (7.6). However, the value of γ does have an impact on the large-signal performance of the modulator. For modulator inputs larger than -4 dB, the density functions shown in Figure 7.9 corresponding to couplings in which γ equals -1 , 0 , or 1 include small tails that extend slightly beyond $\pm 0.5\Delta_1$. These tails grow with increasing modulator input levels and result in the premature reduction in SNDR observed in Figure 7.10. The

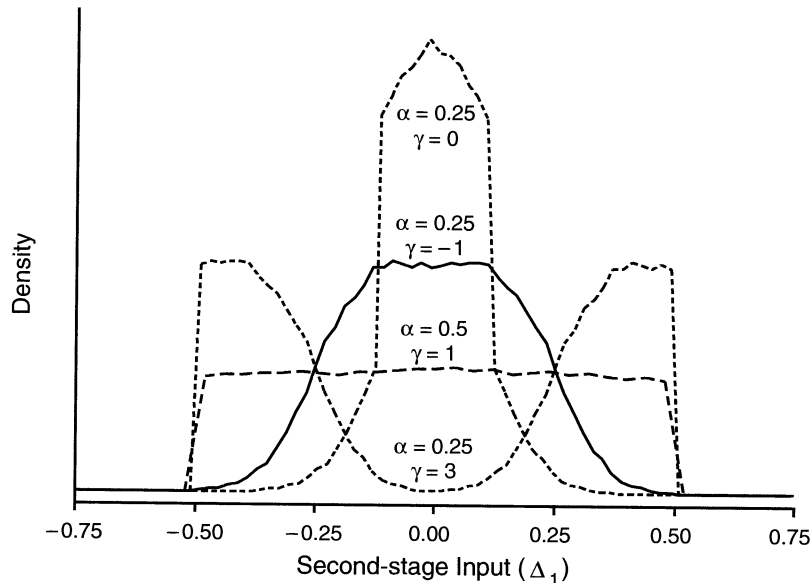


Figure 7.9 Simulated probability density functions at the input of the second stage for -5 dB modulator input.

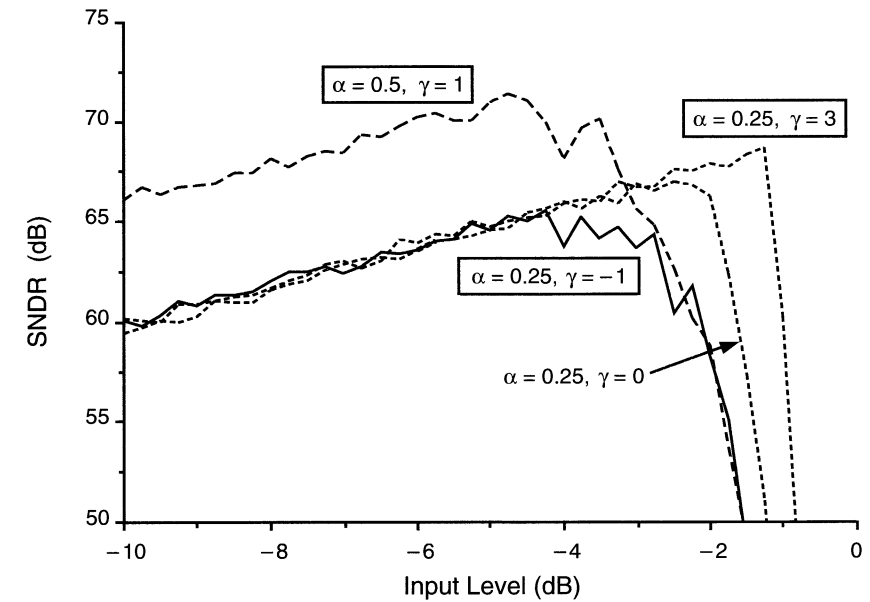


Figure 7.10 Simulated SNDR at $R = 16$ for various pairs of interstage coupling coefficients.

density function for the $\alpha = 0.25$, $\gamma = 3$ coupling does not include tails that extend beyond $\pm 0.5\Delta_1$, and as a result, this coupling provides the best performance for large input signals.

Figure 7.10 indicates that there are trade-offs between small-signal performance, large-signal performance, and hardware complexity. If $\alpha = 0.5$ and $\gamma = 1$, the dynamic range and peak SNDR are maximized. If $\alpha = 0.25$ and $\gamma = 3$, the input signal level at which the SNDR begins to decrease is maximized. Finally, if $\alpha = 0.25$ and $\gamma = -1$, the complexity of the modulator is reduced by eliminating the subtraction node at the input to the second stage. The experimental modulator described in Section 7.5 allows selection among the three pairs of coupling coefficients enclosed in boxes in Figure 7.10.

Note that the applicability of the preceding discussion concerning interstage coupling is not limited to the cascaded multibit modulator shown in Figure 7.2. Specifically, the discussion is independent of the second-stage architecture, which could be, for example, a 1-bit first-order stage or a second-order stage. Additional information concerning the selection of appropriate coupling coefficients is presented elsewhere [12].

7.4 IMPLEMENTATION OF THE CASCADED MULTIBIT MODULATOR

The performance of cascaded modulators such as those discussed in Chapter 6 and that of the cascaded multibit modulator shown in Figure 7.2 is typically more sensitive to circuit nonidealities than that of single-stage modulators because of the absence of feedback around the entire modulator. The cancellation of $E_1(z)$ depends on the noise shaping performed in the first stage precisely matching the shaping provided by the error cancellation

logic. In practical implementations, the first-stage noise shaping deviates from $1 - z^{-1}$ or $(1 - z^{-1})^2$ because of circuit nonidealities such as gain errors from capacitor mismatch as well as finite bandwidth and dc gain in the operational amplifiers.

7.4.1 Gain Error

Mismatch between the sampling and integrating capacitors results in a gain error in the transfer function of a switched-capacitor integrator:

$$H(z) = \frac{1 + \lambda}{1 - z^{-1}} \quad (7.7)$$

where λ represents the fractional gain error. As a result of this gain error, the quantization noise of the first stage is not completely canceled. Shaped first-stage quantization noise proportional to λ leaks through to the combined modulator output and degrades its performance. The resulting baseband quantization error power is

$$S_B = S_{B0} + \lambda^2 S_{B1} \quad (7.8)$$

where S_{B0} is the nominal baseband noise power in the combined modulator output in the absence of gain error and S_{B1} is the baseband error power of the first stage alone. Note that the modulator of Figure 7.2 has an advantage over a cascade of three first-order stages in that uncanceled quantization error from the first stage is attenuated in the baseband by second-order shaping rather than by only first-order shaping.

In the cascaded multibit modulator, the expected magnitude of the uncanceled quantization error from the first stage due to integrator gain error influences the resolution chosen for the multibit quantizer in the second stage. In an ideal implementation, the multibit quantizer resolution N can be increased indefinitely to improve the dynamic range according to Eq. (7.1). However, in practical implementations, increasing the resolution of the multibit quantizer reduces the second-stage quantization error and thereby increases the sensitivity to uncanceled quantization error from the first stage. A useful measure of this sensitivity is the achievable increase in dynamic range beyond that provided by the first stage alone,

$$\begin{aligned} \text{DRI} &= \left(\frac{\text{DR}_{\text{cascade}}}{\text{DR}_1} \right) \\ &= \left(\frac{S_B}{S_{B1}} \right)^{-1} \\ &= \left(\frac{S_{B0}}{S_{B1}} + \lambda^2 \right)^{-1} \end{aligned} \quad (7.9)$$

Figure 7.11 presents analytical results derived using Eq. (7.9), as well as supporting simulation results, that provide a basis for choosing the resolution of the second-stage multibit quantizer. With the quantizer resolution as a parameter, the dynamic range increase (DRI) provided by the second stage is plotted as a function of the gain error in the first-stage integrators. For example, the second-order first stage provides a dynamic range of 49 dB at an oversampling ratio of 16, according to Eq. (7.1). The cascaded multibit

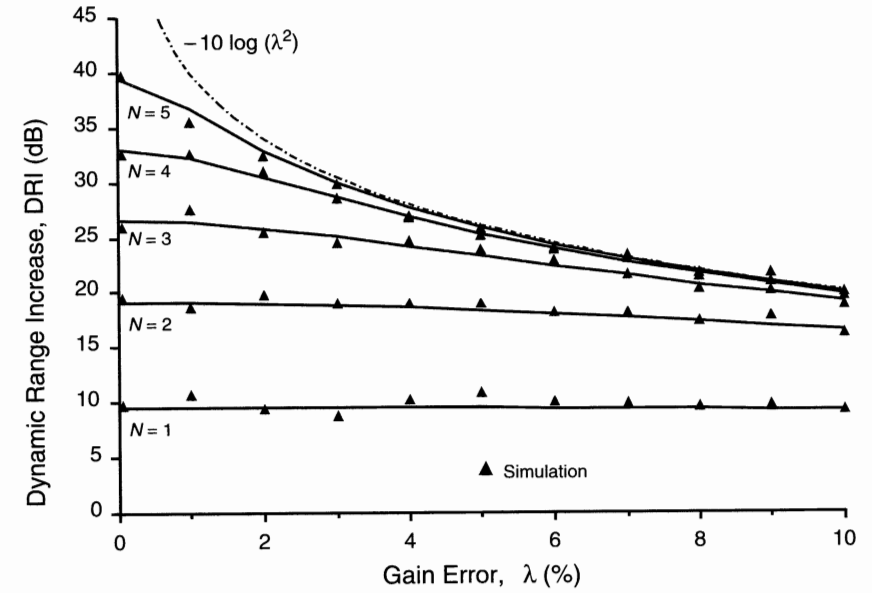


Figure 7.11 Dynamic range increase as a function of the gain error at $R = 16$.

modulator with a 3-bit quantizer in the second stage ($N = 3$) achieves a dynamic range of 75.5 dB, including the 6-dB loss due to interstage coupling with $\alpha = 0.5$. Thus, in the absence of gain error ($\lambda = 0$), the dynamic range increase provided by the second stage is 26.5 dB. In the presence of gain errors, the dynamic range increase is reduced but remains at least 26.0 dB for gain errors as large as 2%.

Several observations can be made concerning the results presented in Figure 7.11. Increasing the resolution N of the multibit quantizer in the second stage increases both the dynamic range and the sensitivity to gain error. At increasing levels of gain error, the performance of the modulator becomes dominated by uncanceled first-stage quantization error, and less benefit is derived from increasing the second-stage quantizer resolution. The use of a 1-bit quantizer in the second stage [9, 15] increases the dynamic range by less than 10 dB at low oversampling ratios, but this improvement is very insensitive to gain error. The use of a multibit second stage provides a means of trading some of this large gain error tolerance for increased dynamic range without imposing strict constraints on the precision of the multibit D/A converter. Thus, the modulator can be tailored to the expected capacitor matching of the fabrication process.

A 2% gain error margin was selected to ensure a robust implementation of the modulator in the present work. While this tolerance may be larger than required by capacitor matching considerations, it eases the requirements on other circuit nonidealities, such as incomplete linear settling of the integrator outputs. At a gain error of 2%, the benefit of a 4-bit quantizer over a 3-bit quantizer is approximately 4.5 dB and does not justify doubling the size and loading of the second-stage quantizer. Hence, a 3-bit quantizer was chosen for the second stage.

Finally, note that at higher oversampling ratios the performance of a cascaded modulator is more sensitive to uncanceled quantization error from the first stage, and the additional tolerance provided by a 1-bit second-stage quantizer may be required. Thus, the use of multibit quantization in the second stage is most attractive at low oversampling ratios.

7.4.2 Incomplete Settling

Incomplete settling of the integrator outputs due to the finite bandwidth of operational amplifiers translates into an equivalent gain error as long as the settling process is linear. As an example, consider an integrator whose transient response during each sampling period is characterized by a single-pole exponential:

$$\Delta V_{\text{out}} = V_{\text{in}}(1 - e^{-T/\tau}) \quad (7.10)$$

where T is the sampling period and τ is the settling time constant. For a constant sampling period, the term in parentheses represents a constant reduction in the gain of the integrator. A settling time as short as 4τ reduces the equivalent gain of the integrator by less than 2%. Slew rate limiting in the response of the integrator causes a departure from the linear settling characterized by Eq. (7.10) and must be avoided.

7.4.3 Integrator Leakage

The performance of a cascaded modulator may also be degraded by integrator leakage resulting from the finite dc gain of operational amplifiers. The transfer function of a leaky integrator is given by

$$H(z) = \frac{1}{1 - (1 - A_{\text{dc}}^{-1})z^{-1}} \quad (7.11)$$

where $A_{\text{dc}} = H(1)$ is the dc gain of the operational amplifier. For this case the transform of the output of the first stage given in Eq. (7.2) becomes approximately

$$Y_1(z) \cong z^{-1}X(z) + [(1 - z^{-1})^2 + 2A_{\text{dc}}^{-1}z^{-1}(1 - z^{-1}) + A_{\text{dc}}^{-2}z^{-2}]E_1(z) \quad (7.12)$$

The first term in the bracket is canceled by the error cancellation logic. However, the second and third terms are not canceled and result in first-order shaped and unshaped first-stage quantization error appearing at the combined modulator output, $Y(z)$. The first-order shaped error is inversely proportional to A_{dc} , while the unshaped error is inversely proportional to the square of A_{dc} . The resulting baseband quantization error power, relative to the power of a sine wave with a peak-to-peak amplitude of Δ_1 , is given by

$$S_B = S_{B0} + \frac{8\pi^2}{9R^3A_{\text{dc}}^2} + \frac{2}{3RA_{\text{dc}}^4} \quad (7.13)$$

where S_{B0} is the nominal baseband error power with infinite dc gain. For typical dc gain values and oversampling ratios, the contribution of the unshaped error represented by the last term in Eq. (7.13) is negligible.

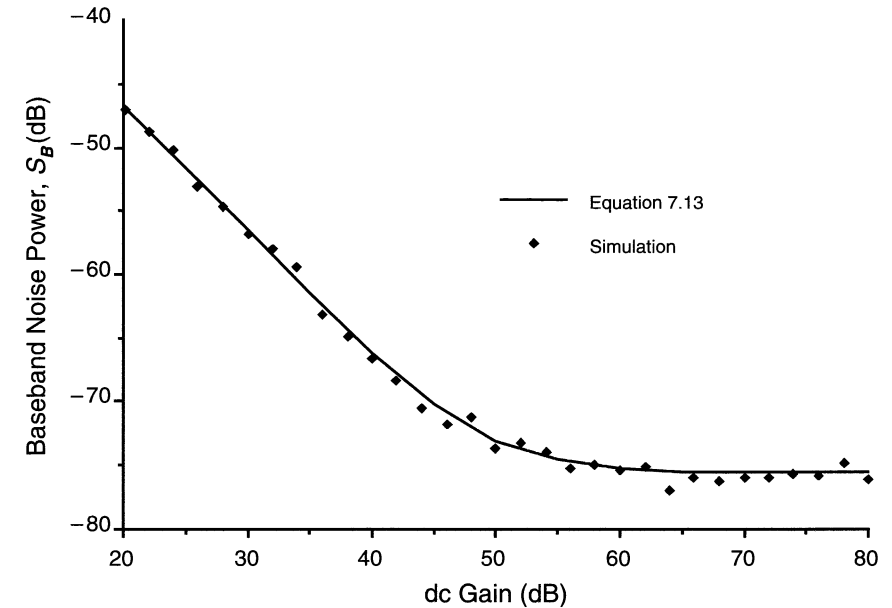


Figure 7.12 Dependence of the baseband error power on operational amplifier dc gain.

Figure 7.12 shows the dependence of the baseband quantization error power on the dc gain given by Eq. (7.13), along with supporting simulation results, for the cascaded multibit modulator. Approximately 60 dB of dc gain is required to prevent performance degradation and maintain a 12-bit dynamic range. This gain is lower than required for a cascade of three first-order stages, wherein *unshaped* first-stage quantization error inversely proportional to the dc gain is leaked to the output of the modulator [14].

Cascaded modulators are susceptible to the appearance of discrete noise tones in their output spectrum when the first-stage quantization noise is not entirely canceled because of integrator gain error or leakage [4, 5]. The second-order first stage can produce tones as strong as 52 dB below full scale at an oversampling ratio of 16 [5]. In the absence of integrator gain error and leakage, the tones are canceled. In the presence of 1 and 2% gain errors, the tones are reduced by 40 and 34 dB, respectively, to levels that are well below the -74-dB noise floor of a 12-bit converter. Integrator leakage resulting from an operational amplifier dc gain of 60 dB produces tones that are 91 dB below full scale.

7.5 DESIGN OF THE CASCADED MULTIBIT MODULATOR

The cascaded multibit $\Delta\Sigma$ modulator depicted in Figure 7.2 has been designed for fabrication in a 1- μm CMOS VLSI technology with the goal of verifying experimentally the aforementioned attributes of this modulator. The performance objective was a Nyquist conversion rate greater than 1 MHz and a dynamic range of 12 bits while operating from a single 5-V power supply.

Depicted in Figure 7.13 is a fully differential, switched-capacitor CMOS implementation of the cascaded multibit $\Delta\Sigma$ modulator. The first stage consists of two parasitic-insensitive integrators, a comparator that serves as the 1-bit ADC, and a distributed two-level (1-bit) DAC. The second stage consists of a single integrator, a 3-bit flash ADC, and a 3-bit differential DAC. The modulator operates on a two-phase nonoverlapping clock consisting of a sampling phase and an integration phase. During phase 1, the integrators sample their inputs by closing switches S_1 and S_3 and the first-stage comparator and second-stage ADC are strobed. The nodes labeled V_{cmi} set the common-mode input voltage of the fully differential operational amplifiers. Switches S_2 and S_4 conduct during phase 2 to perform the subtraction and integration functions. Switches S_3 and S_4 are opened slightly ahead of switches S_1 and S_2 , respectively, to reduce signal-dependent charge injection [15]. Note that the pipelined nature of the implementation of Figure 7.13 reduces the critical path delay to one integrator delay per clock cycle.

Two modifications of the modulator in Figure 7.2 are evident in Figure 7.13. First, both integrators in the first stage include delays in their forward paths, as well as gain factors of one-half at their inputs that are set by the ratio of their sampling and integrating capacitors. Thus, the transfer function of both first-stage integrators is

$$H(z) = \frac{1}{2} \frac{z^{-1}}{1 - z^{-1}} \quad (7.14)$$

A straightforward transformation of the first-stage architecture shown in Figure 7.2 into the form employed in Figure 7.13 results in gain factors of $\frac{1}{2}$ and 2 at the inputs of the first and second integrators, respectively. However, the decision of the single-threshold (1-bit) ADC is unaffected by a reduction in the gain preceding the second integrator. Hence, the output of the first stage given in Eq. (7.2) and the combined output of the cascaded modulator given in Eq. (7.6) are changed only slightly to include an additional delay (z^{-1}) preceding $X(z)$. Moreover, this configuration reduces the signal range required at the outputs of the first-stage integrators to about 1.7 times the modulator's input range, Δ_1 [16].

The second modification present in Figure 7.13 is that the input to the second stage is simply the differential output of the second integrator in the first stage. The combination of this simple interstage coupling and the $\frac{1}{2}$ gain factors in the first-stage integrators implements $\alpha = 0.25$ and $\gamma = -1$. In the experimental circuit two additional sampling capacitors and additional switches were included in the second-stage integrator, as shown in Figure 7.14, to allow selection among the three pairs of coupling coefficients that are enclosed in boxes in Figure 7.10. The nodes labeled V_{cmo} in Figure 7.14(b) are biased at the reference voltage used to establish the common-mode output level of the operational amplifiers.

The operational amplifier used in the integrators is the most critical element of the modulator. As noted in Section 7.4.2, incomplete settling of the integrator outputs can be considered a gain error if the settling process is linear. However, the settling speed of the operational amplifiers ultimately limits the achievable sampling rate of the modulator, even if complete settling is not required. Slew rate limiting represents a departure from linear settling and must be avoided.

The need for fast settling coupled with a relatively modest gain requirement of 60 dB to suppress harmonic distortion as well as leakage of quantization error from the first stage

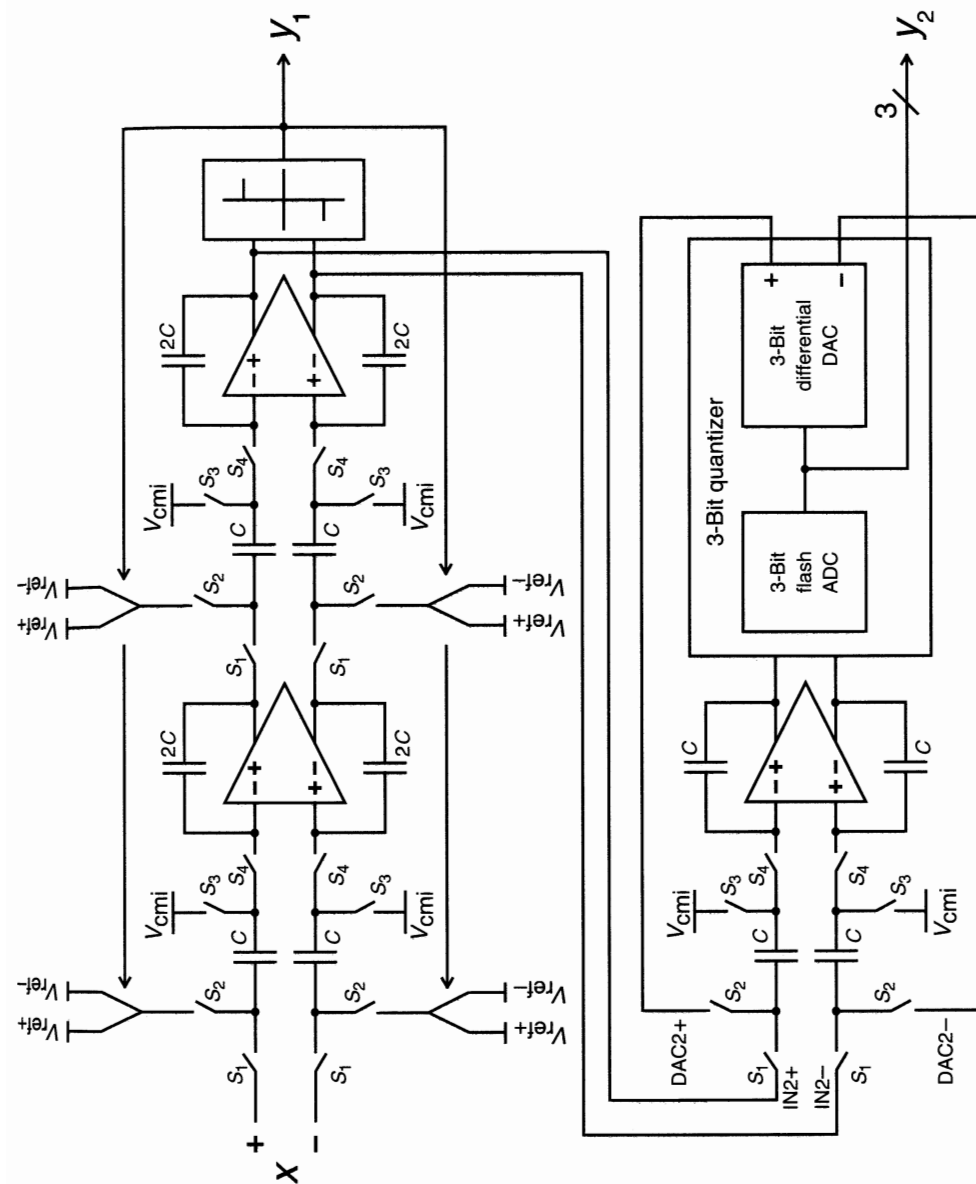


Figure 7.13 Fully differential CMOS implementation of the cascaded multibit $\Delta\Sigma$ modulator (shown for $\alpha = 0.25$ and $\gamma = -1$).

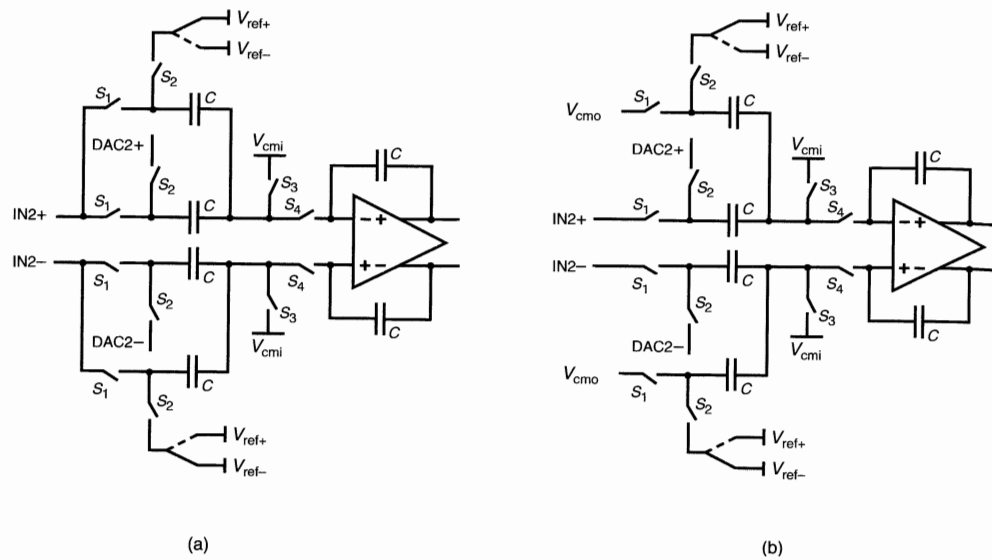


Figure 7.14 Configuration of the second-stage integrator: (a) $\alpha = 0.5$, $\gamma = 1$; (b) $\alpha = 0.25$, $\gamma = 3$.

encouraged the use of the fully differential folded-cascode operational amplifier shown in Figure 7.15 [17]. The unity-gain bandwidth of this amplifier can be quite high because the nondominant pole is set by the ratio of the transconductance of the 40/1 NMOS cascode transistors to the capacitance at their sources. Moreover, the fully differential implementation eliminates the need for PMOS current mirrors, and their associated poles, in the signal path. The 12-bit resolution objective for the cascaded multibit modulator permitted the use of small sampling capacitors. The small load capacitance resulting from the use of 200-fF sampling capacitors allows the folded-cascode amplifier to provide adequate slewing current while simultaneously satisfying gain and output range requirements.

The common-mode output level of the amplifier is maintained by the switched-capacitor feedback circuitry also shown in Figure 7.15. Capacitors C_1 and C_2 have equal value (100 fF) and form a voltage divider to drive node A, the gates of the NMOS current source transistors in the output stage. Only changes in the common-mode output are coupled to node A, which returns the common-mode output voltage to the desired level through negative feedback. During phase 1, corrective charges are transferred onto C_1 and C_2 from refresh capacitors to prevent drift in the common-mode output voltage.

The performance of the modulator is relatively insensitive to offset and hysteresis in the first-stage comparator because the effects of these impairments are attenuated in the baseband by second-order noise shaping. The regenerative latch shown in Figure 7.16 has been used to implement the comparator. In this latch the cross-coupled devices, M2A–M2B and M3A–M3B, are strobed at their drains, rather than at the sources, to eliminate backgating effects and promote faster regeneration [18, 19].

During the integration phase (phase 2) the clock input ϕ_1 is low, thereby disabling the regeneration while nodes A and B are pulled high. Nodes R and S, which are the inputs to an RS latch (not shown), are consequently both low. When the comparator is strobed by

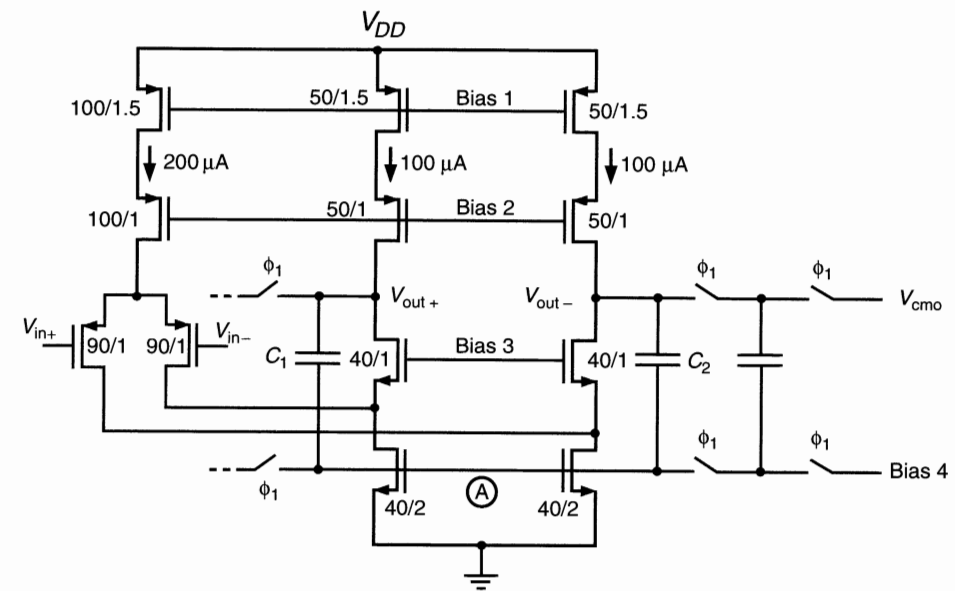


Figure 7.15 Fully differential folded-cascode operational amplifier with common-mode feedback.

bringing ϕ_1 high, nodes A and B are released and regenerative feedback is produced by the cross-coupling of transistors M2A–M2B and M3A–M3B. Differential pull-down currents developed by the input transistors M1A and M1B preferentially latch the comparator in one direction and drive one of the inputs to the RS latch high to store the result when the comparator is reset during the next clock phase.

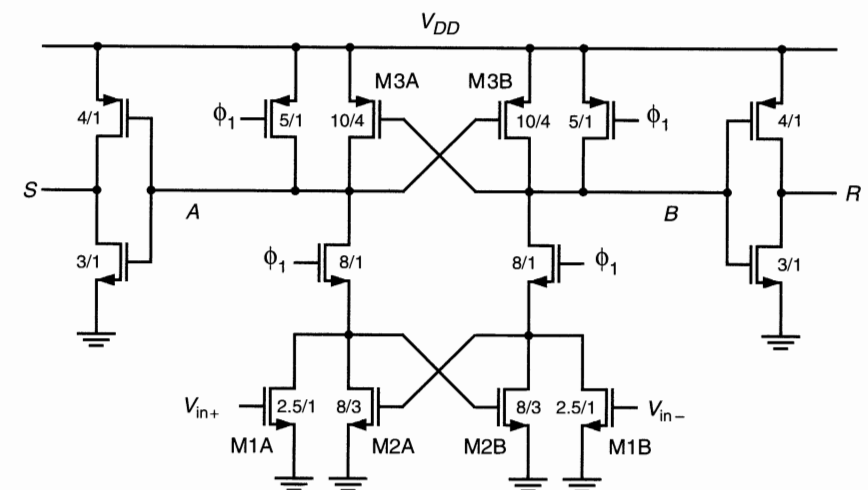


Figure 7.16 Regenerative feedback comparator.

Note that the switching threshold of the output inverters is set low intentionally. In the event that the comparator is unable to come to a decision before ϕ_1 falls, the result of the previous comparison will remain in the RS latch. Also, because no preamplification or offset cancellation circuitry precedes this latch, offset voltages were controlled by using nonminimum gate lengths in the cross-coupled transistors.

The modulator performance is also very tolerant of nonlinearity and hysteresis in the second-stage 3-bit ADC because their effects are attenuated in the baseband by third-order noise shaping. The design of this ADC is complicated by the fact that it must compare the *differential output voltage* of the second-stage integrator to seven *differential reference voltages*. This is accomplished by charging seven pairs of capacitors to unique differential voltages derived from a reference resistor string during phase 1, as shown in the left portion of Figure 7.17 [20]. During phase 2, the left sides of the capacitors are driven by the outputs of the second-stage integrator. Seven regenerative latches of the same type used for the first-stage comparator are strobed at the end of phase 2 to perform the 3-bit conversion. In the actual implementation, source followers are placed at the outputs of the second-stage integrator to buffer the loading that results from the seven comparators. Equivalent buffers are placed between the resistor string and the comparators to compensate for gain error and nonlinearity introduced by the source followers.

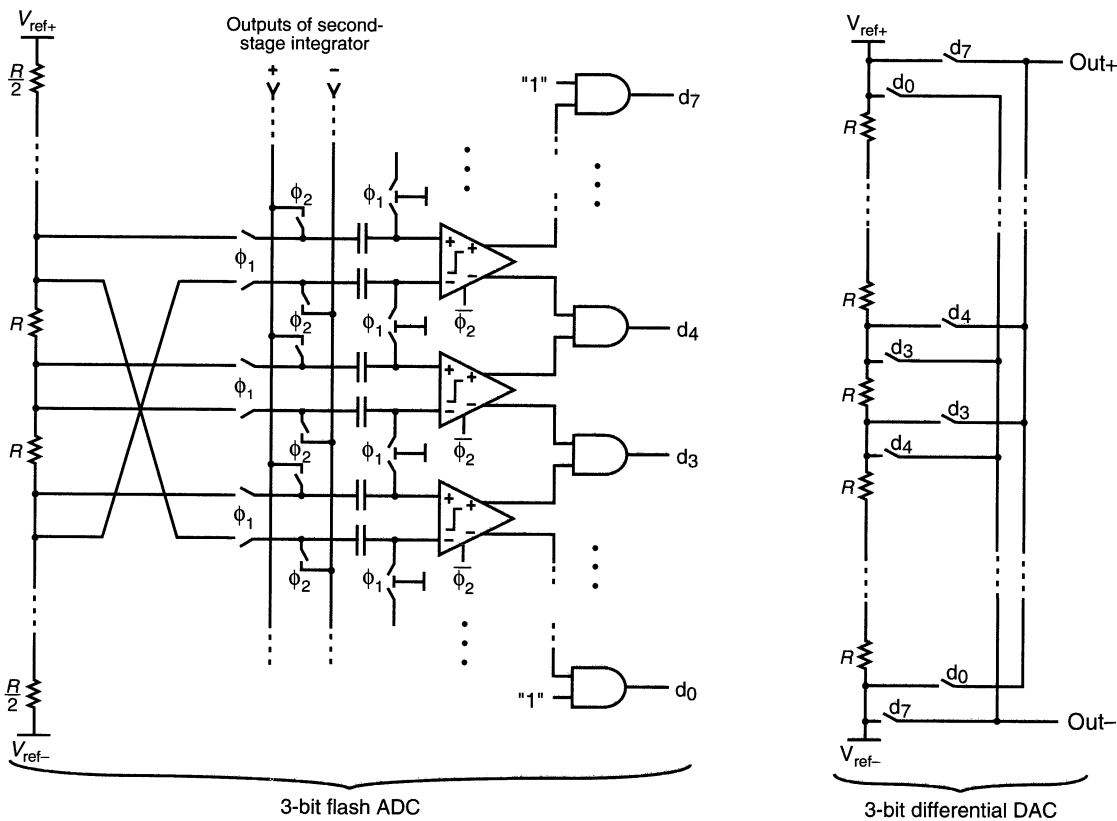


Figure 7.17 Second-stage 3-bit quantizer.

The modulator's tolerance of nonlinearity in the second-stage 3-bit DAC permits the use of a simple differential tapped resistor string for its implementation, as is also illustrated in Figure 7.17. A 1-out-of-8 code produced by the AND gates in the 3-bit flash ADC selects the proper pair of taps from the resistor string. The 1-out-of-8 code is also converted into the 3-bit binary output of the second stage by simple encoding circuitry (not shown).

7.6 EXPERIMENTAL RESULTS

The cascaded multibit $\Delta\Sigma$ modulator has been fabricated in a 1- μm CMOS technology with metal-to-polycide capacitors and polysilicon resistors [21]. The modulator has an area of 0.65 mm² and operates from a single 5-V supply with a power dissipation of 41 mW. The performance of the modulator was evaluated by driving its input with a high-quality differential sinusoidal signal source [22], acquiring the digital outputs from the first and second stages, and transferring the acquired data to a workstation for processing. The test fixturing is described in detail in Appendix A of [4]. The simple digital processing required to cancel the quantization error of the first stage, as well as decimation filtering and signal analysis, was performed on a workstation.

Figure 7.18 shows the measured SNDR as a function of the input sine-wave amplitude. An input level of 0 dB represents a sine wave whose peak-to-peak amplitude equals the input range of the modulator, Δ_1 , which is 3 V (differential) in this implementation. The frequency of the input sine wave was 100 kHz and the modulator sampling rate was

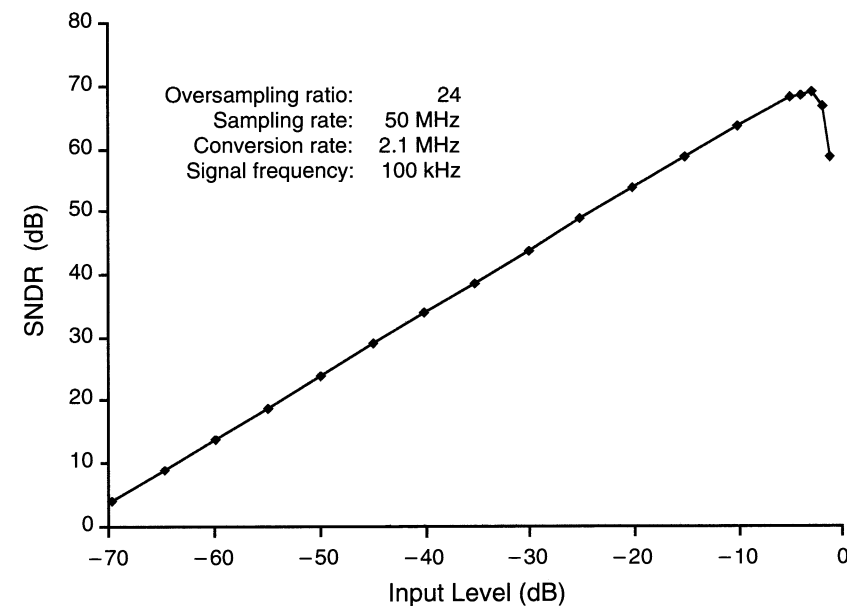


Figure 7.18 Measured signal-to-(noise + distortion) ratio for a 1-MHz baseband.

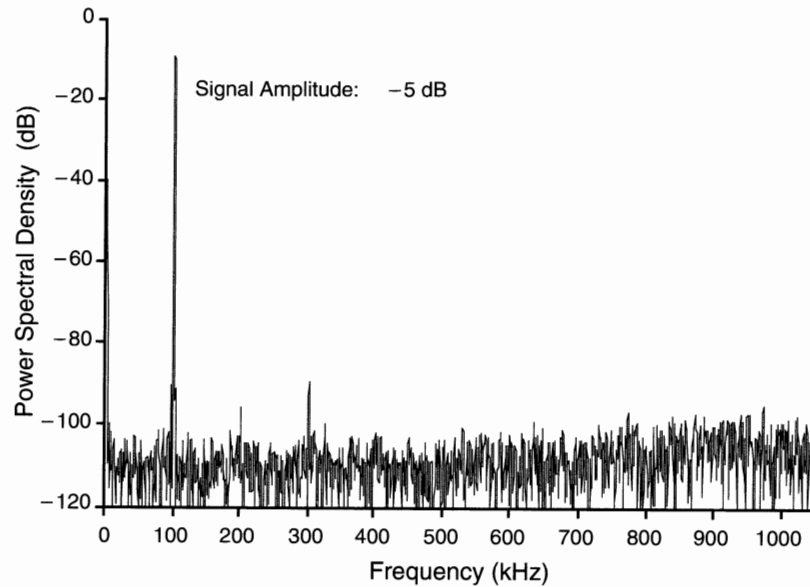


Figure 7.19 Measured baseband spectrum.

50 MHz, which produced a 2.1-MHz Nyquist conversion rate and a 1-MHz signal bandwidth at an oversampling ratio of 24. The modulator achieves a 74-dB dynamic range and a peak SNDR of 69 dB.

The measured baseband spectrum for a 100-kHz sine-wave input is shown in Figure 7.19. This spectrum was obtained by windowing the output data of the decimation filter with a window defined in Eq. (33) of [23] and then performing a 4096-point fast Fourier transform. The small second and third harmonic components present in the spectrum also appear in the output spectrum of the first stage and are therefore not attributable to nonlinearity in the multibit D/A converter in the second stage. The linearity of the metal-to-polycide capacitors was far better than required in this application, as established earlier [5]. The noise floor is higher than predicted from thermal noise generated by the sampling switches onto the 200-fF sampling capacitors in the first integrator. The noise floor is also higher than calculated for thermal and flicker noise in the operational amplifiers. Quantization noise limitations alone should allow a dynamic range exceeding 80 dB at an oversampling ratio of 24.

Both the noise floor and the harmonic distortion were quite sensitive to changes in package and test board features such as the grounding configuration, the location of decoupling capacitors, and the loading on the output pins. A second generation of the test board, designed around a leadless chip carrier instead of a 40-pin dual in-line package (DIP), extended the sampling rate from 36 to 50 MHz. There was also strong evidence of substrate coupling to the analog modulator from the output digital pad drivers. On-chip circuitry performed a serial-to-parallel conversion to slow the output data rate by a factor of 4. However, this increased the number of output pad drivers from 4 to 16. The voltage swing of these CMOS drivers had to be reduced to 1.5 V to minimize the coupling to the sensitive analog circuitry.

TABLE 7.1 CASCADED MULTIBIT $\Delta\Sigma$ MODULATOR PERFORMANCE SUMMARY

Dynamic range	74 dB (12 bits)
Peak SNDR	69 dB
Sampling rate	50 MHz
Oversampling ratio	24
Conversion rate	2.1 MHz
Signal bandwidth	1 MHz
Differential input range	3 V
Supply voltage	5 V
Power dissipation	41 mW
Area	0.65 mm ²
Technology	1- μ m CMOS

As mentioned previously, the experimental modulator was designed to allow selection among the three pairs of coupling coefficients enclosed in boxes in Figure 7.10. Experimentally, there was less than a 2-dB difference in dynamic range among the three pairs of coupling coefficients because the modulator's performance was not limited by quantization noise. The results presented in Figures 7.18 and 7.19 were obtained for $\alpha = 0.25$ and $\gamma = -1$ since this coupling simplifies the implementation of the modulator by eliminating the need for the subtraction node at the input of the second stage.

The performance of the modulator degrades above a sampling rate of 50 MHz principally because of incomplete settling of the integrator outputs. At 50 MHz, the integrator outputs have about 3.5 time constants in which to settle. This is equivalent to a gain error of 3% if the settling is linear. Thus, at sampling rates exceeding 50 MHz, the leaked quantization noise from the first stage increases and surpasses the noise due to the test fixture, packaging, and substrate coupling. The frequency of the sine-wave input was limited to 100 kHz by the signal generator.

Key performance parameters for the cascaded multibit modulator and the operational amplifier are summarized in Tables 7.1 and 7.2, respectively. The operational amplifier performance measurements were obtained from isolated test structures. Measurements of the unity-gain frequency, slew rate, and settling time constant were obtained using low-capacitance active probes [24].

TABLE 7.2 MEASURED OPERATIONAL AMPLIFIER PERFORMANCE SUMMARY

dc Gain	58 dB
Unity-gain frequency	200 MHz
Settling time constant	2.5 ns
Slew rate	350 V/ μ s
Linear output range	4.5 V

7.7 SUMMARY

The implementation of CMOS oversampling ADCs with conversion rates exceeding 1 MHz is facilitated by lowering the oversampling ratio. Delta-sigma modulators based on multibit quantization are particularly attractive at low oversampling ratios, but they may impose stringent linearity requirements on DAC linearity. The cascaded multibit modulator presented here avoids the dependence on DAC linearity by placing the multibit quantizer in the second stage, where the effects of DAC nonlinearity are attenuated by second-order noise shaping. The modulator also avoids the need for additional circuitry to perform digital correction, calibration, or random element swapping.

An experimental implementation of the proposed modulator has demonstrated that oversampling analog-to-digital converters can achieve 12-bit resolution at conversion rates exceeding 2 MHz in a 1- μ m CMOS technology. The advantages of using an oversampling approach are evident in this modulator's ability to achieve 12-bit resolution without requiring more than 6-bit precision in any of the analog circuit components. While the sampling rate of the modulator is limited by the settling of the integrator outputs, the resolution of the modulator could possibly be extended through improved packaging and test fixturing as well as better isolation techniques to reduce substrate coupling. Such improvements could allow the modulator to achieve 12-bit resolution at an oversampling ratio of 16.

REFERENCES

- [1] Y.-M. Lin, B. Kim, and P. Gray, "A 13-b 2.5-MHz self-calibrated pipelined A/D converter in 3- μ m CMOS," *IEEE J. Solid-State Circuits*, vol. 26, pp. 628–636, April 1991.
- [2] R. Koch, B. Heise, F. Eckbauer, E. Englehardt, J. Fisher, and F. Parzefall, "A 12-bit sigma-delta analog-to-digital converter with a 15-MHz clock rate," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 1003–1010, Dec. 1986.
- [3] U. Roettcher, H. Fiedler, and G. Zimmer, "A compatible CMOS-JFET pulse density modulator for interpolative high-resolution A/D conversion," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 446–452, June 1986.
- [4] B. Brandt, "Oversampled Analog-to-Digital Conversion," Ph.D. Dissertation, Stanford University, Aug. 1991.
- [5] B. Brandt, D. Wingard, and B. Wooley, "Second-order sigma-delta modulation for digital-audio signal acquisition," *IEEE J. Solid-State Circuits*, vol. 26, pp. 618–627, April 1991.
- [6] J. Fattaruso, S. Kiriaki, G. Warwar, and M. de Wit, "Self-calibration techniques for a second-order multibit sigma-delta modulator," *ISSCC Dig. Tech. Papers*, pp. 228–229, February 1993.
- [7] L. Longo and M. Copeland, "A 13 bit ISDN-band oversampled ADC using two-stage third order noise shaping," *Proc. IEEE 1988 Custom Integrated Circuits Conf.*, pp. 21.2.1–4, May 1988.
- [8] R. Gray, "Quantization noise spectra," *IEEE Trans. Inform. Theory*, vol. 36, pp. 1220–1244, Nov. 1990.
- [9] N. He, A. Buzo, and F. Kuhlmann, "Multi-loop sigma-delta quantization: spectral analysis," *Proc. ICASSP*, pp. 1870–1873, 1988.

- [10] N. He, F. Kuhlmann, and A. Buzo, "Double-loop sigma-delta modulation with dc input," *IEEE Trans. Inform. Theory*, vol. 38, pp. 487–495, April 1990.
- [11] J. Fattaruso, S. Kiriaki, G. Warwar, and M. de Wit, "Self-calibration techniques for a second-order multibit sigma-delta modulator," *IEEE J. Solid-State Circuits*, vol. 28, Dec. 1993.
- [12] L. Williams and B. Wooley, "Third-order cascaded sigma-delta modulators," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 489–498, May 1991.
- [13] D. Ribner, R. Baertsch, S. Garverick, D. McGrath, J. Krisciunas, and T. Fuji, "16b third-order sigma-delta modulator with reduced sensitivity to nonidealities," *ISSCC Dig. Tech. Papers*, pp. 66–67, Feb. 1991.
- [14] M. Rebeschini, N. van Bavel, P. Rakers, R. Greene, J. Caldwell, and J. Haug, "A 16-b 160-kHz CMOS A/D converter using sigma-delta modulation," *IEEE J. Solid-State Circuits*, vol. 25, pp. 431–440, April 1990.
- [15] K. Lee and R. Meyer, "Low-distortion switched-capacitor filter design techniques," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1103–1113, Dec. 1985.
- [16] B. Boser and B. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1298–1308, Dec. 1988.
- [17] T. Choi, R. Kaneshiro, P. Gray, W. Jett, and M. Wilcox, "High-frequency CMOS switched-capacitor filters for communications application," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 652–664, Dec. 1983.
- [18] A. Yukawa, "A CMOS 8-bit high speed A/D converter IC," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 775–779, June 1985.
- [19] J. Wu, "High-Speed Analog-to-Digital Conversion in CMOS VLSI," Ph.D. Dissertation, Stanford University, March 1988.
- [20] S. Lewis and P. Gray, "A pipelined 5 MHz 9b analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 954–961, Dec. 1987.
- [21] C. Kaya, H. Tigelaar, J. Paterson, M. de Wit, J. Fattaruso, D. Hester, S. Kiriaki, K. Tan, and F. Tsay, "Polycide/metal capacitors for high precision A/D converters," *IEDM Tech. Dig.*, pp. 782–783, Dec. 1988.
- [22] Tektronix, *SG505 Option 02 Oscillator Instruction Manual*, Tektronix Inc., Beaverton, OR, Nov. 1982.
- [23] A. Nuttall, "Some windows with very good sidelobe behavior," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-29, pp. 84–91, Feb. 1981.
- [24] G. Industries, *Model 19 Picoprobe Operating Instructions*, GGB Industries Inc., Naples, FL, 1990.

Richard L. Carley
Richard Schreier
Gabor C. Temes

Delta-Sigma ADCs with Multibit Internal Converters

8.1 INTRODUCTION

One-bit noise-shaping modulators have achieved popularity for use in integrated circuit data converters [1–4]. In part, their attractiveness for IC systems that incorporate digital filtering and signal processing with analog-to-digital and/or digital-to-analog conversion is due to the fact that they employ a 1-bit internal DAC that does not require precision component matching. Delta-sigma modulators can be implemented using a standard digital CMOS process without the economically costly addition of precision thin-film resistors or the use of laser trimming. However, as was shown in Chapter 4, the resolution that a 1-bit $\Delta\Sigma$ modulator can achieve at a given oversampling ratio is limited. Although the achievable resolution does improve with increasing loop filter order, these improvements diminish rapidly due to instability. In addition, because of the substantial out-of-band quantization noise power in $\Delta\Sigma$ modulators, the design of analog output filters for oversampled DACs can be quite difficult [5]. One solution to the above problems is to use a multibit quantizer in the oversampled converter loop.

The primary advantage of noise-shaping modulators employing multibit quantizers is that the ratio of the total quantization noise power to the signal power at the modulator's output is dramatically reduced from that of a 1-bit modulator; typically by 6 dB per additional bit. Therefore, we can increase the overall resolution of any oversampled data converter, without increasing the oversampling ratio, simply by increasing the number of levels in the internal data converters. Equivalently, the multibit noise-shaping coder can achieve resolution comparable to that of a single-bit modulator at a lower sample rate. For example, the prototype noise-shaping DAC presented in [6], which operated at 3.2 MHz and employed a 3-bit internal quantizer, achieved performance comparable to that of the $\Delta\Sigma$ modulator pre-

sented in [2], which operated at 11.3 MHz and employed a 1-bit internal quantizer. This performance increase can be a significant advantage in applications requiring high bandwidth; for example, digitizing video signals. Another advantage of the lower clock rate possible with multibit modulators is the decreased power consumption in the digital circuitry [7].

The same decrease in the quantization noise power that improves resolution also relaxes the requirements on the output filter that must remove the out-of-band quantization noise power. The use of a multibit internal quantizer in the oversampled feedback loop also facilitates the design of feedback loops with high-order transfer functions because the low-frequency oscillations sometimes observed in higher order $\Delta\Sigma$ modulators [8–10] are a result of the 1-bit quantizer being overloaded [10, 11]. As Chapter 14 will explain in detail, a quantizer can be made overload free, for a given modulator and input magnitude, if it has enough levels. Assuming that quantizer overload does not occur, the design of multibit noise-shaping loops is quite simple compared to the design of 1-bit $\Delta\Sigma$ modulators. This simplicity is realized because the gain of the quantizer is known (1 LSB per digital level), because the quantization error is bounded between $+\frac{1}{2}$ LSB and $-\frac{1}{2}$ LSB and because the quantization error can be accurately modeled as an additive noise that is independent of the input signal [12–14]. Nonlinear numerical techniques have been applied to optimize the conversion system's performance for a given internal DAC through the choice of the loop filter pole and zero locations [12, 13].

A notable disadvantage of multibit systems is that they lack the ability of single-bit systems to achieve excellent integral linearity without the use of matched components. The integral linearity of a noise-shaping conversion system is no better than the integral linearity of the multibit internal DAC [1, 7, 15]. Therefore, achieving high integral linearity and low total harmonic distortion (THD) appears to require precisely matched components. As the smallest component mismatch that can be achieved is on the order of 0.1–0.5% in the inexpensive CMOS IC fabrication technologies normally employed for consumer electronics [16–18], the harmonics created by multibit modulators can approach –60 dB relative to a full-scale fundamental. A secondary disadvantage of multibit modulators is that more analog circuitry, which is generally more difficult to design than digital circuitry, is required.

In this chapter we consider a number of alternative approaches to achieving high integral linearity while requiring only modest component matching—at a level substantially lower than the required integral linearity. The approaches that will be considered run the gamut from circuit techniques for electronic trimming of element values to digital characterization and correction of element mismatches to interconnections of multiple modulator loops that achieve multibit performance.

8.2 MULTIBIT NOISE-SHAPING MODULATOR ARCHITECTURES

Noise-shaping modulators employing multibit internal quantizers can be used for both ADCs and DACs (see Figure 8.1).

In the case of an ADC system, the quantizer is a true ADC in its own right and is commonly implemented with a bank of comparators. In the case of a DAC system, the quantizer merely corresponds to a truncation of the digital word at the output of the accumulator. In both systems, a multibit DAC is necessary. As shown in Figure 8.1, the DAC

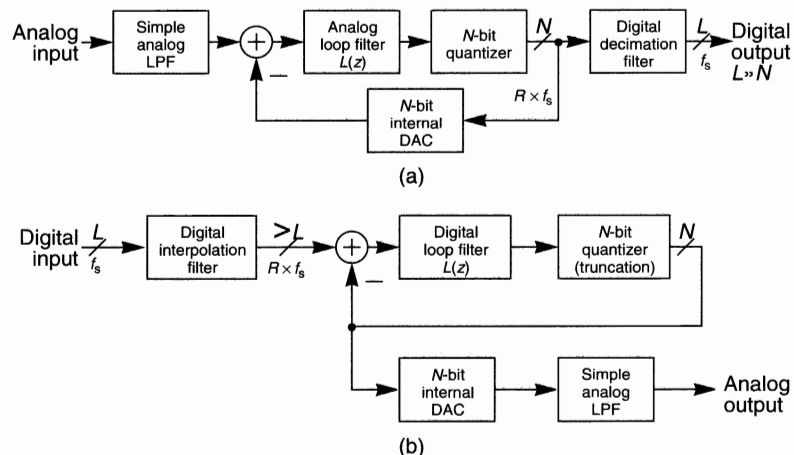


Figure 8.1 Oversampled multibit (a) ADC and (b) DAC block diagrams.

is inside the feedback loop for a multibit ADC system, whereas for a multibit DAC system the DAC is outside the feedback loop. In order to reduce system complexity, there are generally only a few quantizer and DAC levels: 4–16 levels (2–4 bits) are typical.

A notable limitation of multibit systems is that the property of perfect linearity characteristic of single-bit systems is lost. We can model the nonlinearities in an ADC system as additive noise sources, as shown in Figure 8.2. The quantizer is replaced by two additive noise sources. The one labeled $e(n)$ represents the quantization errors of an ideal converter while $c(n)$ represents the errors caused by the deviation of the comparator switching thresholds from their ideal values. Similarly, $d(n)$ represents the errors due to the deviation of the internal DAC outputs from their ideal values. For noise shaping to occur, the gain of $L(z)$ must be large at low frequencies. Therefore, both quantization errors $c(n)$ and $e(n)$ are reduced by this large gain when referred back to the input $u(n)$. However, the nonlinearity of the internal DAC, $d(n)$, resides in the feedback path where its nonlinearities due to mismatches in levels are not reduced by the negative feedback. Similarly, DAC systems have an M -bit DAC sitting outside of the noise-shaping loop where its nonlinearities clearly are not mitigated by negative feedback. Thus, the ultimate linearity of both

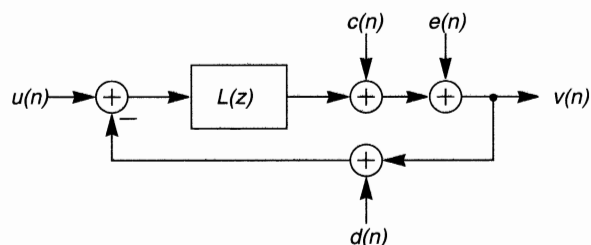


Figure 8.2 Simplified block diagram of an oversampled ADC system with sources indicating the quantization error, internal ADC nonlinearity, and internal DAC nonlinearity.

ADC and DAC systems is no better than the linearity of the N -bit internal DAC. For example, 16-bit linearity in an oversampled DAC system can only be achieved if the internal DAC has its levels placed with an accuracy better than 1 part in 100,000. Although this degree of matching can be obtained by careful trimming after fabrication [19, 20] (e.g., by laser trimming of resistors), architectural changes and circuit design techniques [21, 22, 23] that avoid the need for this high degree of matching may result in substantially lower manufacturing costs. Because it is the internal DAC that controls the performance, we will concentrate on its design and ignore the design of the internal multibit ADC, which is usually implemented as a parallel bank of comparators (i.e., as a flash ADC [20]).

8.3 DAC ARCHITECTURES FOR IMPROVED LINEARITY

In this section, we consider the design of the internal DAC for multibit systems. First, the most common architecture for internal DACs will be described and its overall accuracy will be derived. Then, various approaches for improving the overall accuracy of the internal DACs will be described.

8.3.1 Internal DAC Topology

Although there are a great variety of circuit topologies that can be used to implement a DAC [20], one common architecture [1, 6, 24, 25] employs 2^N parallel unit elements of approximately equal value, where N is the number of bits (see Figure 8.3). Note that an N -bit parallel-unit-element DAC can actually be implemented using only $2^N - 1$ elements since the possible digital output levels range from 0 elements being active up to $2^N - 1$ elements being active. However, because it simplifies the analysis, we will add one extra unit element to bring the number of elements in an N -bit parallel-unit-element DAC up to 2^N . In a parallel-unit-element DAC, the K th output level is generated by activating K approximately equal-valued elements (typically resistors, transistor current sources, or capacitors) and summing up their charges or currents. The novel characteristic of the internal DAC is that it requires relatively few output levels, but these output levels must be

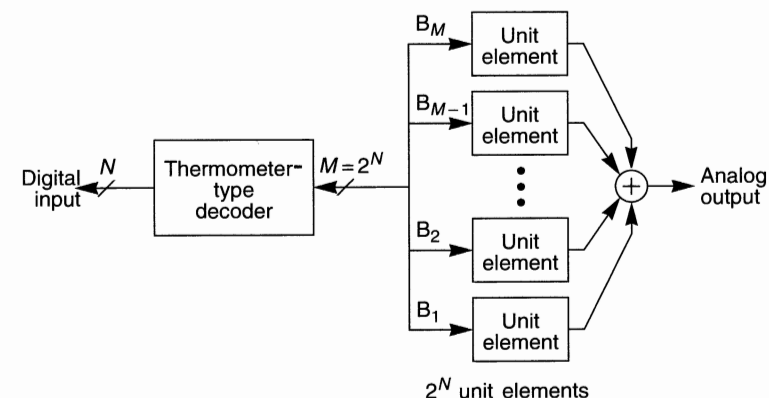


Figure 8.3 Block diagram of parallel-unit-element DAC.

extremely accurate. This requirement is quite different from a standard DAC in which the required accuracy is on the order of $\pm \frac{1}{2}$ LSB. In addition, because the internal DAC must operate at the clock frequency of the oversampled converter, it is advantageous to select a DAC topology that is capable of high-speed operation. These characteristics are all well suited to implementation using 2^N parallel unit elements. One would probably never consider implementing a 12-bit converter using this topology, as it would require 4096 parallel unit elements. However, implementing a 4-bit converter, even one with output levels having 12-bit accuracy, only requires 16 unit elements. In addition, the overall accuracy of the parallel-unit-element DAC topology is substantially greater than the accuracy of an individual unit element.

Assuming that all of the element values are drawn from an identical Gaussian probability distribution having a mean E and a standard deviation of ΔE , the standard deviation of the output voltage for a digital code word K can be expressed as

$$\Delta V_O = (\Delta E)\sqrt{K} \quad (8.1)$$

For an N -bit internal DAC, there will be $M = 2^N$ individual unit elements and the full-scale output will be $M \times E$. The standard deviation of the output voltage can be expressed as a fraction of the full-scale voltage:

$$\frac{\Delta V_O}{V_O} = \frac{(\Delta E)\sqrt{K}}{E \times M} \quad (8.2)$$

The greatest error comes for the largest value of K , which is M . However, in most data conversion applications, a slight error in the overall scale factor is not important. Instead, deviation of each output value from a best fitting straight line is commonly used. In this case, let us assume that the zero output and the full-scale output ($M \times E$) define the line from which we will measure the error at each output code. The deviation from this line is

$$\Delta V_O(K) = \left(\sum_{i=1}^K E_i \right) - \left(-\frac{K}{M} \left(\sum_{i=1}^M E_i \right) \right) \quad (8.3)$$

which we can rewrite as

$$\Delta V_O(K) = \frac{M-K}{M} \left(\sum_{i=1}^K E_i \right) - \left(\frac{K}{M} \left(\sum_{i=K+1}^M E_i \right) \right) \quad (8.4)$$

We can divide both sides of this equation by the nominal value of the full-scale voltage, $M \times E$, in order to express the fractional error of the output voltage in terms of the fractional error of the individual elements:

$$\frac{\Delta V_O(K)}{V_O(M)} = \frac{M-K}{M^2} \left(\sum_{i=1}^K \frac{\Delta E_i}{E} \right) - \left(\frac{K}{M^2} \left(\sum_{i=K+1}^M \frac{\Delta E_i}{E} \right) \right) \quad (8.5)$$

Under the assumption that all of the element values are independent samples of a normal probability density, the variance of the output voltage, as a fraction of the nominal full-scale output voltage, is given by

$$\sigma^2 \left[\frac{\Delta V_O(K)}{V_O(M)} \right] = \left(\frac{M-K}{M^2} \right)^2 K \sigma^2 \left[\frac{\Delta E_i}{E} \right] + \left(\frac{K}{M^2} \right)^2 (M-K) \sigma^2 \left[\frac{\Delta E_i}{E} \right] \quad (8.6)$$

This expression can be simplified to get the “gain” between the standard deviation of the element fractional mismatch and the standard deviation of the internal DAC’s output as a fraction of nominal full scale:

$$\sigma \left[\frac{\Delta V_O(K)}{V_O(M)} \right] = \sqrt{\frac{1}{M} \frac{K(M-K)}{M}} \sigma \left[\frac{\Delta E_i}{E} \right] \quad (8.7)$$

Note that the variance of the output voltage is a parabolic function of the digital input code. It goes to zero at both zero output and full-scale output (because we chose the straight line between those two points as our reference for measuring errors) and it rises to a maximum when $K = M/2$. Since the accuracy of the DAC must be met at all possible output codes, we need to set K to the worst-case value. The worst-case standard deviation of the normalized output then simplifies to

$$\sigma \left[\frac{\Delta V_O(\text{Worst})}{V_O(M)} \right] = \frac{1}{2\sqrt{M}} \sigma \left[\frac{\Delta E_i}{E} \right] \quad (8.8)$$

The worst-case improvement in accuracy between the elements and the overall DAC is therefore $2\sqrt{M}$. Note that the $\sqrt{M}$ term is the expected improvement for formulating an output that is the sum of M independent random variables. We show the distortion that results from this element mismatch for a 0.1% gradient error in Figure 8.4. In particular, the spike in Figure 8.4(b) represents the dominant second harmonic distortion component, which is only about -70 dB relative to the input signal.

8.3.2 Element-Trimming Approaches

One straightforward approach to improving the accuracy of the internal DAC is to improve the matching of the individual elements. Approaches of this type can generally be divided into two distinct groups: one-time trims which are part of the manufacturing process and repeated trims which are carried out continuously during the operation of the internal DAC. Note that the approaches described in this section are not specific to internal DACs of oversampled data converters. They are generally applied to many different types of DACs and ADCs in order to improve the matching of their elements. Common elements that are used are resistors, capacitors, and current sources.

8.3.2.1 One-Time Trimming Methods Many approaches exist for trimming the values of elements in DACs. One of the most common approaches for improving the accuracy of DACs is laser trimming of resistors [20]. For example, laser trimming of resistors is commonly used in the fabrication of 16-bit DACs for digital audio applications [19]. Trimming of capacitors is typically done by switching in or out very small capacitors in parallel with the capacitor being trimmed [22]. This trimming can be done at the factory and the settings of the switches can be stored in some form of programmable read-only memory (PROM), for example, fusible links or erasable prom (EPROM). When the trim

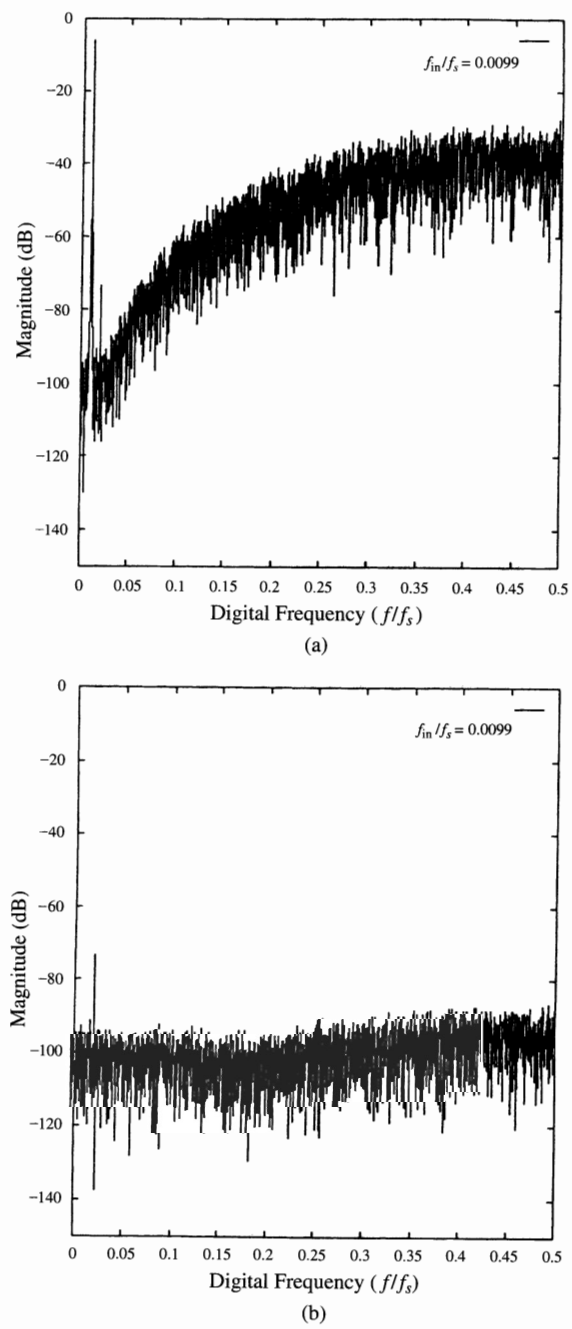


Figure 8.4 Power spectral density of (a) the signal at the output of a third-order oversampled A/D conversion system using 3-bit internal DAC and (b) of only the internal DAC error. The input signal is -6 dB below full scale. Elements of the internal DAC have a systematic linear gradient mismatch of 0.1% [25].

signal is an analog voltage, it can be permanently stored as a charge on a CMOS floating gate [23, 26]. The major advantage of one-time trimming methods is that sophisticated off-chip test equipment can be used to determine the best possible trim value for each element so that little or no extra on-chip circuitry needs to be added to the DAC for element trimming. There are two drawbacks to one-time trimming methods. First, variations in matching with temperature, power supply voltage, age, and so on, cannot be compensated for by this type of trimming. Second, techniques such as laser trimming can add significantly to the cost of an IC.

8.3.2.2 Repeated Trimming Methods There are many different types of repeated trimming methods. Some of them are performed each time power is applied. Other repeated trimming methods are performed periodically during operation. The primary requirement of repeated trimming methods is that there be some on-chip hardware for determining how to trim the elements. Since the accuracy requirements for this on-chip measurement hardware are as demanding as the requirements on the overall accuracy, most repeated trimming schemes rely on using a single piece of on-chip measurement hardware and switching in the unit elements to be calibrated one at a time (see, e.g., [29]). One advantage of periodic calibration is that the trimming signal can be stored as an analog voltage on a standard capacitor since it is refreshed each time the element is re-trimmed. All other analog trim methods require some long-term method of analog storage, such as storing charge on a CMOS floating gate [23, 26]. An advantage of all digital trimming methods, even ones that are performed only when power is first applied, is that the trimming values can be stored in digital registers, which do not require any specialized fabrication steps like those required for EPROMs.

8.3.2.3 Other Element-Matching Methods One other important method for achieving improved accuracy for the internal DAC of oversampled data converters is pulse density modulation. If the operating frequency of the oversampled feedback loop is sufficiently low, a single element can be used multiple times to achieve the effect of multiple elements. For example, when the internal DAC is part of an oversampled ADC system and the first-stage integrator is implemented using switched-capacitor circuits or when an oversampled DAC system is followed by a switched-capacitor output filter, then by dividing the period of the sampling clock into multiple phases, we can charge the capacitor multiple times and dump that charge onto the integrating capacitor in each sample period [5]. Instead of improving the element matching by trimming element values, the accuracy is improved by using the same element multiple times. Because the power required to operate switched-capacitor circuits at higher frequencies increases rapidly, this approach is typically limited to internal DACs with very few (2 or 3) bits.

8.3.3 Dynamic Element Matching

In the special case of an oversampling data converter, we can exploit the fact that the output of the data converter is followed by a filter that will remove high-frequency energy by converting the static error into a wide-band noise signal. This conversion is the basis of many dynamic element-matching algorithms. To illustrate how element mismatch can be converted into a wide-band noise signal, consider a three-element DAC. Figure 8.5(a)

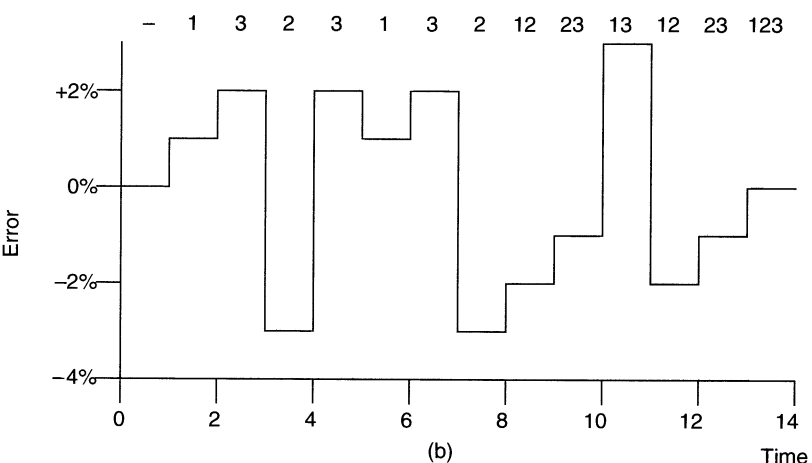
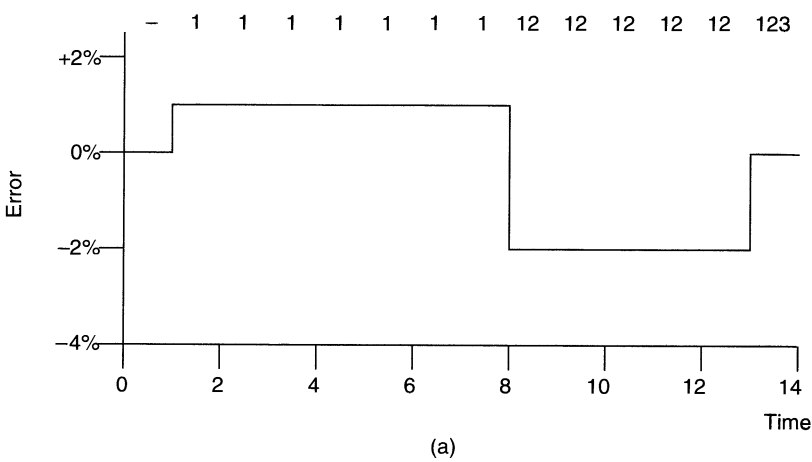


Figure 8.5 Examples of the output of a three-element DAC with mismatch between elements as a function of time: (a) unmodified parallel-unit-element architecture; (b) randomized element selection. Note that the numbers across the top indicate which elements are active during that clock period.

shows the error at the output of the three-element DAC, as a fraction of the full scale, with the following sequence of digital inputs: 0, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3. In this example, element 1 is 1% high, element 2 is 3% low, and element 3 is 2% high. Note that for any given digital input code the error remains fixed, which is the cause of the integral non-linearity in the DAC.

In general, element mismatch is converted from a dc error into a wide-bandwidth noise by choosing different elements to represent a digital input code K at different times, a technique often referred to as *dynamic element matching*. In the next four sections we will discuss four different approaches to determining how to choose different elements at different times.

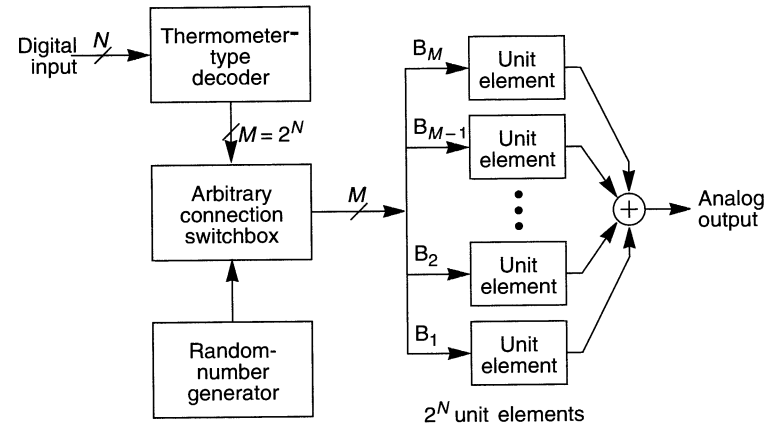


Figure 8.6 Block diagram of the parallel-unit-element internal DAC architecture with randomized element selection.

8.3.3.1 Dynamic Element Randomization Dynamic element matching can be implemented by randomly choosing different elements to represent the K th level as a function of time [5, 6]. The “randomizer” block determines which elements will be used to represent the K th level on each clock cycle (see Figure 8.6). Figure 8.5(b) illustrates the effect of randomizing the element choices on the example waveform used in Figure 8.5(a). In essence, the interconnection between the output of the thermometer decoder and the unit elements is determined at random each time period. However, each unit element ends up assigned to one and only one thermometer decoder output for that time period. The goal of this approach is to convert the error due to element mismatch from a dc offset into a time-varying signal of equivalent power that, in an oversampling converter, can be partially removed by the output filter. With ideal randomization, there will be no correlation between the mismatch error at one time and the mismatch error at any other time. Therefore, the mismatch error has been converted into a white noise. Note that although the oversampling data conversion system shapes the quantization noise, the internal DAC element mismatch noise either is not in the feedback loop (as in a DAC- see Figure 8.1) or appears added to the input (as in an ADC; see Figure 8.1 and Figure 8.2). In either case, the element mismatch noise is not affected by the feedback loop and is not shaped. However, in oversampling data conversion systems, which typically operate at oversampling ratios of 64:1 or more, nearly all of the error power is out of band and hence can be filtered out.

First, let us next consider the linearity of this DAC. For a dc input code of K , each element is active, on average, K times out of every M clock cycles, where M is the total number of elements. Therefore, each element of the DAC acts individually as a binary pulse-density modulator, and the integral linearity is limited only by the product of the fractional element mismatch ($\Delta E/E$) and the fractional clock jitter ($\Delta T/T$) [14, 28]. A second practical limit on the integral linearity results because there is often a small change in the charge (or current) transferred by each element as a function of the number of elements active. With careful choice of a DAC topology and the use of a precision clock, high dc integral linearity can be achieved, even when the elements match very poorly. For

example, no distortion components are visible in Figure 8.7 though it differs from Figure 8.4 only in that the element choices were randomized. However, as can be seen by comparing Figure 8.7(b) with Figure 8.4(b), the element mismatch now appears as an extra noise at the DAC's output. The constraint on the element matching has changed from a constraint based on the converter's linearity to a more lenient constraint based on the converter's dynamic range.

With a fixed digital input code K the output of the dynamic element randomized DAC would be a noise signal. Hence, the expected value of the variance of the noise signal is the same as the variance computed above for the output voltage error for the code K . Fortunately, a large portion of this noise power can be removed by the output filter following the oversampling data converter. If we assume that the output filter is an ideal low-pass filter with its cutoff at the Nyquist rate (half of the sampling frequency divided by the oversampling ratio R), then because the noise is white, only a fraction $1/R$ of the noise power is in the output filter's passband. The expected variance of the noise signal after the output filter can be expressed as a function of the digital input code word K as follows:

$$\sigma_{\text{in-band}}^2 \left[\frac{\Delta V_O K}{V_O M} \right] = \sigma^2 \left[\frac{\Delta V_O K}{V_O M} \right] \times \frac{1}{R} \quad (8.9)$$

As with the expected value of the error, the in-band variance of the noise signal varies in a parabolic fashion with the digital input code; going from zero at either zero or full scale to a maximum at half of full scale, $K = M/2$. At this maximum, the relationship between the in-band rms noise divided by full scale and the percentage element mismatch is

$$\sigma \left[\frac{n_{\text{in-band}}}{V_O M} \right] = \frac{1}{2\sqrt{R}\sqrt{M}} \sigma \left[\frac{\Delta E_i}{E} \right] \quad (8.10)$$

The ratio between the percentage accuracy limit set by this noise floor and the relative accuracy of the original elements is now $2\sqrt{R}\sqrt{M}$. For example, if $M = 16$ and $R = 256$, the in-band noise (relative to full scale) due to element mismatch is 128 times smaller than the relative element mismatch. A typical matching accuracy for high-quality capacitors on an integrated circuit process is $\sigma \left[\frac{\Delta E_i}{E} \right] = 0.005\%$, which would result in

$$\frac{\text{rms}\{n_{\text{in-band}}(t)\}}{V_O(M)} = \left(\frac{1}{128} \right) \sigma \left(\frac{\Delta E_i}{E} \right) \cong 3.9 \times 10^{-6} \quad (8.11)$$

For this example, the dynamic element randomizing DAC built from elements with only 0.05% matching achieved nearly perfect integral linearity, subject to the above discussion of timing jitter and element output dependence on digital input code and a noise floor nearly 108 dB below full scale. Note that in multibit oversampling systems, only a relatively small number of quantization levels is required to handle quantization errors; therefore, nearly all of the full-scale range can be used by the signal [12, 13]. Since the rms power in a full-scale sine wave is -3 dB, the SNR for this input is about 105 dB.

The randomizer connects the M outputs from the decoder to the M switching elements in a time-varying fashion. The number of possible connections is $M!$. Therefore,

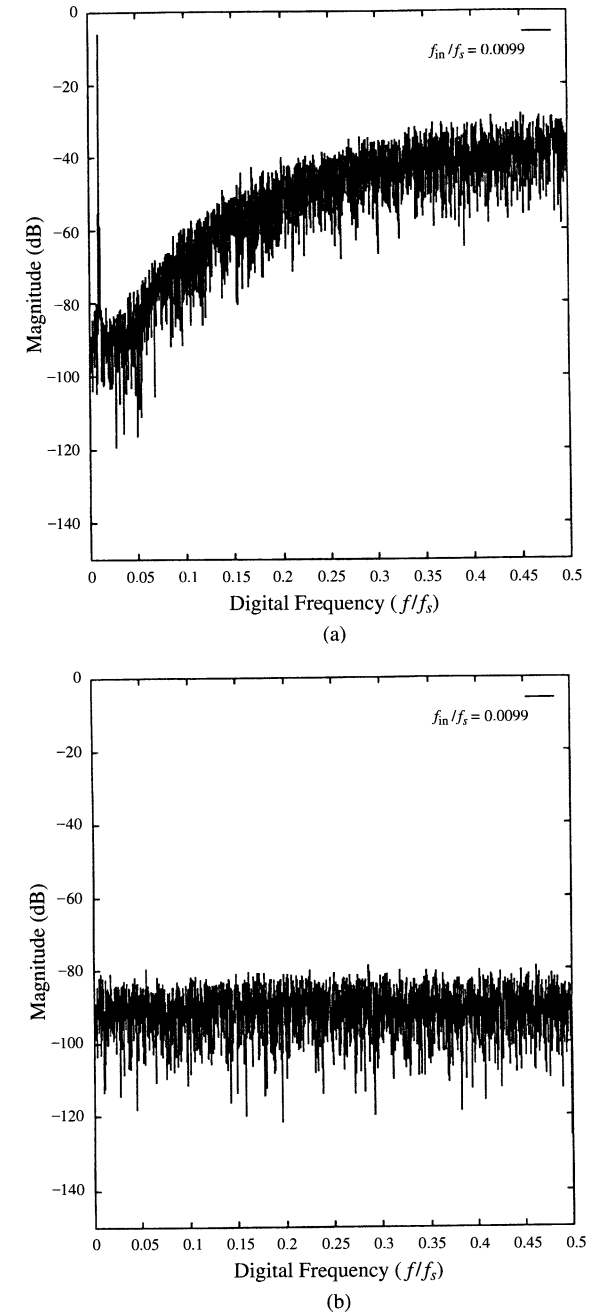


Figure 8.7 Power spectral density (a) of the signal at the output of a third-order oversampled A/D conversion system using 3-bit internal DAC with dynamic element randomization and (b) of only the internal DAC error. The input signal level is -6 dB from full scale. Elements of the internal DAC have a systematic linear gradient mismatch of 0.1% [25].

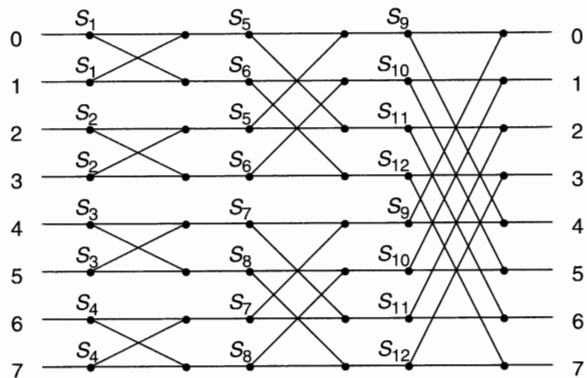


Figure 8.8 Block diagram of a butterfly structure for randomizing element selection.

when M is small (on the order 3 or 4), it is possible to randomly select between all possible connections. However, when M is large (e.g., 8 or 16), the number of possible connections is so large that it may be necessary to select a subset of connections in order to conserve die area. For example, an ideal eight-level randomizer that connects each of the eight inputs to eight outputs would have to include 40,320 possible connections.

One simple approach to randomizing over a subset of possible connections would be to have an M -port barrel shifter whose rotation is randomly changed each clock period. This represents only M of the $M!$ possible permutations. This approach would work best if the mismatch between elements were independent of the element's position on the die. Unfortunately, just the opposite is typically true. Adjacent elements are normally much more likely to match than distant elements due to gradients in the process parameters across the wafer. This results in a substantially larger noise power than that predicted by assuming that element mismatch is independent of position.

A compromise between these two extremes is the “butterfly” randomizer proposed by Kenney [5]. The butterfly randomizer circuit consists of a series of butterfly networks (such as those used in FFT architecture) coupling the inputs to the outputs (see Figure 8.8). In order that any input can be connected to any output, the number of butterfly stages should be at least equal to the number of bits in the internal DAC. More butterfly stages can be added if it is desirable to cover a larger fraction of possible connections. A pseudo-random sequence generator can be used to generate the random control sequences for the butterfly switches [29–31].

8.3.3.2 Dynamic Element Rotation–Barrel Shifter Although the dynamic element randomization approach achieved the desired goal of nominally eliminating integral nonlinearity, it did so at the cost of decreased in-band SNR. Another method, dynamic element rotation, modulates the nonlinearity error around subharmonics of the sampling clock frequency by making the mismatch noise a periodic signal [24, 28] instead of making the element mismatch noise white.

From a hardware perspective, the simplest way to arrange that all elements are used for all digital codes in a periodic manner is to rotate the connections between the thermometer decoder and the unit elements. This rotation would typically be implemented by

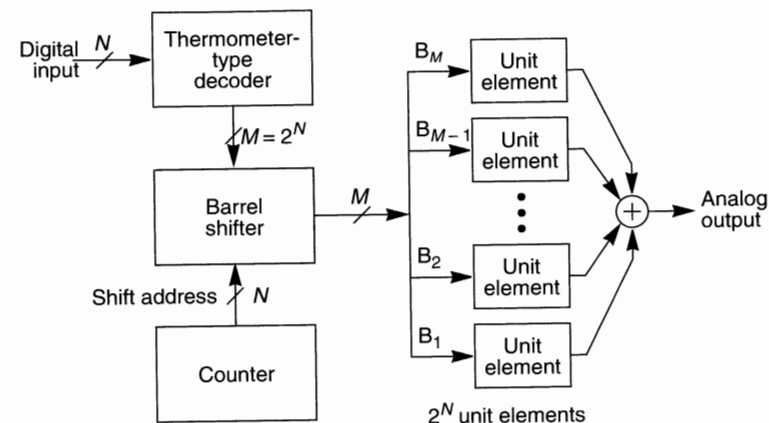


Figure 8.9 Block diagram of the parallel unit element internal DAC architecture with rotating element selection.

placing a barrel shifter between the thermometer decoder and the unit elements and turning the barrel shifter by one position on each clock pulse (see Figure 8.9). Note that the order of the wires from the outputs of the barrel shifter should also be scrambled in order to decorrelate the element mismatch and position as much as possible. One way to look at rotation is as a form of duty cycle modulation. For an output digital code of K , every element is used in K out of M clock cycles. For the example with three unit elements, when the input digital code is constant, the element mismatch is converted into a noise at a frequency of $(2\pi)/3$, which would be completely removed by the output filter following the data converter, resulting in *no integral nonlinearity* and *no additional noise* at the output of the data conversion system. In general, rotation of the unit elements will result in tones at the DAC's output at a frequency $(2\pi)/M$ and its harmonics. The amplitudes of these tones are determined by the actual pattern of the element mismatches. In order for the output filter to remove the element mismatch signal, the oversampling ratio must be greater than the number of unit elements.

The above analysis assumed that the input to the DAC was a fixed digital code. In an oversampled data converter, the input to the DAC contains both the input signal and shaped quantization noise. Mixing of the quantization noise with the element mismatch noise may result in folding additional quantization noise down into the passband of the output filter. In addition, the input signal will also mix with tones at $(2\pi m)/M$ ($m = 1, 2, \dots$), resulting in the input signal and its harmonics appearing around each of the element mismatch tones (see Figure 8.10). The limitation on the oversampling ratio must therefore be such that none of the tones fold back into the output filter's passband. This folding is most likely to happen for the tone that is nearest the passband, that is, when $m = 1$. Assuming that the maximum possible input frequency is π/R , the constraint that the j th harmonic of the input signal not fall back into the output filter passband requires that

$$\frac{\pi}{R} < \frac{2\pi}{R} - \frac{j\pi}{R} \quad (8.12)$$

Assuming that the maximum harmonic of the input signal that has sufficient power to appear above the noise floor is the J th harmonic, then the necessary constraint on the over-

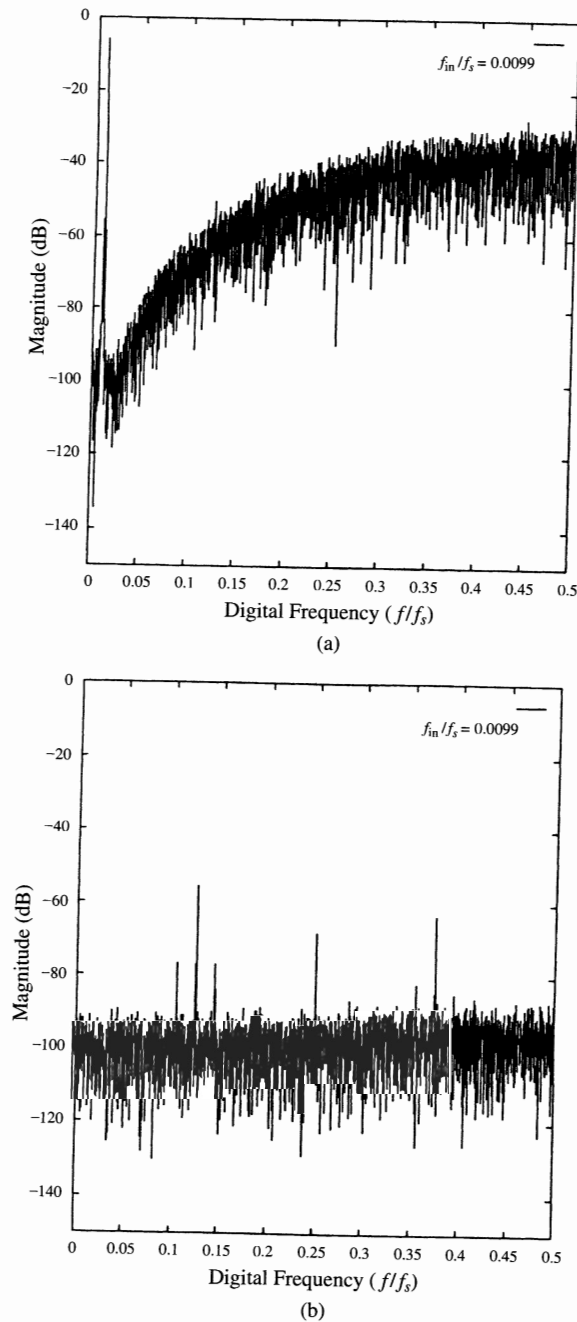


Figure 8.10 Power spectral density (a) of a signal at the output of a third-order oversampled A/D conversion system using 3-bit internal DAC with dynamic element rotation and (b) of only the internal DAC error. Input signal is -6 dB from full scale. Elements of the internal DAC have a systematic linear gradient mismatch of 0.1% [25].

sampling ratio is

$$R > (1 + J) \frac{M}{2} \quad (8.13)$$

Unfortunately, the value of J depends on the pattern of element mismatches and can typically only be determined from simulations. An upper bound on J is M , since an M -element DAC can have at most M segments in its transfer function, which can in turn generate at most M harmonics in the output. Applying this upper bound gives

$$R > \frac{1}{2}(1 + M)M \quad (8.14)$$

If the internal DAC has 3 bits, then $R > 32$, and for 4 bits $R > 128$. For internal DACs of over 4 bits, satisfying this constraint quickly becomes infeasible. However, this upper bound may be overly conservative.

When the input to the internal DAC is not constant, some noise power does appear in the passband. In addition, tones may appear in the passband as well. Tones in the output of the dynamic element rotation internal DAC may be a result of mixing between the element mismatch noise and the input signal. However, assuming that the oversampling ratio R satisfies the above constraint, all of the tones should lie outside of the passband of the output filter. Tones may also result from a mixing between the element mismatch noise and the tones in the quantization noise (limit-cycle oscillations of the oversampled converter feedback loop) [25]. In many applications, particularly digital audio, tones in the passband can be unacceptable even when their power is below the overall noise floor. The limit-cycle oscillations of oversampled data converters can be broken up and randomized by the addition of dither noise to the feedback loop [6, 14, 32–34]. In this case the quantization noise spectrum and the dither noise spectrum are both smooth and do not contain spikes (tones) that can be aliased down into the passband. However, the dither signal power at frequencies near the element mismatch tones will be aliased down into the passband and result in a decrease in the SNR. In general, this decrease in SNR is substantially smaller than the SNR decrease that results from dynamic element randomization [24].

8.3.3.3 Individual Level Averaging Another form of dynamic element matching, called *individual level averaging*, has been proposed by Leung and co-workers [25, 32]. The goal of individual level averaging is to improve the SNR in the passband compared to dynamic element randomization while avoiding the generation of tones. The tones in the passband due to quantization errors are a result of the interaction between the interpolation waveform generated by the oversampling converter feedback loop and the element mismatch waveform [25]. The fundamental idea behind individual level averaging is to guarantee that each of the elements is used with equal probability for each digital input code. Note that this is equivalent to dynamic element rotation when the input digital word is fixed. However, for individual level averaging the algorithm decides which elements are used for a specific digital code each time it occurs in such a way as to equalize across elements the number of times each one has been used to generate that specific digital code.

Leung and Sutarja [25] suggest two straightforward ways of implementing individual level averaging: “rotation” and “addition.” In both cases, a single digital register R_K of

$\log_2(M)$ bits is required for each of the M possible digital input codes. For the rotation method, the elements to be used at time i are selected by the indices $R_K(i)$ through $R_K(i) + K - 1$. If $R_K(i) + K - 1 > M$, then we wrap around and use the first elements. Then we can update $R_K(i + 1)$ to $R_K(i) + 1$ for use the next time code K is required. Note that for a fixed digital input code this is exactly the same thing that dynamic element rotation would do. However, in the case when K is varying in time, we still guarantee that eventually each code's average error is driven to zero regardless of the digital input code sequence. The number of times a digital input code must be used before all possible output values have been generated is the period of the individual averaging method for that input. For the rotation style of individual element averaging it requires M uses of any particular code to return R_K to its starting state. Note that for the rotation style of individual level averaging the length of the cycle does not depend on the code word K .

Leung and Sutarja [25] also describe a second method for implementing individual level averaging: the addition method. For the addition method, the indices of the elements to be used at time i are the same as for the rotation style: $R_K(i)$ through $R_K(i) + K - 1$, and if $R_K(i) + K - 1 > M$, then we wrap around to the first elements. However, a different update is used. For the addition style we update $R_K(i + 1)$ to $R_K(i) + K$ for use the next time code K is required. For the addition style of individual element averaging it takes only M/K uses of any particular code in order to return R_K to its starting state. When M and K are relatively prime, this will require a full M uses of code K . However, when M and K share common factors, the number of uses required before complete averaging has taken place will be smaller. This shorter period for complete averaging results in less element mismatch noise appearing at low frequencies in the passband of the output filter. We can see the difference between the rotation method and the addition method in Figure 8.11. There is nearly 10 dB of improvement in the SNR in the region of the input frequency when comparing the rotation style to the addition style of individual level averaging.

8.3.3.4 Noise-Shaped Element Usage More recently, it has been realized that it is possible to apply the noise-shaping principle to the errors caused by element mismatch [40, 41]. By modulating the element control signals in a manner analogous to $\Delta\Sigma$ modulation, the element mismatch errors can be endowed with a noise-shaped spectrum [42].

A block diagram of the element selection logic that implements noise-shaped element usage with a NTF of H_2 is shown in Figure 8.12. The input to the logic is the digital code $v(n)$, which takes on values from zero to M ; the number of levels in the DAC is $M + 1$. The output of the system is the selection vector, $sv(n)$, a collection of bits that enable individual elements in the unit element array. The element selection logic itself is essentially a collection of M digital $\Delta\Sigma$ modulators, each possessing a NTF equal to H_2 , implemented with the error feedback structure and supplied with a common input. The chief difference between the modulators in the element selection logic and a set of regular modulators is that the quantizers (which are binary) are required to produce a combined total of $v(n)$ 1s at time n . As a result of this requirement, the vector quantizer provides coupling between the modulators.

The input to the vector quantizer is a time-varying collection of M digital numbers, $sy(n)$, which represent the “desired usage” of each of the M unit elements. If it were

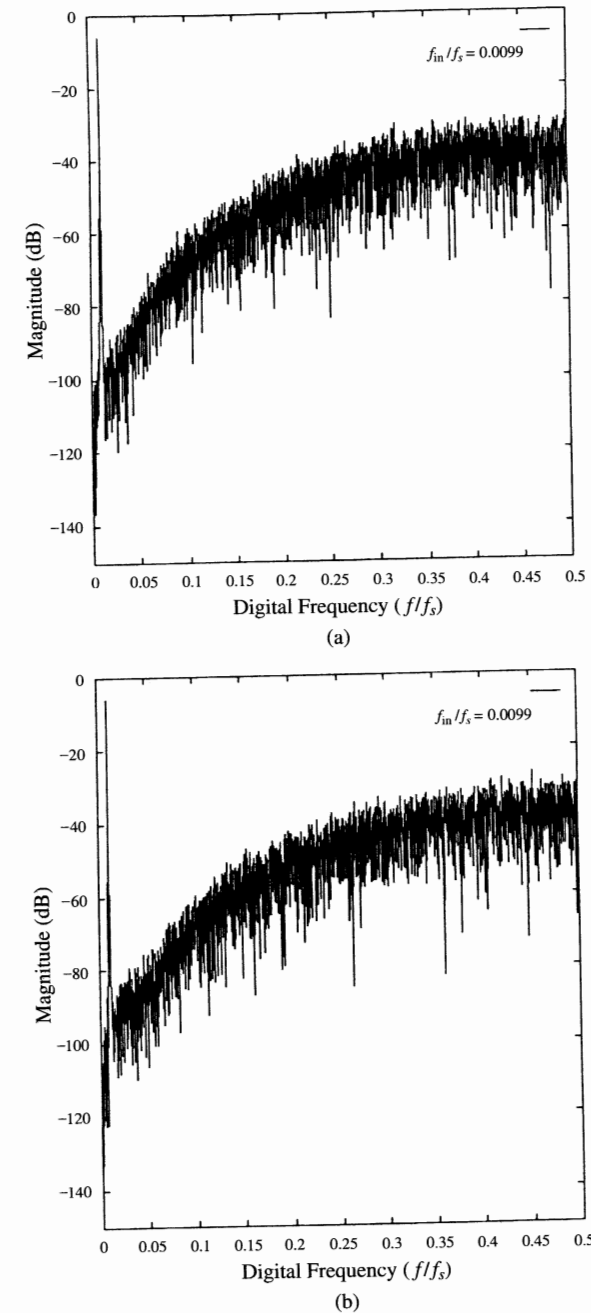


Figure 8.11 (a) Power spectral density of signal at output of a third-order oversampled A/D conversion system using 3-bit internal DAC with individual level averaging using rotation. (b) Power spectral density of signal at output of a third-order oversampled A/D conversion system using 3-bit internal DAC with individual level averaging using addition (b). The input signal is -6 dB from full scale. Elements of the internal DAC have a systematic linear gradient mismatch of 0.1% [25].

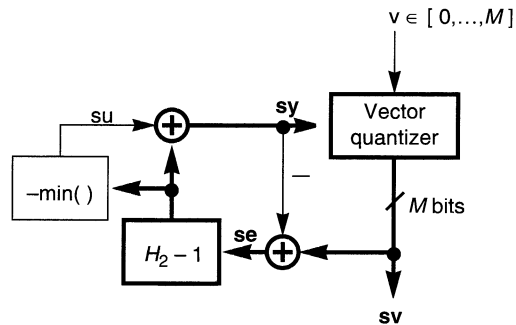


Figure 8.12 General block diagram of element selection logic which results in noise-shaped element mismatch. Vector-valued signals are bold.

possible to give each element in the array a weight equal to the corresponding component of $sy(n)$, error-free conversion would result. However, each element may only be given a weight of 0 or 1 and the sum of the weights must equal $v(n)$. The vector quantizer uses the information in the $sy(n)$ vector to select which $v(n)$ elements to enable. Selecting those elements with the largest $sy(n)$ components results in the least “selection error,” $se(n) = sv(n) - sy(n)$. The $se(n)$ vector is fed back to the quantizer input after undergoing filtering by the $H_2 - 1$ filter and a shifting operation which sets the minimum component in $sy(n)$ to zero. The purpose of the shifting operation is simply to reduce the magnitude of the $sy(n)$ vector, in a manner that does not disturb the noise-shaping property of the selection logic. In an actual implementation, this normalization step could be implemented in the vector quantizer itself.

To see how this system results in noise-shaped DAC errors, define the element errors as the difference between the actual element value and the average of all elements and assemble these into a (static) M -element column vector de . As a result of this definition, the sum of the components of de is precisely zero:

$$[1 \cdots 1] \cdot de = 0 \quad (8.15)$$

By analogy with the error-feedback topology, the z -transform of the output of the element selection logic, expressed as an M -element vector of scalar z -transforms, is

$$SV(z) = SU(z)[1 \cdots 1] + H_2(z)SE(z) \quad (8.16)$$

Assuming without loss of generality that the average element value is 1, the output of the DAC is

$$DV(z) = SV(z) \cdot ([1 \cdots 1] + de) \quad (8.17)$$

The constraint on the vector quantizer results in

$$SV(z) \cdot [1 \cdots 1] = V(z) \quad (8.18)$$

Thus, by (8.15), (8.16) and (8.18), (8.17) becomes

$$DV(z) = V(z) + H_2(z)(SE(z) \cdot de) \quad (8.19)$$

This equation shows that the DAC output is composed of the digital input v plus a noise term due to element mismatch that is shaped by H_2 . As generalizations of this system, one might consider adding dither to the sy vector to whiten the noise caused by a deterministic selection algorithm or to use a multibit quantizer and selection vector. To maintain the invariants of the DAC errors and hence the noise-shaping property of the system, the implementation of a “multibit unit element” would necessitate the repeated use of an individual element in each clock period, as discussed in Section 8.3.2.3.

An open question regarding the element selection logic is its stability. It is easy to show that $H_2(z) = 1 - z^{-1}$ results in a system that is stable. This choice for H_2 also results in a simple pattern of element usage: the elements are chosen in a circular fashion, starting from the element adjacent to that which was most recently used [41]. Consequently, an implementation of first-order shaping of the element mismatch noise is trivial, requiring only one register of length $\log_2 M$ bits and some combinational logic. Simulations indicate that other choices for H_2 also yield stable selection logic, including second-order shaping [$H_2(z) = (1 - z^{-1})^2$], bandpass noise shaping [$H_2(z) = (1 - z^{-1} + z^{-2})$], as well as more general NTFs subject to a constraint on their peak out-of-band gain. However, for such NTFs, the realization of the selection logic requires many more bits of storage (about 4 bits per element for second-order shaping) and a great deal more combinational logic.

Figure 8.13 shows the output spectrum of a third-order converter for a half-scale sine-wave input when connected to a 16-element DAC with 1% element mismatch and various orders of mismatch noise shaping. With an ideal DAC, the SNR at an oversampling ratio of 50 is 118 dB. With second-order mismatch noise shaping, the SNR is degraded by less than 7 dB if the element mismatch is less than 1%. With first-order mismatch shaping, mismatch errors less than 0.2% are needed for comparable performance while unshaped mismatch noise would require the mismatch to be less than 0.002%.

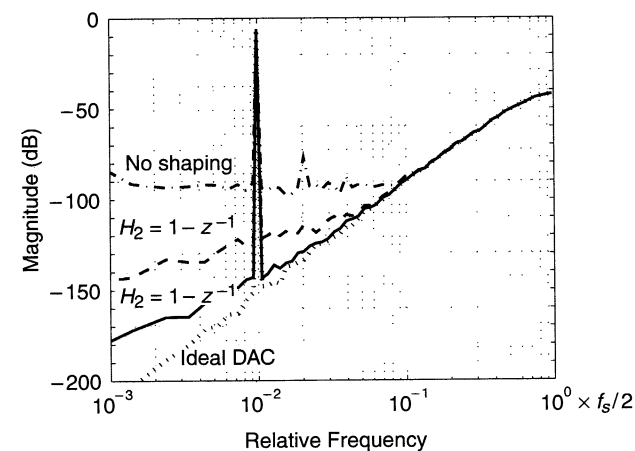


Figure 8.13 Output spectra for a third-order low-pass modulator feeding a 16-element DAC with 1% element mismatch.

Clearly, noise-shaping element mismatch can result in a very high performance even with moderately well matched components.

8.4 DIGITAL CORRECTION TECHNIQUES

The preceding sections of this chapter described techniques that can convert the effects of DAC nonlinearities into a time-varying pseudorandom noise and possibly also shift this noise to some higher out-of-band frequencies. Most of the noise introduced by the nonlinearities could then be removed by subsequent digital filtering. A totally different strategy, based on converting the noise due to DAC error into a digital form and then canceling it in the digital domain, was proposed in [43, 44]. The details of this approach will be discussed in this section.

8.4.1 $\Delta\Sigma$ ADC Architectures with Error-Storing Random-Access Memory

The basic concept for a digitally corrected ADC containing a multibit internal quantizer is illustrated in Figure 8.14. The correction is carried out in the digital block following the $\Delta\Sigma$ loop. The input to the block is the N -bit output of the loop; its output is a data stream with an initially much higher resolution, representing the corrected data. In its conceptually simplest implementation, the correction block is simply a random-access memory (RAM). In each clock period, the input word to the RAM selects an M -bit word ($M \gg N$) for the RAM output. The sequence of these words is the corrected output data stream.

It is easy to see that for correct operation the data stored in the RAM in this implementation should be simply the accurate digital equivalents of actual output levels of the N -bit DAC. To see this, assume that the RAM does contain these values, each stored at an address that is given by the corresponding DAC input code. Assume also that $N = 3$ and that the word $v(n) = 101$ appears at the output of the loop. As a result, a dc analog voltage $v'(n)$ [ideally equal to $(5/8)V_{\text{ref}}$ but in reality slightly different due to the inevitable imperfections of the DAC circuit] appears at the output of the N -bit DAC. Simultaneously, an M -bit digital word $w(n)$ that is a very accurate representation of this (imprecise) DAC output voltage is retrieved from the RAM and is fed into the decimation filter. In a typical case, this word will be very nearly equal to the RAM input code $v(n)$ (here, 101); thus, for the case of a 10-bit accurate DAC and a required linearity of 16 bits, it may be of the form $w(n) = 1010000001100101$. In the baseband, where the loop gain provided by the filter $L(z)$ is very high, the spectrum of the actual DAC output signal $v'(n)$ follows that of the input $u(n)$ very closely and with excellent linearity. Since at the same time $w(n)$ is, by

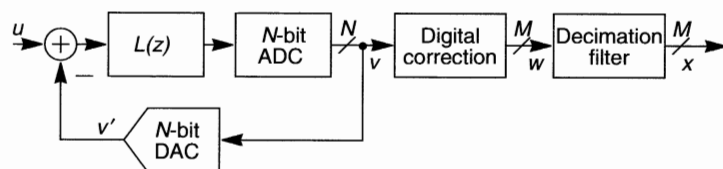


Figure 8.14 General scheme of a digitally corrected $\Delta\Sigma$ ADC.

assumption, an accurate digital replica of the DAC output, its baseband spectrum must also correspond very accurately to the input spectrum.

Figure 8.15 (from [44]) verifies these statements. Part (a) shows the measured output spectrum of a $\Delta\Sigma$ ADC with a 3-bit precision internal ADC and DAC. Part (b) illustrates the output spectrum when the internal ADC and DAC had a large ($\frac{1}{2}$ LSB) nonlinearity; finally, part (c) shows the spectrum for the corrected converter. It is very similar to the ideal spectrum.

An alternative way of incorporating digital correction into the $\Delta\Sigma$ ADC with a multibit quantizer is illustrated in Figure 8.16 [43]. Here, the RAM is inside the $\Delta\Sigma$ loop, and the correction is accomplished by cascading a digital $\Delta\Sigma$ loop containing the RAM with the multibit DAC. An argument similar to the one given for the system of Figure 8.14 proves that *if* the in-band loop gains of both analog and digital $\Delta\Sigma$ loops are sufficiently high, *if* the overall system remains stable, and *if* the RAM contains the accurate digital equivalents of the output levels of the multibit RAM, then an accurate correction of the DAC nonlinearity errors will be achieved.

The systems of Figures 8.14 and 8.16 require that an accurate digital equivalent of each output level of the DAC be acquired and stored in the RAM. The resolution and accuracy of these data must be at least as high as the overall resolution of the $\Delta\Sigma$ ADC system. Also, as shown in the figures, the decimation filter in the corrected system must be able to process multibit input data at a fast clock rate. This speed requirement makes its realization expensive. Finally, for proper operation, the settling behavior of the DAC must be the same during conversion as it was during the error acquisition (calibration) stage. This can be achieved if the circuit can fully settle in every clock period, both during the calibration as well as during the conversion operation.

In the next section, the calibration process will be discussed. Then, in the following section, some modifications of the correction system, which reduce the complexity of the correction hardware and ease the burden on the decimation filter, will be described.

8.4.2 The Calibration of the Digitally Corrected $\Delta\Sigma$ ADC

As discussed above, the accurate acquisition of the required DAC data for the RAM is necessary for the proper functioning of the correction process. Fortunately, this can easily be performed using on-chip components [43]. The process is illustrated in Figure 8.17. An N -bit digital counter produces successively all the possible 2^N input codes for the DAC; the DAC output level V_c for each code is held for a duration of at least 2^M clock periods, where M is the required linearity, expressed in bits. During this time, V_c is converted into a 1-bit data stream using the original $\Delta\Sigma$ ADC reconfigured into a single-bit one. [This is achieved by simply using only the most significant bit (MSB) in the internal quantizer.] A digital filter (usually simply a counter, borrowed from the existing decimation filter system) finds the digital equivalent of V_c as the mean value of the data stream. For the required M -bit accuracy, if a counter is used as the averaging filter, this process needs at least 2^M clock periods, plus the settling time of the overall system. After the M -bit word representing the converted value of V_c has been found, it is stored in the RAM at the address defined by the N -bit input code generated by the input counter. Next, the process is repeated for the following digital input code until all 2^N DAC output levels have been converted and stored in the RAM.

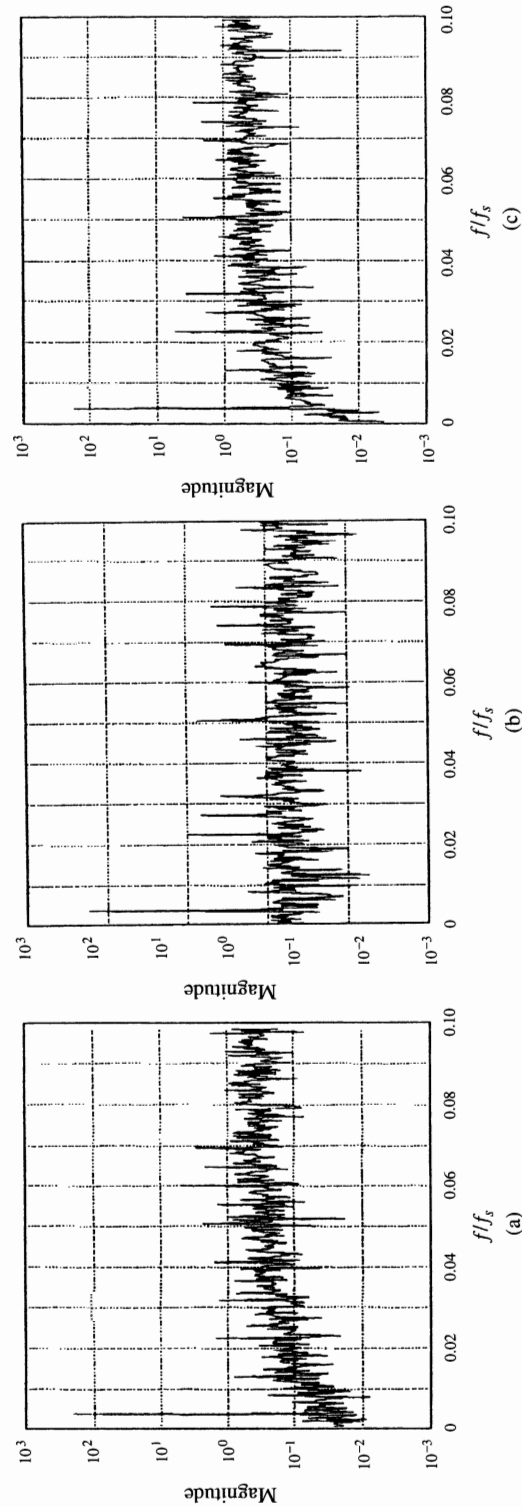


Figure 8.15 (a) Measured spectrum of the linear 3-bit $\Delta\Sigma$ ADC. (b) Measured spectrum of the uncorrected nonlinear 3-bit $\Delta\Sigma$ ADC. (c) Measured spectrum of the corrected nonlinear 3-bit $\Delta\Sigma$ ADC.

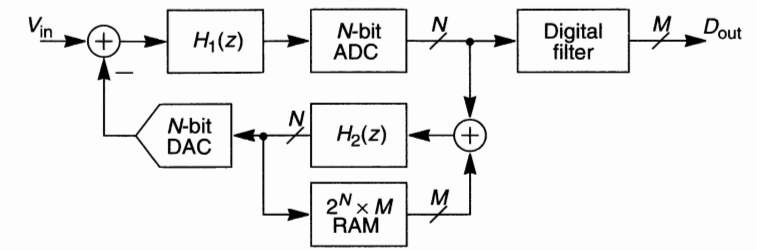


Figure 8.16 Digitally corrected N -bit $\Delta\Sigma$ ADC.

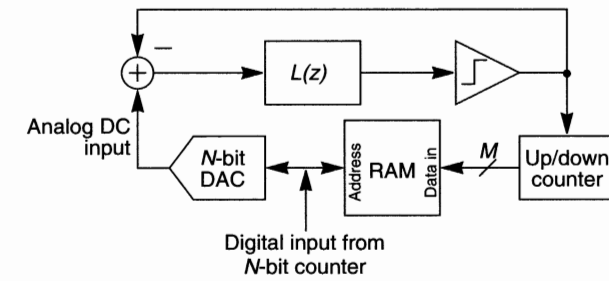


Figure 8.17 Calibration scheme for digital correction.

The total memory capacity of the RAM needs to be $M \times 2^N$, and the total calibration process requires at least 2^{M+N} clock periods. The calibration can be performed at power-up time only. Alternatively, the analog front end may be duplicated on the chip, and the two front-end stages then take turns converting and being calibrated. This way, thermal and drift effects can also be continuously corrected.

8.4.3 An Improved Digital Correction System

The digitally corrected multibit system described above can be made more practical and economical by taking advantage of the special form of the calibration data stored in the RAM. As explained above, the multibit binary number stored at the address (say) 101 is very close (usually with an error only in the 10th and/or lower binary positions) to its ideal value 101. Hence, a large saving in the digital hardware can be obtained by processing the bits corresponding to the ideal values separately from the error bits. The general principle is illustrated in Figure 8.18a; an implementation of the correction circuitry for the specific case of a second-order 16-bit $\Delta\Sigma$ ADC with a 4-bit internal quantizer [45] is illustrated in Figure 8.18b. Here, as before, the 4-bit output of the analog front end is used as the address code for the correction RAM. The data in the RAM, however, now represents the *errors* of the 4-bit DAC levels, rather than their actual values. The number of bits that must be stored can thus be reduced. In the chip described in [45], a worst-case linearity of 9 bits was assumed for the DAC, and an 18-bit accurate correction was performed. Thus, the original system would have required the storing and subsequent processing of 18-bit data. By contrast, in the error-processing system of Figure 8.18a it was sufficient to

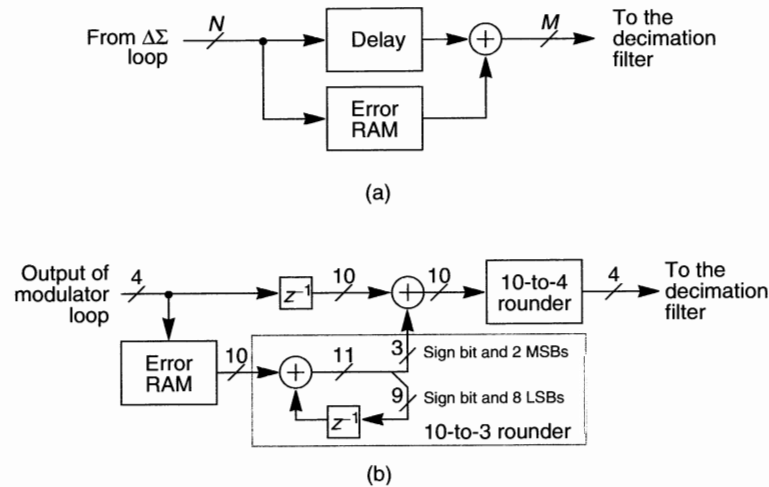


Figure 8.18 (a) Separation of the ideal and error codes in a digitally corrected $\Delta\Sigma$ ADC. (b) An example of the digital correction system.

store 9-bit (plus sign bit) data. More importantly, it was also possible to reduce the word length of these error data from 10 bits to 3 bits, by including a simple first-order digital $\Delta\Sigma$ loop at the output port of the RAM. The resulting compressed error words were then added to the uncorrected 4-bit output data of the analog loop, in order to obtain the corrected output data stream. Finally, to make the subsequent decimation filtering more economical, the word length of the corrected data was also reduced to 4 bits in an additional second-order digital $\Delta\Sigma$ stage (Figure 8.18b).

The calibration process for the modified system is illustrated in Figure 8.19. It differs slightly from the one described earlier (Figure 8.18a) in that only the difference between

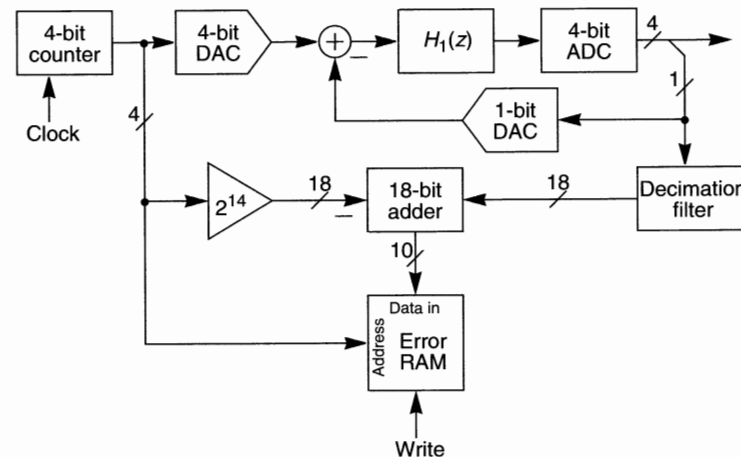


Figure 8.19 Calibration scheme for an error-storing digital correction system.

the 4-bit address code and the averaging (decimation) filter output needs to be stored, requiring only a $9 + 1 = 10$ -bit word length. The 18-bit adder used to perform the subtraction does not need to operate at a fast speed, since it has to subtract data only once in every 2^{18} clock periods.

In addition to the numerous advantages of using a multibit internal quantizer, as discussed in Section 7.1, another major advantage can be discerned in the system described above. The slew rate of the op-amp in the input stage of the analog front end is a crucial design parameter in every $\Delta\Sigma$ ADC, since the op-amp input signal varies rapidly due to the (normally very large) DAC output steps. If the time needed for slewing becomes comparable to the time available for the settling of the input stage, then nonlinear distortion appears in the converted signal. Thus, the op-amp has to be designed with sufficient slew rate to follow the largest possible input step rapidly. Since the step size of the DAC output is cut in half with every bit added to the word length of the quantizer, the necessary slew rate of the input op-amp for $\Delta\Sigma$ ADC with, say, a 4-bit quantizer is only about one-eighth of that needed in an ADC with single-bit quantization.

Figure 8.20 (from [45]) compares the measured input and output spectra of the correction stage for a digitally corrected $\Delta\Sigma$ ADC for a sine-wave analog input signal. Clearly, the correction reduces the noise floor by nearly 20 dB and completely eliminates the large harmonic content caused by the DAC nonlinearities. The linearity of the DAC used was about 10.5 bits.

8.4.4 Cascade $\Delta\Sigma$ ADC Systems Using Digital Correction

The concept of digital correction is also applicable to cascade (MASH) $\Delta\Sigma$ structures. Figure 8.21 shows a two-stage cascade ADC [46] with a correction RAM in the first stage and with additional scaling by a factor $A > 1$ in the second stage. This scale factor is useful since it allows the subsequent reduction of the quantization error in the second stage by a $1/A$ factor; it is made possible by the fact that the input signal to the second stage is now significantly (by about 2^{N_1} times) smaller than the input to the first one. Hence, a gain block with a scale factor A close to 2^{N_1} can be used between the first and the second stages, and a block with a scale factor $1/A < 1$ can follow the second stage. The second block will reduce the quantization noise and nonlinearity error introduced by the second loop. It follows that under ideal conditions the quantization noise at the output of this system is equivalent to the noise output of a single-stage $\Delta\Sigma$ ADC, with a loop filter transfer function $[L(z)]^2$ and with a linear $(N_1 + N_2)$ -bit ADC and DAC. Note that the second stage of this system normally does not require any digital correction, since it processes only the quantization error of the first stage and not the input signal, and hence the harmonic distortion caused by the N_2 -bit DAC is not a significant consideration. Also, the noise due to the nonlinearity error in the N_2 -bit DAC is filtered by the high-pass filter $H_D(z)$ and is hence suppressed in the baseband.

If the full accuracy of the system described above is not needed, the second stage of the cascade ADC of Figure 8.21 can be simplified. As shown in Figure 8.22, the second $\Delta\Sigma$ loop can be simply replaced by an internal N_2 -bit ADC. The system's output quantization noise is now the same as that of a single-stage ADC with a loop filter transfer function $L(z)$ and with an $(N_1 + N_2)$ -bit internal quantizer. The N_2 -bit ADC need not be a parallel ("flash") converter; it may be pipelined and its latency absorbed by a matching delay (shift register) cascaded with the digital correction circuit.

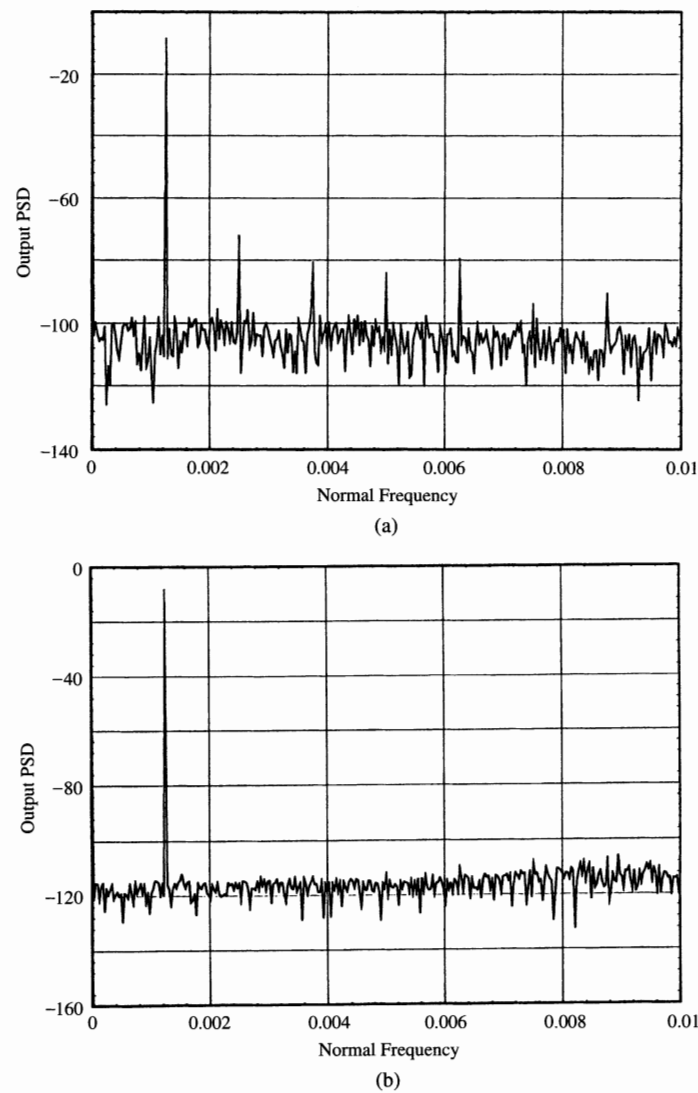


Figure 8.20 Measured spectrum of (a) the uncorrected modulator output and (b) the corrected modulator output.

8.4.5 Digitally Corrected $\Delta\Sigma$ ADC with Companding Quantizer

The characteristics of the internal multibit ADC and DAC need not be linear. By assigning an exponential input–output characteristic to the internal ADC and a logarithmic one to the internal DAC, it is possible to realize a companding $\Delta\Sigma$ ADC. Such a converter exhibits an SNR that is higher for smaller inputs, but lower for large inputs, than that of a $\Delta\Sigma$ ADC with a linear ADC output quantizer [47]. Figure 8.23 shows the quantizer charac-

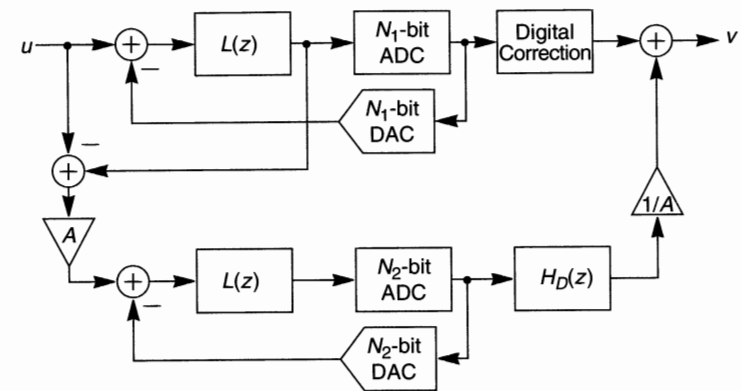


Figure 8.21 Two-stage digitally corrected MASH ADC.

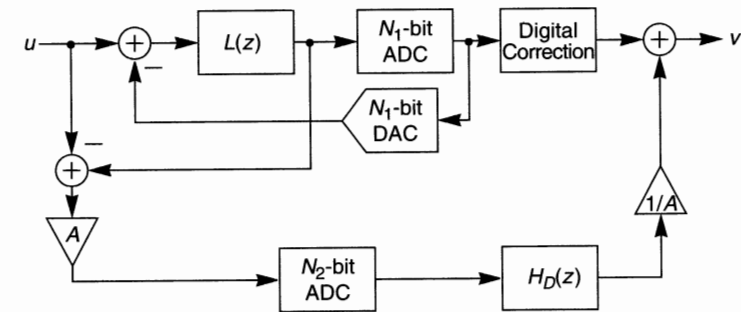


Figure 8.22 Simplified digitally corrected MASH ADC.

teristics, and Figure 8.24 the output signal of such an ADC with a sine-wave input, when the μ -law companding characteristics [48] are used (with $\mu = 100$) for determining the quantization steps. Clearly, the quantization noise is very small when the input is small, but it increases quite rapidly for larger input signals. Thus, the SNR is nearly independent of the input signal amplitude over a broad amplitude range. This statement is verified in Figure 8.25, which shows the simulated SNR versus input amplitude characteristics for a first- and a second-order $\Delta\Sigma$ ADC with linear as well as μ -law quantizers. For small amplitudes, the SNR of the ADCs with nonlinear quantizers is much larger than that of the linear-quantizer ones. Only for very large amplitudes is the linear quantizer preferable. Such a performance may be useful, for example, in digital audio systems where the tolerable noise level is usually lower for smaller input signals.

Since the overall input–output relation between the analog input signal and the final digital output must be linear, the digital correction circuitry in this system must perform, not only the DAC error correction, but also the added task of compensating for the companding relation used. Both of these goals are easily achieved, however, if the correction RAM that is cascaded with the $\Delta\Sigma$ loop accurately reproduces in digital form all levels of the multibit nonlinear internal DAC. To achieve a wide dynamic range and a low-complexity realization, a binary floating-point representation should be used for the stored data.

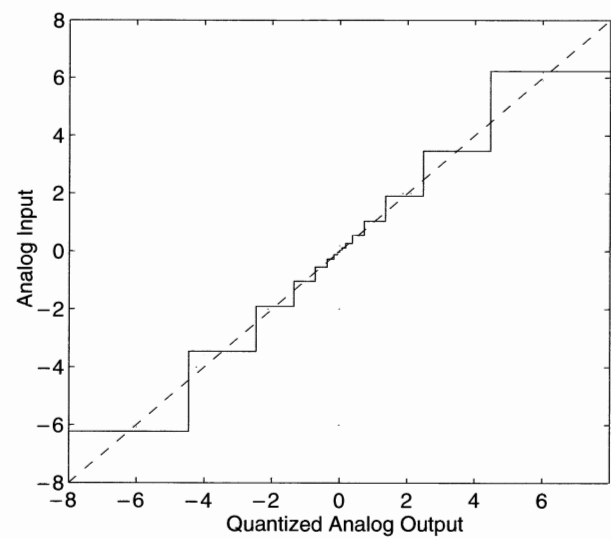


Figure 8.23 Quantizer characteristics for a companding ADC (μ -law with $\mu = 100$).

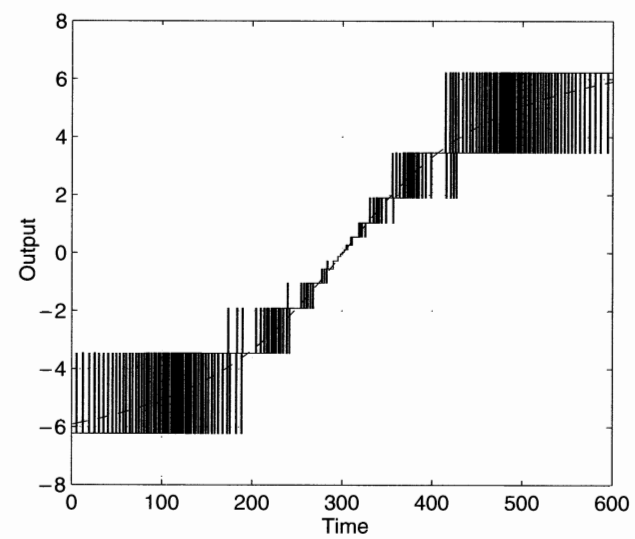


Figure 8.24 Output waveform of a nonuniformly quantized first-order $\Delta\Sigma$ ADC with μ -law ($\mu = 100$).

Finally, it should be pointed out that the correction of the internal DAC's nonlinearity can also be carried out in the analog domain by adding a calibration network to each unit capacitor within the DAC [49] or adding a small calibration DAC in parallel with the internal DAC [50]. Also, it is possible to combine digital correction and randomization of the nonlinearity noise [49].

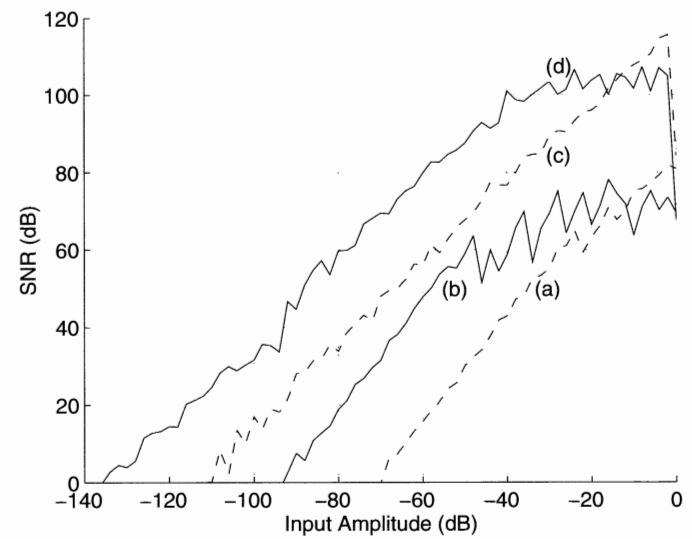


Figure 8.25 Simulated SNR vs. input signal amplitude (relative to full scale) for four 16-level $\Delta\Sigma$ ADCs: (a) first-order modulator with uniform quantization; (b) first-order modulator with μ -law quantization; (c) second-order modulator with uniform quantization; (d) second-order modulator with μ -law quantization. All μ -law quantizers use $\mu = 100$.

8.5 DUAL-QUANTIZER ADC ARCHITECTURES

An alternative approach to the design of a $\Delta\Sigma$ ADC with a multibit internal quantizer is to use *two* quantizers. One is a single-bit circuit contained in a $\Delta\Sigma$ loop, which includes a single-bit DAC in the feedback path between the output and the input of the modulator. Since this DAC plays the key role in determining the linearity of the modulator, its inherent linearity is used to full advantage. An added path with a second quantizer, a multibit one, is used to convert and cancel the large quantization error generated by the single-bit quantizer. This cancellation will then reduce the overall quantization error of the modulator to that of one with a linear multibit internal quantizer. Several schemes have been developed based on this principle. They will be discussed next.

8.5.1 The Leslie–Singh Architecture

The basic scheme of the dual-quantizer architecture proposed by Leslie and Singh [51] is shown in Figure 8.26a. As the diagram shows, a multibit (N -bit) ADC is used in the forward path, but only the MSB is fed back to the single-bit internal DAC. An equivalent circuit, which shows the quantizer separated into a 1-bit and an N -bit ADC, is shown in Figure 8.26b. It can be seen that the upper path in the system converts into digital form the input signal plus an added noise consisting of the high-pass-filtered quantization error E_1 of the 1-bit ADC. The lower path converts the analog input signal of the 1-bit quantizer,

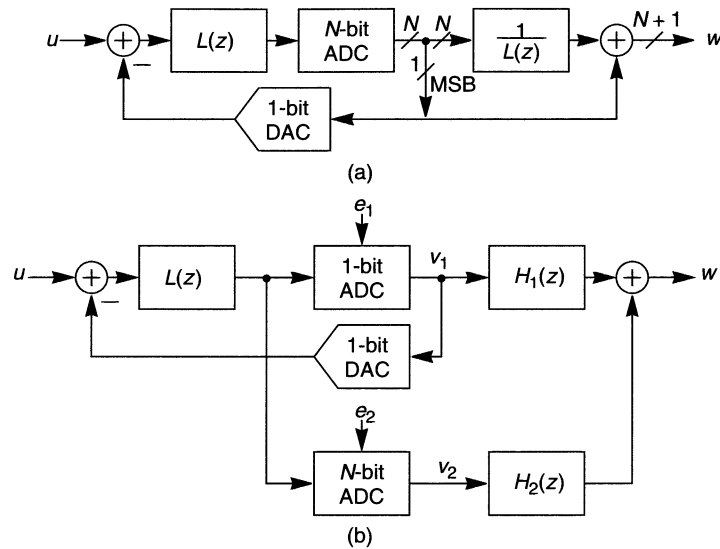


Figure 8.26 (a) Leslie-Singh structure. (b) An equivalent representation.

which also contains the signal plus a differently filtered version of E_1 . For appropriate choice of the internal blocks $L(z)$, $H_1(z)$, and $H_2(z)$, the error E_1 can be canceled in the output signal W . Clearly, the operation is analogous to that of the circuit of Figure 8.22 and of the cascade ADC system discussed in Chapter 6, with the lower path playing the role of the second stage of the cascade architecture.

Analysis of the system of Figure 8.26b gives for the output signal the expression

$$W = H_1 V_1 + H_2 V_2 = G(H_1 + H_2)U + (H_1 H + H_2 H - H_2)E_1 + H_2 E_2 \quad (8.20)$$

Here, G and H are the signal and noise transfer functions, respectively, of the 1-bit loop; also, E_1 and E_2 are the quantization errors of the 1-bit and N -bit ADCs, respectively, in the z -domain. As the equation shows, the large error E_1 can be canceled by satisfying the condition

$$\frac{H_1}{H_2} = \frac{1}{H} - 1 \quad (8.21)$$

It is usually advantageous to choose the upper path signal transfer function G and the overall signal transfer function $G(H_1 + H_2)$ both as delays of k clock periods:

$$G = G(H_1 + H_2) = z^{-k} \quad (8.22)$$

Then the design equations become

$$H_1 = 1 - H \quad (8.23)$$

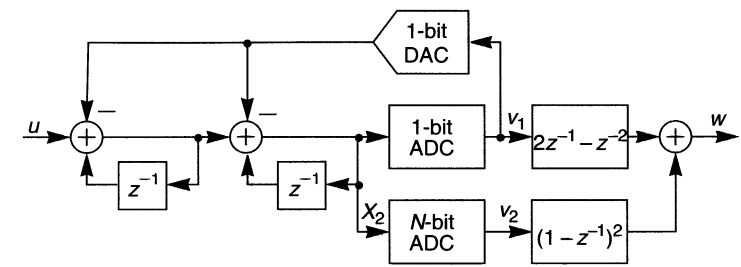


Figure 8.27 Second-order dual-quantizer modulator.

$$H_2 = H \quad (8.24)$$

$$W = z^{-k}U + HE_2 \quad (8.25)$$

As the last relation shows, the output noise now consists of the smaller (N -bit) quantization error E_2 , filtered by the noise transfer function of the $L(z)$ upper path loop.

For the important case of a second-order $\Delta\Sigma$ ADC, the transfer functions $G = z^{-1}$ and $H = (1 - z^{-1})^2$ can be used. Then, Eqs. (8.23) and (8.24) give $H_1 = 2z^{-1} - z^{-2}$ and $H_2 = (1 - z^{-1})^2$. A system realizing these transfer functions is shown in Figure 8.27, where it is assumed that each ADC contains an internal delay of one clock period.

In practice, condition (8.21) cannot be exactly satisfied. The transfer functions H_1 and H_2 can be realized accurately by digital blocks, but the exact form of the noise transfer function H depends on the analog components of the 1-bit loop. As (15) shows, if H is inaccurate by an amount dH , an added noise

$$dE = (H_1 + H_2)dH \cdot E_1 = dH \cdot E_1 \quad (8.26)$$

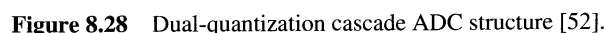
leaks to the output. Thus, the sensitivity of the overall circuit to noise leakage is the same as that of the upper path. A likely cause of such leakage is the finite gain of the input op-amp. This will cause an added noise that is (to a good approximation) equal to E_1 filtered by a high-pass filter of order M , where M is the order of the loop filter in the 1-bit loop.

The noise leakage due to analog inaccuracies usually limits the achievable accuracy if the resolution of the multibit ADC is high. This generally makes it useless to choose the value of N larger than 3–5 bits.

8.5.2 Dual-Quantization Cascade ADC Architectures

A useful dual-quantization system can be obtained by using a multibit quantizer in the second stage of a two-stage cascade ADC. The resulting structure is shown (for a third-order 2–1 modulator) in Figure 8.28. Detailed analysis [52] shows that the output signals of the upper and lower $\Delta\Sigma$ loops are given by

$$V_1 = z^{-1} \cdot U + (1 - z^{-1})^2 \cdot E_1 \quad (8.27)$$


$$V_2 = z^{-1}(E_1 - E_D) + (1 - z^{-1}) \cdot E_2 \quad (8.28)$$
$$W = H_1 V_1 + H_2 V_2 = z^{-2} U + z^{-1} (1 - z^{-1})^2 E_D - (1 - z^{-1})^3 E_2 \quad (8.29)$$
$$dH = (2/A)(1 - z^{-1}) + z^{-1}/A^2 \quad (8.30)$$

8.5.3 Dual-Feedback Single-Path ADC Architecture

[illegible]

ensuring the linearity of the signal transmission to the overall output W . By feeding the analog replica of the N -bit ADC output V_2 into the last integrator, it becomes possible to cancel the large 1-bit quantization error E_1 in W .

$$W = z^{-1}U + 2(1 - z^{-1})^3 E_2 - 2z^{-1}(1 - z^{-1})^2 E_D \quad (8.31)$$

As was the case for the circuit of Figure 8.28, under finite op-amp gain conditions a leakage of uncanceled 1-bit quantization noise will occur. Again, this noise will be subject to a first-order high-pass filtering and thus reduced in the baseband. Figure 8.30 shows the measured signal/(noise+distortion) performance of a recently published ADC based on this principle [54].

There are many advantages to employing a multibit, rather than single-bit, internal quantizer in an oversampled data conversion system. There is a significant increase in the dynamic range due to the increased resolution (and hence reduced quantization noise) of the internal quantizer. In fact, it can be shown [52] that the dynamic range increase in decibels equals $20 \log(2^N - 1)$, or nearly $6N$ decibels, when an N -bit quantizer replaces the single-bit one. Thus, using a 4-bit quantizer, close to 24 dB can be gained. In addition, the stability of high-order loops is strongly enhanced when multibit quantizers are used. Also, the design of such loops becomes easier, since the assumption that the quantization noise has white-noise behavior is much more accurate, and it is also simpler to assign an equivalent gain to the quantizer.

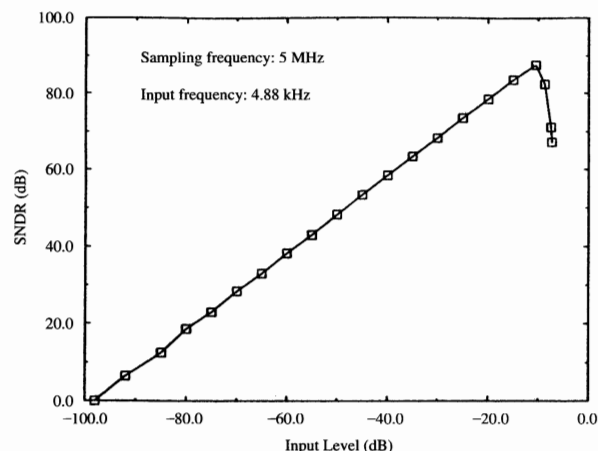


Figure 8.30 Measured signal/(noise+distortion) vs. input level characteristic of an ADC based on the scheme of Figure 8.29.

In spite of these important advantages, until quite recently little commercial use has been made of multibit quantizers in oversampling data converters, because of the difficulty of achieving high integral linearity. In this chapter, we presented a number of approaches that can achieve very high overall linearity for the converter and require only a modest accuracy of the analog components used in the system.

REFERENCES

- [1] R. W. Adams, "Design and implementation of an audio 18-bit analog-to-digital converter using oversampling techniques," *J. Audio Eng. Soc.*, vol. 34, no. 3, pp. 153–166, March 1986.
- [2] P. J. A. Naus, E. C. Dijkmans, E. F. Stikvoort, A. J. McKnight, D. J. Holland, and W. Brandinal, "A CMOS stereo 16-bit D/A converter for digital audio," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 3, pp. 390–395, June 1987.
- [3] Y. Matsuya, K. Uchimura, A. Iwata, T. Kobayashi, M. Ishikawa, and T. Yoshitome, "A 16-bit oversampling A-to-D conversion technology using triple-integration noise shaping," *IEEE J. of Solid-State Circuits*, vol. SC-22, no. 6, pp. 921–929, Dec. 1987.
- [4] M. W. Hauser, "MOS ADC-filter combination that does not require precision analog components," *ISSCC Dig. Tech. Papers*, pp. 80–81, New York, NY, 1985.
- [5] L. R. Carley and J. Kenney, "A 16-bit 4th order noise-shaping D/A converter," *Proceedings of the 1988 IEEE Custom Integrated Circuits Conference*, pp. 21.7.1–21.7.4, Rochester, NY, May 1988.
- [6] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 267–273, April 1989.
- [7] R. J. Van De Plassche, "A monolithic 14-bit D/A converter," *IEEE J. Solid-State Circuits*, vol. SC-14, no. 3, pp. 552–556, June 1979.

- [8] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. 33, no. 3, pp. 249–258, March 1985.
- [9] E. F. Stikvoort, "Some remarks on the stability and performance of the noise shaper or sigma-delta modulator," *IEEE Trans. Commun.*, vol. 36, no. 10, pp. 1157–1162, Oct. 1988.
- [10] S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Trans. Circuits Sys.*, vol. CAS-34, no. 6 pp. 593–603, June 1987.
- [11] C. Wolff and L. R. Carley, "Modeling the quantizer in higher-order delta-sigma modulators," *Int. Symp. Circuits Sys.*, vol. 4, pp. 2335–2339, Helsinki, Finland, June 1988.
- [12] J. Kenney and L. R. Carley, "CLANS: A high-level synthesis tool for high resolution data converters," *Proceedings of the 1988 IEEE International Conference on Computer-Aided Design*, vol. 1, Santa Clara, CA, Nov. 1988.
- [13] J. G. Kenney and L. R. Carley, "Design of multi-bit noise-shaping data converters," *Analog Int. Circuits Signal Proc. J.* (Kluwer), vol. 3, pp. 259–272, May 1993.
- [14] L. R. Carley, "An oversampling analog-to-digital converter topology for high resolution signal acquisition systems," *IEEE Trans. Circuits Sys.*, vol. CAS-34, no. 1, pp. 83–91, Jan. 1987.
- [15] J. W. Scott, W. Lee, C. Giancarlo, and C. G. Sodini, "A CMOS slope adaptive delta modulator," *ISSCC Dig. Tech. Papers*, pp. 130–131, 1986.
- [16] J.-B. Shyu, G. C. Temes, and F. Krummenacher, "Random error effects in matched MOS capacitors and current sources," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 948–955, Dec. 1984.
- [17] D. J. Allstot and W. C. Black, Jr., "Technological design considerations for monolithic MOS switched-capacitor filtering systems," *Proc. IEEE*, vol. 71, pp. 967–985, Aug. 1983.
- [18] J. L. McCreary, "Matching properties, and voltage and temperature dependence of MOS capacitors," *IEEE J. Solid-State Circuits*, vol. SC-16, pp. 608–616, Dec. 1981.
- [19] Burr-Brown Corp., *Product Data Book*, Burr-Brown, Tucson, AZ, 1986.
- [20] Analog Devices Engineering Staff, *Analog-Digital Conversion Handbook*, 3rd ed., Prentice-Hall, Englewood Cliffs, NJ, 1986.
- [21] H. S. Lee, D. A. Hodges, and P. R. Gray, "A self-calibrating 15 bit CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 813–819, Dec. 1984.
- [22] P. W. Li, M. J. Chin, and P. R. Gray, "A ratio-independent algorithmic analog-to-digital conversion technique," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 828–836, Dec. 1981.
- [23] E. Säcker and W. Guggenbühl, "An analog trimming circuit based on a floating-gate device," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1437–1440, Dec. 1988.
- [24] Y. Sakina, "Multi-bit $\Sigma\Delta$ Analog-to-Digital Converters with Nonlinearity Correction using Dynamic Barrel Shifting," Electronics Research Laboratory, College of Engineering, University of California, Berkeley CA, Memorandum No. UCB/ERL M93/63, 1993.
- [25] B. H. Leung and S. Sutarja, "Multi-bit $\Sigma\Delta$ A/D converter incorporating a novel class of dynamic element matching," *IEEE Trans. Circuits Syst.-II*, vol. 39, pp. 35–51, Jan. 1992.

- [26] L. R. Carley, "Trimming analog circuits using floating-gate analog MOS memory," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 1569–1575, Dec. 1989.
- [27] H. J. Schouwenaars, D.W.J. Groeneveld, C.A.A. Bastiaansen, and H.A.H. Termeer, "An oversampled multibit CMOS D/A converter for digital audio with 115-dB dynamic range," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1775–1780, Dec. 1991.
- [28] K. B. Klaassen, "Digitally controlled absolute voltage division," *IEEE Trans. Instrum. Measur.*, vol. 24, no. 3, pp. 106–112, June 1975.
- [29] J. H. Lindholm, "An analysis of the pseudo-randomness properties of subsequences of long m -sequences," *IEEE Trans. Info. Theory*, vol. IT-14, pp. 569–576, July 1968.
- [30] J. L. Manos, *Some Techniques for Testing Pseudo-Random Number Sequences*, Lincoln Labs, Lexington, MA, Tech. Note 1974–44, 1974.
- [31] A. C. Davies, "Properties of waveforms obtained by nonrecursive digital filtering of pseudo-random binary sequences," *IEEE Trans. Computers*, vol. C-20, pp. 270–281, March 1971.
- [32] F. Chen and B. H. Leung, "A high resolution multibit sigma-delta modulator with individual level averaging," *IEEE J. Solid-State Circuits*, vol. SC-30, no. 4, pp. 453–460, April 1995.
- [33] L. L. Toumelin, P. Carbou, Y. Leduc, P. Guignon, J. Oredsson, and A. Lindberg, "A 5-V CMOS line controller with 16-bit audio converters," *Proceedings of the 1991 IEEE Custom Integrated Circuits Conference*, pp. 24.5.1–5, San Diego, CA, May 1991.
- [34] S. K. Tewksbury and R. W. Hallock, "Oversampled, linear predictive and noise-shaping coders of order $N > 1$," *IEEE Trans. Circuits Sys.*, vol. CAS-25, pp. 436–447, July 1978.
- [35] B. P. Agrawal and K. Shenoi, "Design methodology for $\Delta\Sigma$," *IEEE Trans. Commun.*, vol. COM-31, pp. 360–369, 1983.
- [36] G. Zames and N. A. Shneydor, "Dither in nonlinear systems," *IEEE Trans. Automatic Control*, vol. AC-21, pp. 660–667, Oct. 1976.
- [37] L. Schuchman, "Dither signals and their effect on quantization noise," *IEEE Trans. Commun. Theory*, vol. COM-12, pp. 162–165, Dec. 1964.
- [38] L. G. Roberts, "Picture coding using pseudo-random noise," *IRE Trans. Info. Theory*, vol. IT-8, pp. 145–154, Feb. 1962.
- [39] A. Chandrakasan, S. Sheng, and R. Broderon, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 27, no. 4, April 1992.
- [40] R. W. Adams and T. W. Kwan, "Data-directed scrambler for multi-bit noise-shaping D/A converters," U.S. Patent 5404142, April 4, 1995.
- [41] R. T. Baird and T. S. Fiez, "Improved $\Delta\Sigma$ DAC linearity using data weighted averaging," *Proc. 1995 IEEE Int. Symp. Circuits Sys.*, vol. 1, pp. 13–16, May 1995.
- [42] R. Schreier and B. Zhang, "Noise-shaped multibit D/A converter employing unit elements," *Electron. Lett.*, vol. 31, no. 20, pp. 1712–1713, Sept. 1995.
- [43] L. E. Larsen, T. Cataltepe, and G. C. Temes, "Multi-bit oversampled $\Sigma\Delta$ A/D converter with digital error correction," *Electron. Lett.*, vol. 24, pp. 1051–1052, Aug. 1988.
- [44] T. Cataltepe, A. R. Kramer, L. E. Larson, G. C. Temes, and R. H. Walden, "Digitally corrected multi-bit $\Sigma\Delta$ data converters," *IEEE Proc. ISCAS'89*, pp. 647–650, May 1989.

- [45] M. Sarhang-Nejad and G. C. Temes, "A high-resolution multi-bit SD ADC with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, pp. 648–660, June 1993.
- [46] R. H. Walden, T. Cataltepe, and G. C. Temes, "Architectures for higher-order multi-bit $\Sigma\Delta$ modulators," *IEEE Proc. ISCAS'90*, vol. 2, pp. 895–898, May 1990.
- [47] Z. Zhang and G. C. Temes, "Multi-bit oversampled S-D A/D convertor with nonuniform quantisation," *Electron. Lett.*, vol. 27, pp. 528–529, March 1991.
- [48] J. C. Candy, W. N. Ninke, and B. A. Wooley, "A per-channel A/D converter having 15-segment μ -255 companding," *IEEE Trans. Commun.*, vol. 24, pp. 33–42, Jan. 1976.
- [49] J. W. Fattaruso, S. Kiriaki, M. de Wit, and G. Warwar, "Self-calibration techniques for a second-order multi-bit sigma-delta modulator," *IEEE J. Solid-State Circuits*, vol. 28, pp. 1216–1223, Dec. 1993.
- [50] J. Goes, J. Franca, N. Paulino, J. Grilo, and G. C. Temes, "High-linearity calibration of low-resolution digital-to-analog converters," *IEEE Proc. ISCAS'94*, vol. 5, pp. 345–348, May 1994.
- [51] T. C. Leslie and B. Singh, "An improved sigma-delta modulator architecture," *IEEE Proc. ISCAS'89*, vol. 1, pp. 372–375, May 1990.
- [52] B. P. Brandt and B. A. Wooley, "A 50-MHz multi-bit sigma-delta modulator for 12-b 2-MHz A/D conversion," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1746–1756, Dec. 1991.
- [53] A. Hairapetian, G. C. Temes, and Z. X. Zhang, "Multi-bit sigma-delta modulator with reduced sensitivity to DAC nonlinearity," *Electron. Lett.*, vol. 27, pp. 990–991, May 1991.
- [54] A. Hairapetian and G. C. Temes, "A dual-quantization multi-bit sigma-delta A/D converter," *IEEE Proc. ISCAS'94*, vol. 5, pp. 437–440, May 1994.

Stephen Jantzi
Richard Schreier
Martin Snelgrove

The Design of Bandpass $\Delta\Sigma$ ADCs

9.1 INTRODUCTION

As described in Chapter 1, the quantization noise of a low-resolution quantizer can be nulled in a narrow frequency range by embedding the quantizer in a feedback loop. This feedback structure allows one to spectrally separate the noise from the input signal. The high degree to which this is feasible for low-frequency signals has led to the rapid commercial development of robust, high-resolution ADCs for such signals. The same principle can also be applied to higher frequency low-bandwidth signals, simply by placing nulls in the quantization noise spectrum across the band of interest [1–3]. The band-reject noise shaping of these bandpass $\Delta\Sigma$ converters results in high signal-to-noise ratios for narrow-band bandpass signals.

Bandpass $\Delta\Sigma$ modulators operate in much the same manner as conventional (low-pass) modulators and retain many of their advantages over Nyquist-rate converters. These advantages include inherent linearity (for single-bit systems), reduced antialias filter complexity, and a robust analog implementation. The design of bandpass converters has much in common with low-pass modulator design. This chapter focuses on the aspects of modulator design which are peculiar to bandpass modulators and so only touches lightly on the elements which are the same as those of low-pass modulators. In particular, Chapter 5 (high-order modulator design) and Chapter 11 (analog circuit design) contain information that applies to bandpass as well as low-pass modulators.

The primary motivation for the development of bandpass converters is the simplicity they impart to systems dealing with narrow-band signals. Such systems include radio-frequency (RF) communication systems, spectrum analyzers, and special-purpose instrumentation for narrow-band sources. In the context of a communication system, early conversion to digital at either the intermediate- or radio-frequency stage results in a more

robust system with improved intermediate-frequency (IF) strip testability and provides opportunities for dealing with the multitude of standards present in commercial broadcasting and telecommunications. In particular, the IF filter can be pushed to the digital side, where testing is systematic and changing filter coefficients is easy, while digital programmability allows a single radio to work in several countries or for several types of communication.

Digital implementation of radios also meshes well with digital modulation schemes, such as those emerging for cellular telephones and digital audio broadcast, since advanced modulation schemes need the type of modem technology which is best implemented digitally. Again, elimination of analog IF filters is a key advantage, because they generally have poor (and poorly controlled) phase performance and hence induce intersymbol interference, whereas digital filters can have exactly linear phase.

The closest competing radio architecture is the “zero-IF” style, in which a quadrature mixer is followed by a pair of low-pass ADCs. The styles can be very similar, both mathematically and in terms of circuits, with the main distinction being whether quadrature mixing is done on the analog or the digital side. Figure 9.1 shows how similar the two techniques are. In practical terms, bandpass converters have to sample faster slewing signals than zero-IF systems but do not need precisely matched analog mixers and modulators. Zero-IF systems can also have difficulties with self-interference, as local oscillators radiate to the front end, and their converters have to contend with $1/f$ noise.

In a manner analogous to low-pass modulators, a bandpass $\Delta\Sigma$ modulator can be constructed by connecting a filter and quantizer in a loop, as shown in Figure 9.2. The resonator may be implemented as a discrete-time filter using, for example, switched-capacitor or switched-current technology or it may be implemented as a continuous-time filter using,

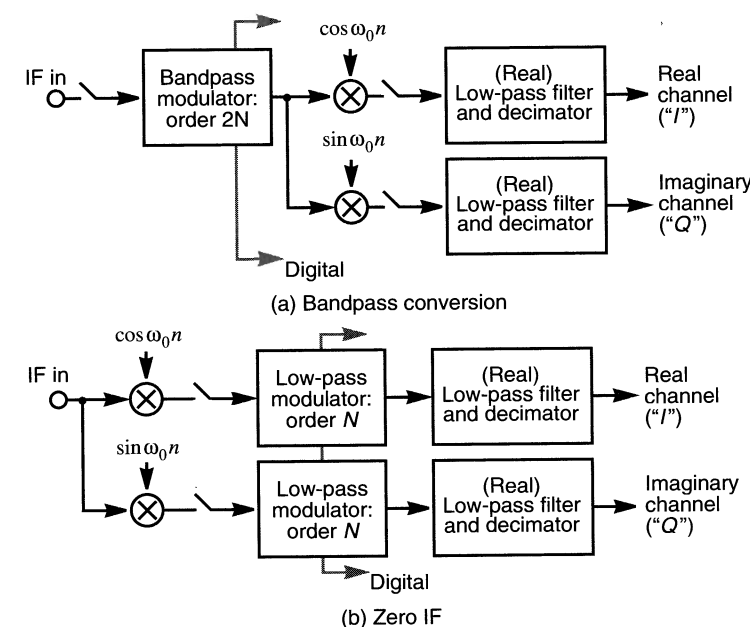


Figure 9.1 Comparison of radios using bandpass $\Delta\Sigma$ and a zero IF with low-pass $\Delta\Sigma$.

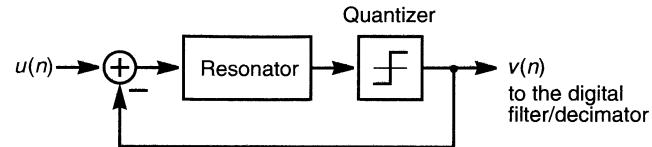


Figure 9.2 A bandpass noise-shaping feedback loop.

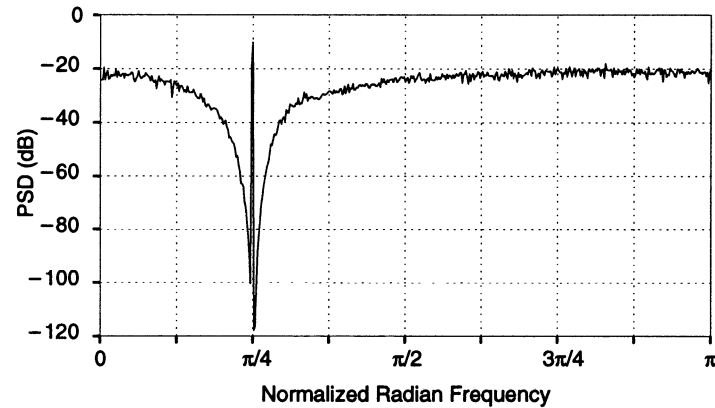


Figure 9.3 The spectrum of the output of an example sixth-order modulator with a -10 -dB peak input (relative to the DAC feedback levels). The bandwidth associated with each point is $\pi/512$ radians.

with a center frequency of $f_s/4$ via a simple mathematical transformation that preserves both the stability performance and the noise properties of the original modulator. For bands centered at frequencies other than $f_s/4$, the design theory developed in Chapters 4 and 5 can be adapted, as we show below.

9.2.1 The Linear Model

Modelling the quantizer in Figure 9.2 as an additive noise source and generalizing to allow the input and feedback to use different feed-ins to the filter yields the “linear model” that was described at length in Chapter 4. The resultant system can be described by

$$V(z) = G(z)U(z) + H(z)E(z) \quad (9.1)$$

The key design issue is to choose a noise transfer function $H(z)$ that minimizes the in-band noise under two constraints: causality and stability.

The loop around the quantizer cannot be delay free, so $H - 1$ must be strictly causal (first impulse-response coefficient zero). This constraint forces

$$\lim_{z \rightarrow \infty} H(z) = 1 \quad (9.2)$$

which indicates that one cannot just force H to zero everywhere. Making $|H|$ small in-band forces it above unity out of band.

Stability is a more difficult problem. Making the linear model stable does not guarantee that the real nonlinear system (which is difficult to analyze because of the hard nonlinearity of the quantizer) is stable. A complete theory of stability adequate for design is lacking, but in general terms if the out-of-band noise gain gets too high overall, then the internal filter states will become very large. This leads to Lee’s rule of thumb for 1-bit quantizers [4], which claims that constraining the peak gain according to

$$|H(e^{j\omega T})| < 1.6 \quad (9.3)$$

will result in a stable modulator. The above form includes some safety margin to allow for component variation and for the approximate nature of the criterion.

These considerations are the same as those used in the design of high-order low-pass modulators and thus bandpass noise transfer function design presents no new hurdles.

9.2.2 Band Location

In a sampled-data system, frequency bands of interest scale with the sampling frequency. For a discrete-time $\Delta\Sigma$ modulator, the noise-shaping band center is placed at a fixed angular frequency ω_0 on the unit circle in the z -plane and thus remains at a fixed fraction of the sampling rate.

For a given input signal center frequency and bandwidth, the choice of angular center frequency is a trade-off among sampling rate, antialiasing filter requirements, and oversampling ratio.¹ Placing the band near $\omega = 0$ increases the oversampling ratio and hence

1. The oversampling ratio is defined, as in the low-pass case, as one-half the sampling rate divided by the width of the band of interest. Thus, a bandpass modulator centered at $\omega_0 = \pi/2$ with a bandwidth of π —which is just sampling at the Nyquist rate—has an oversampling ratio of 1.

9.2 BANDPASS $\Delta\Sigma$ TRANSFER FUNCTION DESIGN

Most of the design art for bandpass modulators can be derived from similar art for the more common low-pass case. In fact, in Section 9.2.3 it will be shown that an arbitrary low-pass $\Delta\Sigma$ converter of order N can be converted to a bandpass modulator of order $2N$

improves the performance achievable by a modulator of a given order. Furthermore, since the first image frequencies that will alias into the signal band are further away, the anti-alias filter requirements are relaxed. Too small an angular frequency, however, can lead to clock rates so high that the loop filter is unable to settle. Moving the band closer to $\omega = \pi$ may be necessary to reduce the clock rate to an acceptable level, but this puts increased demands on the antialiasing filter and on the noise-shaping loop. These trade-offs are similar to those for conventional low-pass modulators.

An additional consideration is that band placement at angular frequencies which are simple fractions of π , such as $\omega_0 = \pi/2$ and $\omega_0 = \pi/4$, allows for innovation in both circuit and decimation algorithm design. In particular, N -path circuits may be used for the modulator (Section 9.3.2), and modulation schemes may be used to simplify the decimator (Section 9.4).

9.2.3 Low-Pass Prototype Method

Figure 9.4 contrasts the pole-zero placements of $H(z)$ for a second-order low-pass modulator and a fourth-order bandpass modulator and illustrates the definitions of ω_B , the normalized bandwidth, and R , the oversampling ratio. In both cases, the modulator will have a small amount of quantization noise in the narrow band surrounding the zeros of $H(z)$, thus ensuring that the modulator output is an accurate representation of the input in the vicinity of those zeros.

The simplest way to design $H(z)$ for a bandpass modulator is to start with a suitable low-pass modulator and apply a low-pass-to-bandpass transformation on it. Such transformations of necessity must increase the order of the modulator. The low-pass prototype must be chosen to satisfy the SNR specifications with an oversampling ratio that is a function of both the oversampling ratio of the bandpass modulator and the transformation employed.

For example, if one were to apply the transformation $z \rightarrow -z^2$ to a low-pass prototype, the zeros of $H(z)$ would be mapped from dc to $\pm\pi/2$. This transformation places the center frequency at $\omega_0 = \pi/2$, and thus for a fixed center frequency the sampling frequency is dictated by the relation $f_s = 4f_0$. Also, since this transformation preserves the oversampling ratio, the oversampling ratio of the prototype modulator is again determined by the signal parameters: $R = 2f_0/B$.

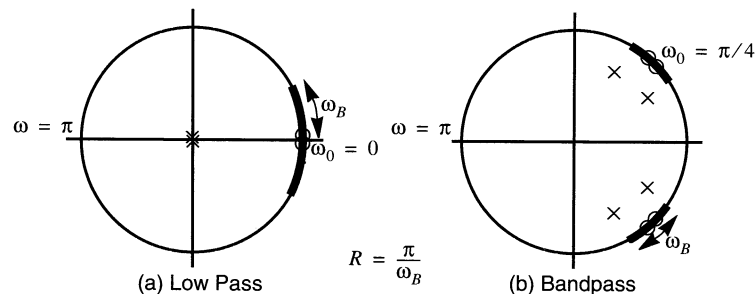


Figure 9.4 The pole/zero locations and passbands of the noise transfer functions for (a) low-pass and (b) bandpass $\Delta\Sigma$ modulation.

The $z \rightarrow -z^2$ transformation is a particularly attractive one since it does not affect the dynamics of the prototype. Specifically, the modulator behaves as a pair of multiplexed low-pass modulators with alternate samples of each modulator negated. As a result, the bandpass modulator is stable if and only if the low-pass modulator is stable and the SNR curves of the modulators are identical. In particular, a fourth-order modulator designed this way [5] can be proven stable, because the prototype low-pass second-order modulator is known to be stable. For the competing radio architectures of Figure 9.1, if the center frequency is to be $f_s/4$, then the bandpass modulator of Figure 9.1(a) can be implemented by transforming the low-pass modulators of Figure 9.1(b) without altering the SNR performance.

Other transformations, such as generalized N -path transformations and low-pass-to-bandpass transformations, are possible but do not possess all the advantages of the $z \rightarrow -z^2$ transformation. Generalized N -path transformations $z \rightarrow \pm z^N$ preserve modulator dynamics but increase the modulator order unnecessarily for $N > 2$ (putting in unnecessary passbands) or result in a passband centered at $f_s/2$ (where aliasing problems occur) for $z \rightarrow z^2$. On the other hand, generalized second-order low-pass-to-bandpass transformations give full control over the passband location but do not preserve modulator dynamics.

Nonetheless, since the discrete-time low-pass-to-bandpass transformation

$$z \rightarrow -z \frac{z+a}{az+1} \quad \text{where } -1 < a < 1 \quad (9.4)$$

preserves both $\lim_{z \rightarrow \infty} H(z)$ and $\max(|H(e^{j\omega})|)$, it preserves both the realizability and Lee stability constraints. The case $a = 0$ degenerates to $z \rightarrow -z^2$; negative a gives systems closer to dc; positive a gives systems with passbands closer to $f_s/2$. The effect on a conventional first-order modulator $H_P(z) = 1 - z^{-1}$ is

$$H(z) = H_P\left(-z \frac{z+a}{az+1}\right) = 1 + \frac{az+1}{z(z+a)} = \frac{z^2 + 2az + 1}{z(z+a)} \quad (9.5)$$

This second-order transfer function can have noise zeros anywhere on the unit circle, approaches 1 as $z \rightarrow \infty$, is stable (as a transfer function, not necessarily as a modulator), and has a peak gain of 2 at $z = \pm 1$. Note also that the new modulator uses a pole at $z = -a$ to control out-of-band gain.

Similarly, transforming the conventional second-order $H_P(z) = (1 - z^{-1})^2$ yields

$$H(z) = \left(\frac{z^2 + 2az + 1}{z(z+a)}\right)^2 \quad (9.6)$$

9.2.4 Design by Generalized Filter Approximator

9.2.4.1 The Design of $H(z)$. A much more flexible approach to transfer function design is via a generalized filter approximator/optimizer [6]. A “least- p th” optimizer available in the filterX program [7] uses a generalization of a least-squares cost function approach to optimization. The optimizer adjusts the poles and zeros of $H(z)$ such that its amplitude response closely matches the measures of ideality defined by the designer. Any deviation from the ideal response increases the cost function. The measures of ideality are set along any number of z -plane contours and are individually weighted for their

importance to the overall optimization. The contours and ideals are set up so as to represent the constraints of Eqs. (9.2) and (9.3). In this way, an “optimum” noise transfer function can be designed for any specifications, while simultaneously meeting the necessary design constraints.

9.2.4.2 The Design of $G(z)$. Simple signal transfer functions do not necessarily require the use of an optimizer. For example, the signal transfer function that results when a single feed-in is used in the cascade-of-resonators structure (see Section 9.3.1) has the same poles as $H(z)$ and $N/2$ zeros at $z = 0$. This is the most convenient signal transfer function for this structure, and by appropriate choice of the poles of $H(z)$, the resulting $G(z)$ can have a very flat passband, as the next section will demonstrate with an example.

If the designer wishes to realize a more exotic signal transfer function, adding feed-ins to each integrator in the loop allows $N - 1$ zeros of $G(z)$ to be placed arbitrarily. This freedom allows a $G(z)$ to be selected that, for example, rejects adjacent radio channels before conversion.

9.2.4.3 An Example Modulator. Figure 9.5 and Table 9.1 give a summary of the optimized z -plane locations of the poles and zeros of $G(z)$ and $H(z)$ for an example

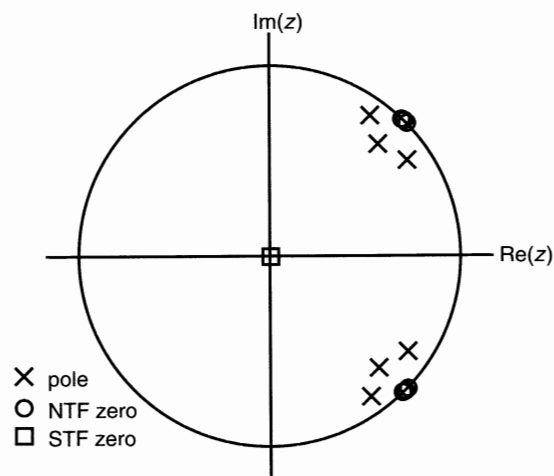


Figure 9.5 Poles and zeros of the noise and STFs. The poles are arranged such that the maximum out-of-band gain of the NTF is 1.5 and the STF is maximally flat at the center frequency.

TABLE 9.1 POLES AND ZEROS OF THE NOISE AND SIGNAL TRANSFER FUNCTIONS

Poles	NTF Zeros	STF Zeros
$0.72708 \pm j0.50177$	$0.71380 \pm j0.70035$	0
$0.57107 \pm j0.59261$	$0.70711 \pm j0.70711$	0
$0.53424 \pm j0.74519$	$0.70035 \pm j0.71380$	0

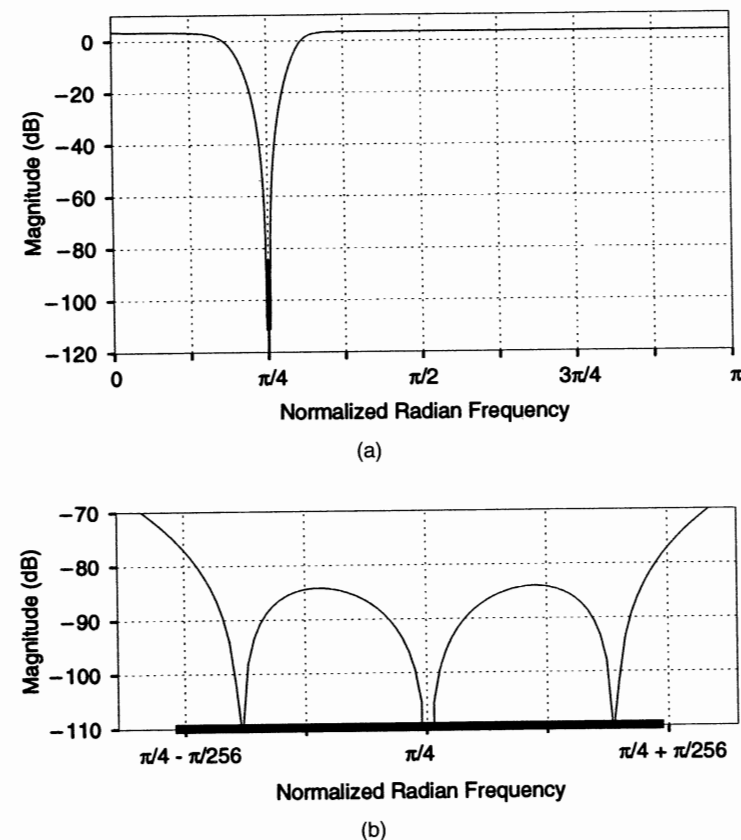


Figure 9.6 The (a) full-band and (b) passband magnitude response of the noise transfer function for the example sixth-order design ($R = 128$).

sixth-order, $f_s/8$ modulator with zeros optimized for $R = 128$. The common poles cluster near $\omega_0 = \pi/4$, noise notches are spread slightly in band, and $G(z)$ has three zeros at $z = 0$. Magnitude plots of $H(z)$ (Figure 9.6) and $G(z)$ (Figure 9.7) show that these transfer functions are of the band-reject and bandpass varieties, respectively.

9.2.5 Modulator Performance

9.2.5.1 Linear Model Predictions. A simple analysis based on the linear model (Section 9.2.1) can be used to estimate the SNR of an N th-order bandpass modulator [3]. One simply assumes the quantization noise is white with power $\frac{1}{3}$ and that the noise transfer function has $N/2$ zeros at ω_0 . The result is that for every doubling of the oversampling ratio the signal-to-noise ratio increases by $3N + 3$ dB, slightly better than half the $6N + 3$ dB/octave rate for a low-pass modulator of order N .

Notice that the total filter order is the same for both radio architectures in Figure 9.1: For the same signal bandwidth and SNR the bandpass approach needs one modulator of order $2N$ to get the same performance that the zero-IF version gets with two modulators of order N . Furthermore, at least when the center frequency is $f_s/4$, a bandpass converter

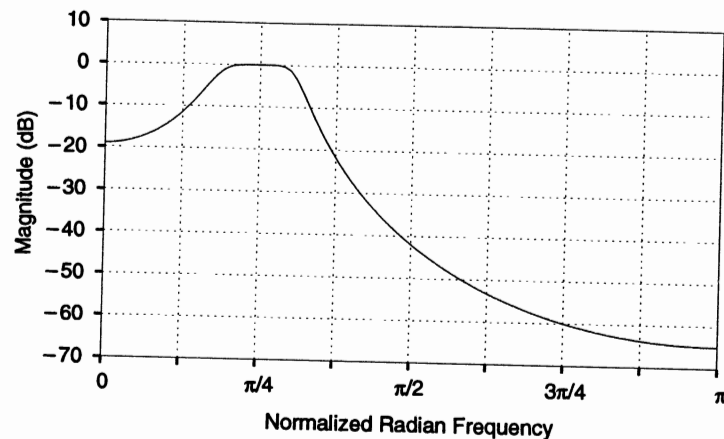


Figure 9.7 The magnitude response of the signal transfer function. The pass-band is flat to within 10^{-4} dB and the phase response is linear to within 0.02 degrees.

with order $2N$ has the same stability performance as a low-pass converter of order N . Finally, just as for low-pass converters, modulator cost is less than proportional to order, since the early stages dominate analog performance and later stages can be implemented more cheaply.

9.2.5.2 Simulations. The linear analysis gives an estimate of the SNR that is good enough to use for a first design iteration and makes it convenient to use a filter approximator for the basic design. However, the linear model neither guarantees stability nor allows an exact prediction of the input signal level that achieves the maximum SNR. Discrete-time simulations can check these overload characteristics, resulting in a curve such as that shown in Figure 9.8. Output noise can be estimated by taking a Hann-weighted discrete Fourier transform (DFT) of the output sequence and summing the power in the in-band “bins,” excluding those containing the input tone. The input should be chosen at a frequency centered in a bin but not at the exact band center because in that case symmetries occasionally cause anomalously good SNR readings.

The noise probability density function and spectrum that are assumed in the linear model are also approximate. The difference equation simulator can be used to check these assumptions. If the quantization noise is white, the output noise spectrum will look like a noisy estimate of $|H(z)|$, and there will be no distortion or in-band tones. Such tones are known to be a problem with some low-pass $\Delta\Sigma$ converters [8–10]. Figure 9.3, given as an example in the Introduction, shows a typical spectrum, and Figure 9.9 gives an expanded view of the in-band spectrum. Since these spectra do not exhibit spikes, it would appear that for this input the modulator does not produce tonal quantization noise. Nonetheless, it is possible for a bandpass modulator to suffer from tone problems identical to those found in low-pass modulators. Specifically, a bandpass modulator derived from a low-pass modulator via a $z \rightarrow \pm z^N$ transformation has exactly the same tonal properties as the prototype.

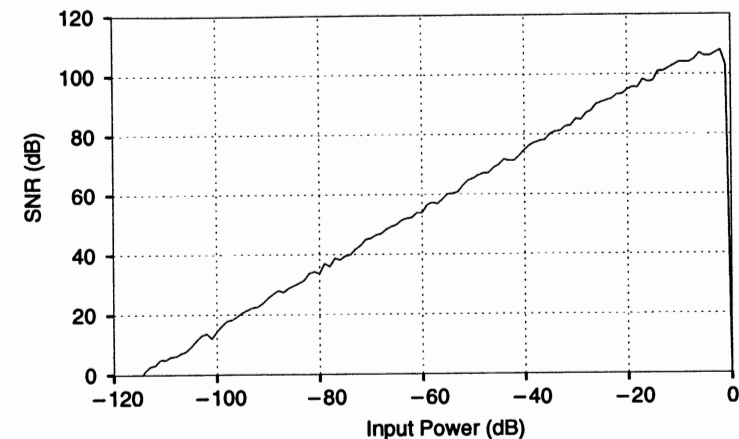


Figure 9.8 The SNR at the modulator output as a function of input signal level. The SNR reaches nearly 110 dB before dropping off at an input peak of -1 dB (relative to the DAC feedback levels).

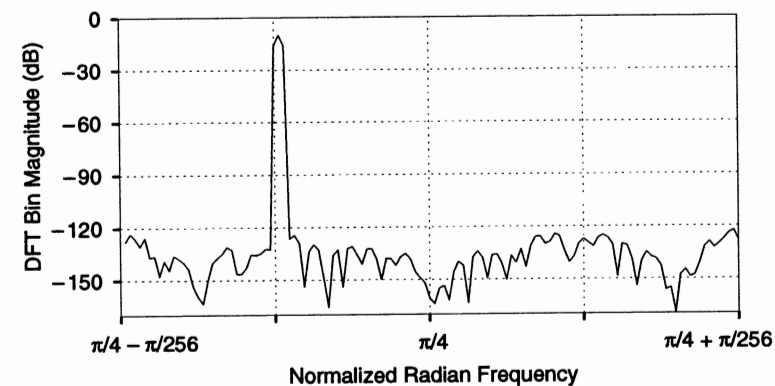


Figure 9.9 The in-band portion of a 32768-point Hann-windowed DFT of the output of the modulator with a -10 -dB input peak (relative to the DAC feedback levels).

9.2.5.3 SNR versus Modulator Order and Oversampling Ratio. Various noise transfer functions were designed, using the methods described above in Section 9.2.4, for several fourth-, sixth-, and eighth-order bandpass $\Delta\Sigma$ modulators. Figure 9.10, the result of a dozen designs tested by nonlinear simulation, verifies the linear model prediction and summarizes the trade-off: SNR increases fairly smoothly with oversampling, with approximately the predicted $3N + 3$ dB/octave slope. A SNR of 80 dB or more is possible at reasonable oversampling ratios.

Note that the zeros of $H(z)$ can be placed coincidentally at band center or placed optimally across the band of interest (as in Section 9.2.4.3) to minimize in-band noise power. A comparison of several modulators (centered at $\pi/2$) shows that a significant SNR

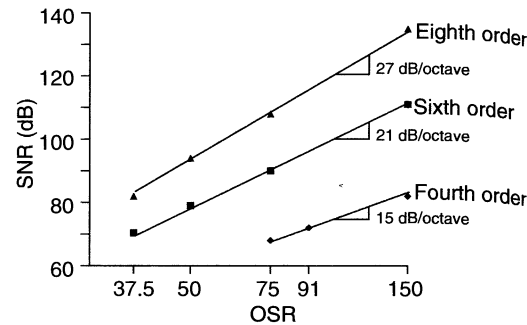


Figure 9.10 SNR versus oversampling ratio and order (for an input peak -10 dB relative to the DAC feedback levels).

advantage is gained with the optimal placement of zeros [11], namely a 5 dB advantage for a spread-zero fourth-order modulator and a 10-dB advantage for a sixth-order modulator. Although coincident zeros simplify the design process and may allow innovative circuits to be used (Section 9.3.2), standard switched-capacitor circuits (Section 9.3.1) can easily incorporate optimized zeros and gain an SNR advantage over circuits that implement coincident zeros.

9.3 BANDPASS $\Delta\Sigma$ MODULATOR DESIGN

In order to produce robust noise shaping in the face of circuit nonidealities, the noise transfer functions must be realized with suitable circuits and structures. Circuit nonidealities can move the zeros of $H(z)$ from their optimal locations, reducing in-band attenuation and increasing in-band noise, which may appear as tones or harmonics.

9.3.1 Standard Switched-Capacitor Design

In a standard switched-capacitor circuit, capacitor ratios set the coefficients of a structure. Mismatch in these ratios, along with other circuit nonidealities, can move the zeros of $H(z)$ and thus lower the modulator SNR. The detrimental effects of zero inaccuracy on bandpass modulators are similar to those for high-order low-pass modulators, but the tolerable zero error (relative to the frequency of the zero) is much lower in bandpass modulators than in low-pass modulators. In rough terms, this occurs because a small percentage error in a capacitor ratio causes a small percentage error in a zero's angular frequency, but the absolute change in the angular frequency, which is what matters, is only small if the nominal frequency is small.

The goal, then, is to find a structure that keeps the in-band magnitude of $H(z)$ small in the face of circuit nonidealities. The gain of the loop filter from the modulator output to the quantizer input is $1 - 1/H(z)$, which shows that the loop filter has poles where noise gain $H(z)$ has notches. Intuitively, infinite feedback gain forces errors to zero. The loop filter therefore has poles on the unit circle and would be an oscillator if left open loop. There are infinitely many structures that implement a given $1 - 1/H(z)$; one example from the literature is shown in Figure 9.11 [12, 13]. This structure is a cascade of three resonators, each of which determines a pole pair of $1 - 1/H(z)$ and hence a notch in the noise

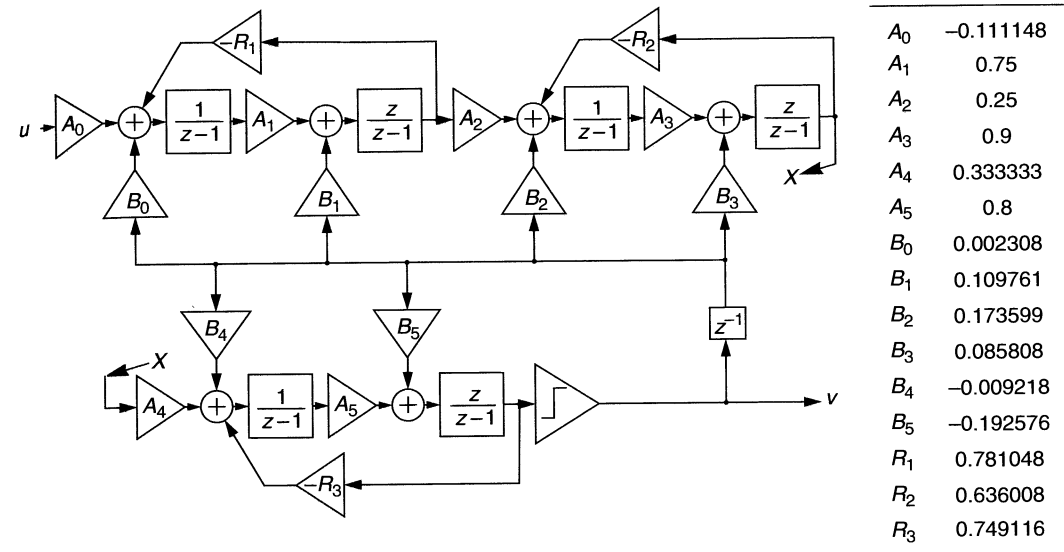


Figure 9.11 A block diagram of the sixth-order example modulator, implemented with the cascade-of-resonators structure and scaled for dynamic range.

gain $H(z)$. Since the lossless discrete integrator (LDI) phased resonators [14] of Figure 9.11 guarantee that the poles are on the unit circle, this is a good topology choice [15]. The “ R ” capacitor ratios set only the angular frequencies of the poles, and thus capacitor mismatch shifts only the angular frequency, not the unit magnitude, of the pole.

Figure 9.12 shows the effect on the in-band noise power of the modulator due to uniformly distributed capacitor errors, as determined by Monte Carlo simulations of the linear model. The noise power distribution suggests that the yield of converters within 3 dB of

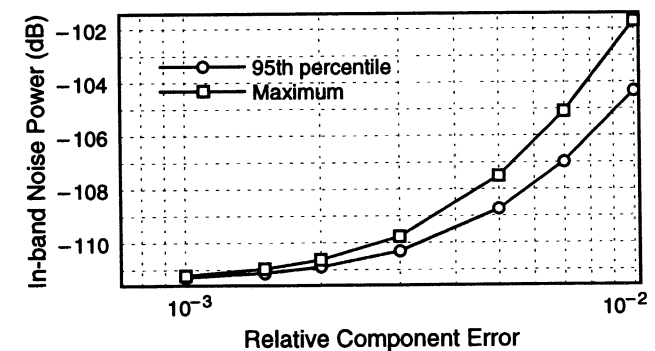


Figure 9.12 The 95th percentile of the in-band noise as a function of the relative error in all capacitors. A 0.5% tolerance causes less than 3 dB loss, whereas a 1% tolerance causes a SNR reduction corresponding to an error of approximately 1 LSB.

nominal SNR should be good.¹ The cascade of resonators is a satisfactory choice for this structure, because even relatively large capacitor errors cause less than one LSB of error.

Finite op-amp dc gain has predictable effects on switched-capacitor-integrator poles and hence on filter pole locations. Finite op-amp gain decreases notch frequency in a manner similar to capacitor mismatch. It also reduces the resonator’s peak gain, thus decreasing pole radius. Decreased pole radius is a more serious concern than a shift in center frequency because it corresponds to a familiar problem in low-pass $\Sigma\Delta$ modulators: stable limit cycles that result in increased distortion and “dead-band” behavior [9, 16]. As is the case for low-pass modulators, higher order modulators are more tolerant of finite gain, with 60 dB of gain being sufficient to maintain the peak SNR within 3 dB of the nominal value for the example modulator.

Figure 9.13 shows the fully differential switched-capacitor circuit that implements the sixth-order example modulator. Note that the fully differential architecture allows one to realize negative coefficients by using polarity-reversed connections to amplifier inputs and outputs.

9.3.2 Switched-Capacitor *N*-Path Design

If the bandpass noise transfer function is derived from a low-pass prototype using a transformation of the form $z \rightarrow \pm z^N$, then the bandpass modulator can be realized by applying the same transformation directly to the circuit of any low-pass modulator. The resultant modulator thus has the same sensitivity properties as the original modulator and can in principle be as immune to component errors as the original modulator. The realization of [5], for example, does this by replacing integrators with two-amplifier circuits constructed with a type of switched-capacitor sample-and-hold (S/H).

Transformation can also be done at the circuit level by replacing the integrators in the low-pass design with “pseudo-*N*-path” integrators. Simulations [17] have shown that, in order for the performance of the bandpass modulator to retain the low sensitivity of the original low-pass modulator, the integrator must be carefully selected. The fact that pseudo-*N*-path circuits clock the switches at frequencies that are subharmonics of the sample clock implies that intermodulation between these lower frequency clocks and the large out-of-band noise present in the modulator loop may be a problem. This intermodulation typically occurs as a result of asymmetries among the various configurations of the circuit, which in turn may be caused by capacitor mismatch, switch mismatch, or even clock mismatch. Intermodulation of this form is disastrous, even at low levels, since it can mix the clocks with out-of-band noise to produce in-band noise and thus degrade the modulator SNR.

Fortunately, circuits exist that are, to first order, independent of the largest source of error, namely capacitor mismatch. Figure 9.14 shows one such block [18]. This circuit employs clocks at $f_s/2$ and so has the potential to mix signals with $f_s/2$. However, since the path capacitors C_A and C_B are used only to store charge and not to form the output voltage from that charge (the C_O output capacitors do that), this circuit does not depend on capacitor matching to achieve perfect matching between paths. As a small aside, note that

1. The distribution is dependent on the oversampling ratio: Tolerable capacitor mismatch is approximately $1/R$.

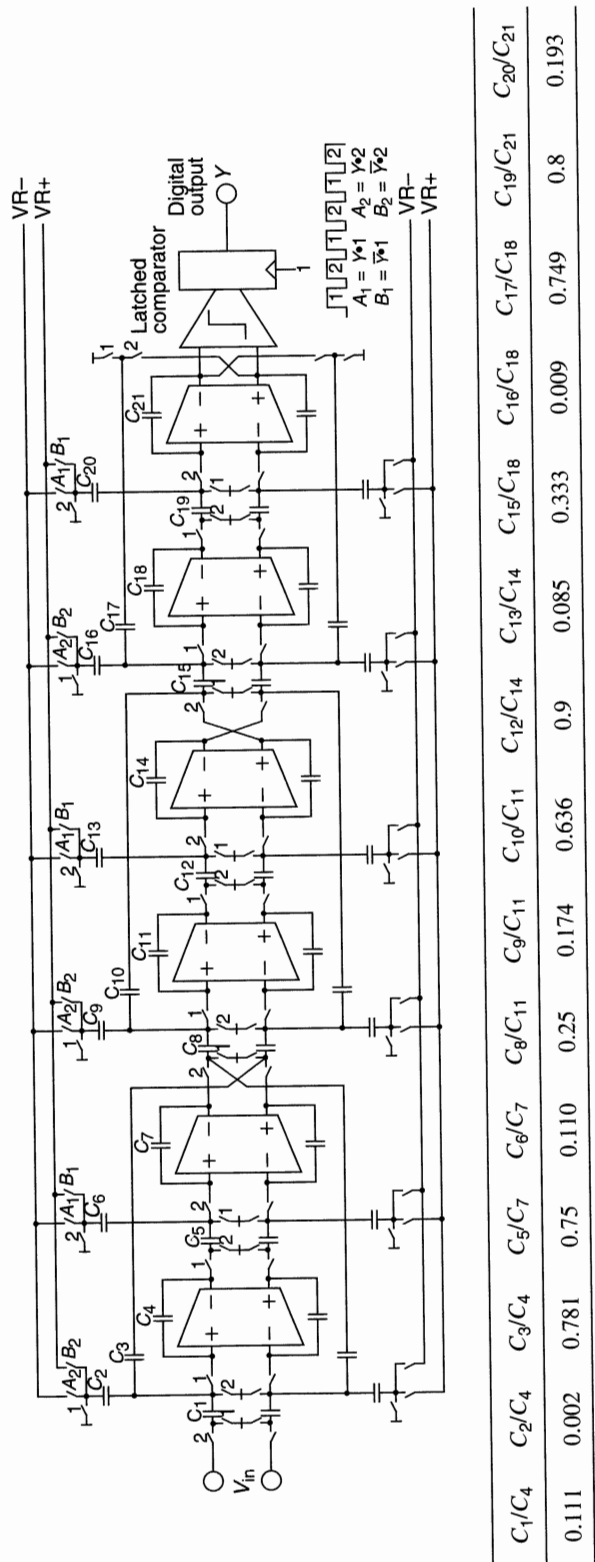


Figure 9.13 The fully differential switched-capacitor circuit that implements the sixth-order modulator. The lower half-circuit is a mirror image of the upper half, except for the DAC feedbacks, which are polarity reversed.

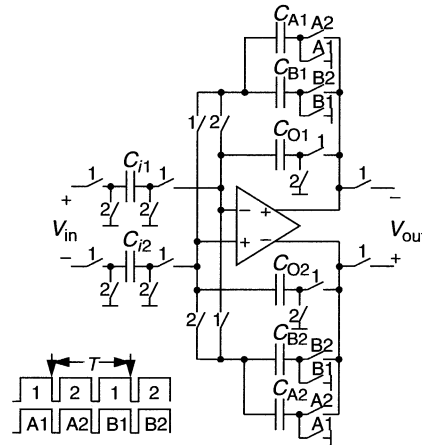


Figure 9.14 An implementation of $-1/(1+z^{-2})$ that is immune to capacitor mismatch.

this circuit can only be implemented in differential form since otherwise the periodic inversion of the state required by the $-1/(1+z^{-2})$ transfer function is impossible using two clock phases.

As with most other switched-capacitor circuits, this circuit depends on the op-amp having infinite dc gain in order to completely transfer charges between the storage capacitors and the output capacitors. Finite amplifier gain reduces resonator Q and also creates coupling between paths. Compensation techniques such as those presented in [19] or [20] can be used to mitigate the effects of finite op-amp gain.

9.3.3 Practical Considerations in Discrete-Time Systems

9.3.3.1 Capacitor and $1/f$ Noise. The kT/C noise present on each capacitor can be assumed to have a flat spectrum from dc to $f_s/2$. As we are concerned strictly with in-band noise, we gain an oversampling factor reduction in the kT/C noise power. This factor allows the minimum required capacitor size to be reduced by the same factor and proves to be an added benefit of oversampling.

Additionally, many capacitors see noise-shaped transfer functions to the output, again reducing the noise gain and hence the minimum required capacitor size. In the sixth-order cascade-of-resonator structure, for example, any capacitor noise injected to the right of a resonator is shaped by that resonator's notch—capacitor noise in the second resonator is shaped by the first resonator's notch, while capacitor noise in the third resonator is shaped by both the first and second resonator notches. The result of these considerations is that the capacitors with the largest contributions to circuit noise are in the first resonator (i.e., the first two op-amp stages) and they are generally the largest capacitors in the circuit.

The $1/f$ noise generated by other circuit components will lie well below the band of interest of most bandpass $\Delta\Sigma$ modulators. This noise will be swamped by out-of-band quantization noise and removed by the digital postfilter and thus is of little concern.

Clock feedthrough and charge injection problems can be reduced by the use of fully differential circuits and early and late clock phases.

9.3.3.2 Op-amp Speed. Satisfactory op-amp bandwidth and slew rate performance are necessary to ensure that sufficient op-amp output settling occurs. As is the case with low-pass $\Delta\Sigma$ converters, it is possible that integrator outputs will change by their maximum swing in each clock cycle, regardless of the input signal voltage. Thus, settling requirements are based on sampling frequency, not on input signal frequency, and bandpass $\Delta\Sigma$ converters have settling requirements that are no more stringent than those of low-pass $\Delta\Sigma$ converters.

9.3.3.3 Sample-and-Hold Circuits. Typical low-pass $\Delta\Sigma$ converters accept inputs with frequencies up to one one-hundredth the clock frequency, whereas a bandpass converter may accept input frequencies up to one-half of the clock frequency. Thus, the input slew rate is much greater in a bandpass converter. For example, a signal at $f_s/4$ can potentially change by its peak voltage in one clock cycle. The S/H operation, which samples the input signal at a specific point in time, must therefore perform within a much narrower window of time in order to give the desired accuracy. Any clock jitter or aperture error can severely degrade the accuracy of the sampled signal.

9.3.4 Continuous-Time Design

Pushing the sampling operation into the noise-shaping loop causes S/H errors to be noise shaped and so reduces the criticality of the S/H block. This is a very strong motivation for designing at least part of the loop with continuous-time circuitry. Continuous-time bandpass modulators are also fast and provide a certain amount of antialias filtering at no cost. On the other hand, continuous-time bandpass modulators, like continuous-time low-pass modulators, are sensitive to memory effects in the DAC feedback and must process continuous-time signals with high linearity. In the bandpass case, meeting the latter requirement is complicated by the fact that the signals of interest are high-frequency ones.

Several researchers have considered using LC [2, 21, 22] or g_m - C [23] resonators to make continuous-time modulators. The LC resonators require the use of (linear) off-chip inductors and complicate the design process by placing added restrictions on the loop transfer function. In contrast, g_m - C resonators offer the advantages of complete system integration and total design freedom, but the linearity of the converter is limited by the linearity of the transconductor.

The basic method for designing a continuous-time bandpass modulator is the same as that for designing a continuous-time low-pass modulator: Start with a prototype discrete-time design and convert it to an equivalent continuous-time design. As described in Section 4.5, the process is, in principle, straightforward, but in practice it can be complicated by numerous details, such as the timing and nature of the DAC feedback. If one assumes that the DAC waveshape is perfectly rectangular and aligned with the clock edge, that is, a non-return-to-zero waveform, then the discrete-to-continuous transformation can be accomplished quite readily using the MATLAB function “d2c”[24].

As an illustration, consider the example modulator of Section 9.2.4. The loop filter transfer function is $L(z) = 1 - 1/H(z)$, and applying a discrete-to-continuous transformation on L results in

$$\hat{L}(s) = -0.6314 \frac{(s - 0.3043)(s^2 + 0.2097s + 0.5133)(s^2 + 0.1510s + 0.7827)}{(s^2 + 0.6169)(s^2 + 0.6319)(s^2 + 0.6020)} \quad (9.7)$$

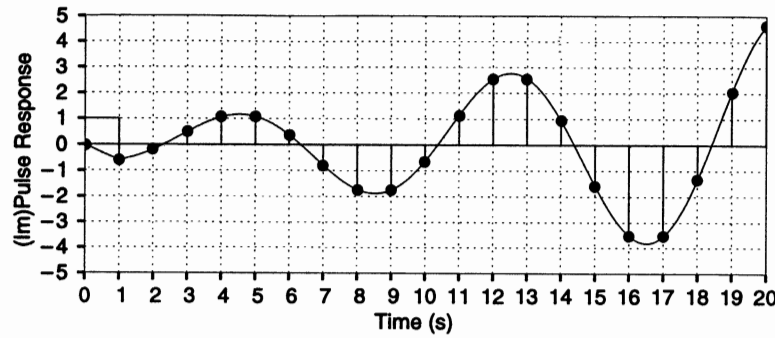


Figure 9.15 The open-loop impulse response of the discrete-time loop filter matches the samples of the pulse response of the continuous-time filter.

This transfer function can be realized readily with a cascade of g_m - C resonators, but it is not possible to realize with a simple cascade of LC resonators driven by current sources since this topology does not allow a numerator as general as that required by the above.

Figure 9.15 plots the open-loop impulse response of the discrete-time loop filter (for a unit input pulse, also shown in the figure) and compares it with the pulse response of the above loop filter. Samples of the latter match the former at integer values of time, as is supposed to be the case.

For other DAC waveforms, such as return-to-zero waveforms, the designer can use time-domain techniques to ensure that $\mathcal{Z}^{-1}\{L(z)\} = \mathcal{L}^{-1}\{\hat{L}(s)DAC(s)\}$ at integer values of time. As an illustration of this process, consider a second-order bandpass modulator with a center frequency of $f_s/4$.

The noise transfer function is $H(z) = (1 + z^{-2})$ and the loop filter is $L_0(z) = 1 - 1/H(z) = 1/(z^2 + 1)$. An unusual feature of this filter is that the first two samples of its impulse response are zero. We can ensure that the same will hold for the continuous-time system if we implement a z^{-1} factor digitally, by delaying the input to the comparator for one clock period with a flip-flop. Thus the system whose impulse response we wish to duplicate is

$$L(z) = \frac{z}{z^2 + 1} \quad (9.8)$$

and the desired impulse response is

$$\sin(\omega_0 n) \quad \text{where } \omega_0 = \pi/2 \quad (9.9)$$

Let the transfer function of the continuous-time loop filter be

$$\hat{L}(s) = \frac{as^2 + b\omega_0 s + c\omega_0^2}{s^2 + \omega_0^2} \quad (9.10)$$

Its step response is then

$$\text{step}(t) = \begin{cases} 0 & t < 0 \\ c + b \sin \omega_0 t + (a - c) \cos \omega_0 t & t > 0 \end{cases} \quad (9.11)$$

If the DAC pulse is of the form

$$\text{dac}(t) = \begin{cases} 0 & t < t_1 \\ 1 & t_1 < t < t_2 \\ 0 & t_2 < t \end{cases} \quad (9.12)$$

then the pulse response of $\hat{L}$ is

$$\text{pulse}(t) = \text{step}(t - t_1) - \text{step}(t - t_2) \quad (9.13)$$

Given t_1 and t_2 , the coefficients for $\hat{L}$ may be found by solving

$$\text{pulse}(n^-) = \sin \omega_0 n \quad \text{for integer } n \quad (9.14)$$

which for $0 < t_1 < t_2 < 1$ results in

$$c \text{ arbitrary} \quad b = \frac{\sin \phi}{2 \sin \theta} \quad a = c - \frac{\cos \phi}{2 \sin \theta} \quad (9.15)$$

and for $0 < t_1 < 1 < t_2 < 2$ results in

$$c = 1 + \frac{\cos(\omega_0 + \phi)}{2 \sin \theta} \quad b = \frac{\sin \phi}{2 \sin \theta} \quad a = c - \frac{\cos \phi}{2 \sin \theta} \quad (9.16)$$

where

$$\theta = \omega_0 \left(\frac{t_2 - t_1}{2} \right) \quad \text{and} \quad \phi = \omega_0 \left(\frac{t_2 + t_1}{2} \right) \quad (9.17)$$

As examples, if $t_1 = 0.5$ and $t_2 = 1$, Eq. (9.14) can be satisfied with $a = 0$, $b = (1 + \sqrt{2}/2)$, and $c = 0.5$, whereas if $t_1 = 0.5$ and $t_2 = 1.5$, Eq. (9.14) can be only satisfied with $a = 0.5$, $b = 1/\sqrt{2}$, and $c = 0.5$.

Now, consider the system shown in Figure 9.16. Its transfer function is

$$T(s) = \frac{sLg_1}{s^2LC + 1} + g_2 = \frac{Lsg_1 + g_2(s^2LC + 1)}{s^2LC + 1} \quad (9.18)$$

which has two free parameters, while a general second-order numerator would give three. In order for this system to be usable as the loop filter in the modulator under consideration, we must have $a = c$, which in turn places the restriction $t_1 + t_2 = 2$ on the DAC waveform. In higher order modulators, where one wishes to implement the loop filter by adding extra LC sections to the left of the system shown in Figure 9.16, there are not a sufficient number of free parameters to realize arbitrary loop transfer functions. In these cases, either

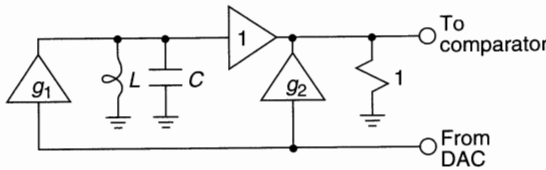


Figure 9.16 An implementation of a second-order loop filter using transconductance amplifiers, an inductor, a capacitor, and a resistor.

the noise transfer function must be constrained or more degrees of freedom created, for example by driving the lower terminals of the inductors with voltage sources, or adding resistors in series with the LC tanks.

9.4 DECIMATION FOR BANDPASS MODULATORS

Just as for low-pass modulators, there are two broad philosophies for decimation of bandpass bit streams: a comb-filter style and a single-stage FIR filter. In the FIR case, changing the coefficients is enough to center the band wherever needed, and a narrow-band signal with all its power in the range (f_1, f_2) can be safely undersampled by a factor R as long as the positive- and negative-frequency bands are kept from overlapping.

As illustrated in Figure 9.17, a complex modulator and (multiplexed) complex low-pass filter may also be used to reduce the bit rate to a manageable level [25]. In this scheme, the bit stream is modulated down to baseband by multiplication by $\exp(-j\omega_0 n)$ and then filtered by a complex low-pass filter. Choosing ω_0 equal to a simple fraction of π allows both the complex modulator and the complex filter to be implemented with relatively simple hardware.

As an example, Figure 9.18 illustrates the conceptual structure of a decimator for the case $\omega_0 = \pi/2$. The multiplying signal is a sum of two period-4 signals containing only 0's and ± 1 's:

$$e^{-j\omega_0 n} = \cos \omega_0 n - j \sin \omega_0 n \quad \text{where } n = 0, 1, 2, \dots \quad (9.19)$$

$$= (1, 0, -1, 0, \dots) + j(0, -1, 0, 1, \dots) \quad (9.20)$$

This allows the multiplication to be performed with a simple exclusive-or gate. The outputs of the two filters are the real (in-phase, or I) and imaginary (quadrature-phase, or Q) channels, which is extremely convenient for the decoding of QPSK signals. Observing that the inputs to the two real filters are zero in alternate cycles allows these filters to be implemented by multiplexing a single low-pass filter.

Similarly, other choices of ω_0 that are rational fractions of π yield periodic modulation signals that can be decomposed into the sum of several 0, ± 1 streams multiplied by

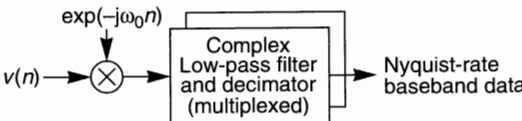


Figure 9.17 A bandpass decimator employing complex modulation.

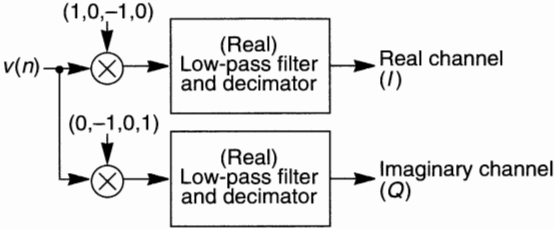


Figure 9.18 A bandpass decimator employing complex modulation. The two filters are actually a single multiplexed filter.

constants [25]. The decomposition can generally be arranged to provide nonoverlapping multiplying streams, which in turn allows a single multiplexed low-pass filter to perform the requisite filtering. After the output of this filter is down-sampled, the streams are recombined with a few additions and multiplications to yield the decimated, bandpass-filtered complex data. The reader may refer to Chapter 13 for a discussion of low-pass decimation filters.

One peculiarity of bandpass $\Delta\Sigma$ is that the I and Q streams are each decimated by a factor of $2R$. Since there are two data streams, the combined word rate is still reduced by a factor of R , just as in the low-pass case.

From this discussion, it should be apparent that the problem of decimation for bandpass $\Delta\Sigma$ converters is only marginally more complex than the low-pass case.

9.5 EXPERIMENTAL RESULTS

The concept of bandpass $\Delta\Sigma$ modulation has been proven experimentally in several discrete-component and monolithic implementations. These implementations have been realized using both the switched-capacitor and continuous-time techniques described earlier.

9.5.1 Reported Implementations of Bandpass $\Delta\Sigma$ Modulators¹

9.5.1.1 September 1990. The first discrete-component bandpass $\Delta\Sigma$ implementation was reported by Dressler in September 1990 [21]. The circuit was a continuous-time implementation of a fourth-order bandpass converter. Bandpass integrators were implemented by bipolar differential amplifiers with LC resonators in their collector paths. The sampling operation was performed by a latched comparator—that is, a comparator followed by a D-type flip-flop. With a 10-MHz sampling frequency, 2.5-MHz signals with an 80-kHz bandwidth were converted with 9–10 bits of effective resolution.

9.5.1.2 September 1991. A second discrete-component continuous-time implementation was described by Thurston et al. in [22]. Their technique converts a digital

1. Working modulators that have been reported in refereed publications are listed in order of their publication date.

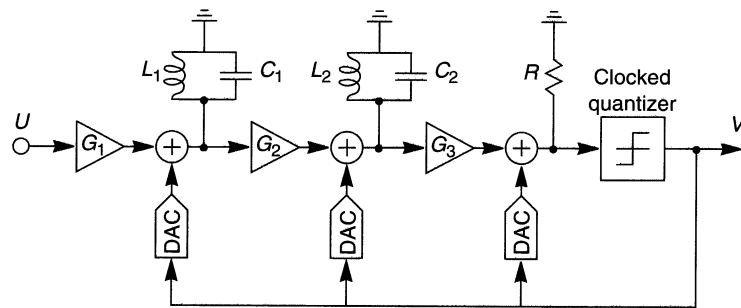


Figure 9.19 A structure for the implementation of a continuous-time bandpass $\Delta\Sigma$ modulator.

modulator into an analog filter based on the impulse-invariant design, where the sampled response to a pulse from the DAC matches the impulse response of the loop filter in the digital modulator as closely as possible (see Section 9.3.4). The structure shown in Figure 9.19 was used, so this equivalence cannot be made exact unless the noise transfer function is constrained.

A sampling frequency of 10 MHz was used for IF signals centered at 2.5 MHz. The quantization noise floor was found to be 117 dB (in a 1-Hz bandwidth) below the converter's overload point, which amounts to a performance level of 67 dB (11 bits) in the desired 100-kHz bandwidth.

9.5.1.3 May 1992. The first monolithic bandpass $\Delta\Sigma$ modulator was reported by Jantzi et al. in May 1992 in [13] and described in more detail in [15] and [26]. A fourth-order modulator was implemented with standard fully differential switched-capacitor techniques. The specifications were relatively lax, and using a 1.82-MHz sampling rate for the conversion of 455-kHz signals with a 10-kHz bandwidth (commercial AM IF), a 63-dB SNR (10.5 bits) performance was achieved.

This modulator was based on the cascade-of-resonators structure described in Section 9.3.1 and was implemented in a 3- μm , $\pm 5\text{-V}$, double-metal, double-polysilicon, CMOS process as a modification of an existing low-pass $\Delta\Sigma$ “platform” chip [12]. This integrated circuit not only proved the feasibility of a silicon implementation but also showed the similarity between low-pass and bandpass $\Delta\Sigma$ modulators by using one to make the other. An unconstrained custom design could have been much more accurate, but the main concern was with the nonlinear dynamics of the overall modulator.

9.5.1.4 June 1992. Another monolithic design was reported shortly after [13] by Tröster et al. in [27] and later, in more detail, in [28]. Their fourth-order discrete-component continuous-time design (see Section 9.5.1.1) was implemented on a 1.2- μm /7-GHz BiCMOS analog/digital array. A latched comparator clocked at 26 MHz performed the sampling operation for the conversion of 6.5-MHz signals with a 200-kHz bandwidth. A performance level of 55 dB SNR (9 bits) was achieved. The converter was designed for a digital pan-European mobile receiver (GSM). BiCMOS technology met the small loop delay requirements of less than 5 ns (with the fast bipolar technology) while still providing CMOS density for the complex digital signal processing.

The bandpass integrators were implemented by differential amplifiers with (off-chip) LC resonators in their collector paths. The second (of two) bandpass integrators was placed in parallel with a positive filter coefficient, α , which was determined empirically by simulation to ensure loop stability.

The main error sources in this design are bandpass integrator leakage, imperfect pulse shaping of the feedback signal (1-bit DAC), loop delay, and noise sources. Loop delay, the sum of the signal delays introduced by each element of the loop, destabilizes the loop: This can be counteracted by increasing α . Increased α , however, causes the loop to behave more as a first-order loop with worse noise performance.

Filtering and decimation were done using a two-stage structure consisting of a 192-tap sinc³ filter and a general low-pass filter.

9.5.1.5 February 1993. A second discrete-time monolithic implementation was described by Longo and Horng [5]. A fourth-order loop filter was designed using standard fully differential switched-capacitor techniques in a 1- μm double-poly CMOS technology with a single 5-V supply. The 75-dB SNDR (12.5 bits) performance was achieved over a 30-kHz bandwidth centered at 1.8 MHz with 7.2 MHz sampling. All poles of the noise transfer function were placed at $z = 0$; thus the dynamics are equivalent to those of a second-order low-pass modulator. As shown in Figure 9.20, unity-gain stages

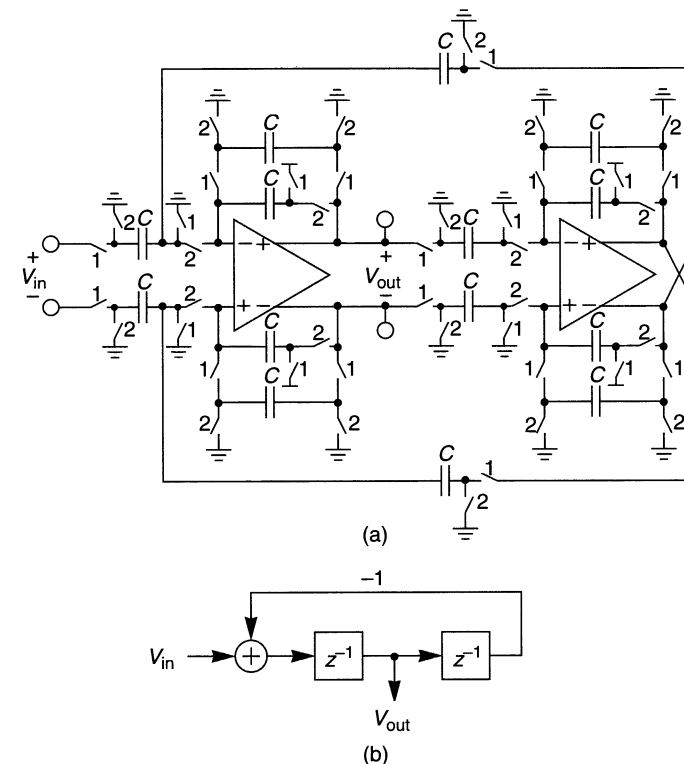


Figure 9.20 (a) The implementation of $z/(1+z^2)$ used in the modulator of [5] and (b) its block diagram equivalent.

were used as analog delay elements to realize the desired transfer function. Since the realization only uses ± 1 coefficients, equal-sized capacitors may be employed, which simplifies the matching problem and also makes capacitor sharing possible.

9.5.1.6 May 1994. A pair of monolithic second-order modulators was reported by Singor and Snelgrove [29]. Both were fully differential switched-capacitor circuits in a 0.8- μm BiCMOS technology with signal bands at $f_s/4$. One used LDI phasing as in [26] and the other used a “forward Euler” loop containing an integrator similar to that of [5]. Both settled completely with a 30-MHz clock. The forward Euler structure performed acceptably (with a 1% error in center frequency resulting from incomplete settling) at a 42.8-MHz clock, allowing for a 10.7-MHz signal. At this clock rate, it obtained a 55-dB SNR in a 200-kHz bandwidth while dissipating 60 mW from a 5-V power supply.

9.5.2 Performance Summary

Figure 9.21 gives a summary of the performance of these modulators based on their bandwidth and resolution. Performance increases as one moves toward the upper right of the figure. Note that the two designs to the lower left are the first implementations of their kind.

9.5.3 Comments on Bandpass $\Delta\Sigma$ Modulator Performance

Bandpass and low-pass converters are used differently, and the fine points of SNR performance are different too. In a radio application the in-band SNR needed is often quite low (FM and most digital schemes do very well at a 40-dB SNR), but there are large out-of-band signals that may intermodulate to produce in-band noise. In contrast, an audio converter requires a 90+ dB SNR and most of the input power is in band. This is one

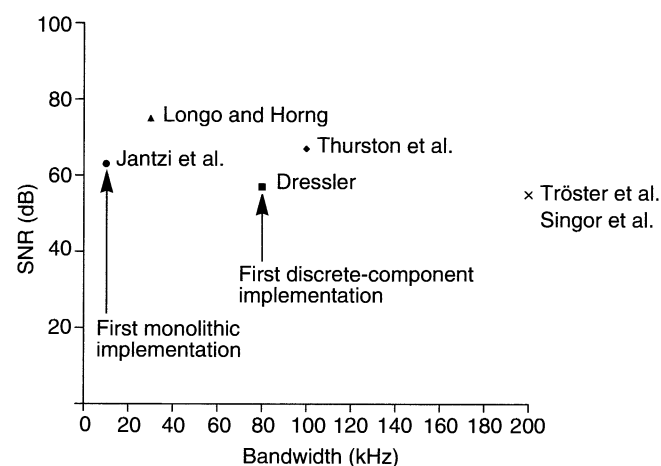


Figure 9.21 Performance summary of the bandpass $\Delta\Sigma$ modulator implementations reported to date. See the following sections for a discussion:
 (■) 9.5.1.1; (♦) 9.5.1.2; (●) 9.5.1.3; (▲) 9.5.1.5 (×); 9.5.1.4 and 9.5.1.5.

reason why shaping the input spectrum with $G(z)$ is desirable in the bandpass case (Section 9.2.4.2).

There is a choice in converting narrow-band high-frequency signals between using a noise-shaped converter and using a conventional video-rate converter, which also benefits from oversampling.

For example, if a signal is centered at 10.7 MHz IF and has a 100-kHz bandwidth, sampling it at $4f_0 = 42.8$ MHz means that it is oversampled at $R = 214$ —almost 8 octaves. “Zero-order” oversampling (i.e., sampling without using feedback to control the spectrum) gives 0.5 bit/octave and an 8-bit ADC could in principle give almost 12-bit resolution in band. This competes with a second-order bandpass $\Delta\Sigma$ converter, which at 1.5 bits/octave gives almost the same resolution by noise shaping a 1-bit quantizer.

Video converters have a great deal of differential nonlinearity, though, and readily produce very high order intermodulation products while $\Delta\Sigma$ converters (particularly at orders of 4 and above) have genuinely noiselike quantization so that filtering really does produce the SNR levels predicted by modeling quantization with noise. For example, a signal just below the threshold of the LSB of an 8-bit converter will never appear in its output stream, and no amount of filtering will bring it up. Likewise, a pair of signals only one or two LSBs in magnitude will systematically produce the intermodulation products of a 1- or 2-bit converter. On the other hand, in a $\Delta\Sigma$ system without deadbands, low-level signals eventually do affect the output and do not produce large intermodulation products. Modulator “deadband” problems are expected to behave in a manner similar to differential nonlinearity, so they have to be controlled.

Filtering does not correct integral nonlinearity, either: If there is an overall nonlinearity from the input S/H circuit, then it will affect the $\Delta\Sigma$ and video converters equally. Integral nonlinearity behaves in the same way as conventional amplifier nonlinearity, becoming more significant for large signals. The $\Delta\Sigma$ converters are therefore [22] more suited to specification by the familiar RF parameters of noise figure, third-order intercept point, spurious-free dynamic range, overload, and bandwidth. Their behavior in radio systems is therefore relatively easy to predict, while video converters can be expected to produce undesirable spurious effects.

It seems that $\Delta\Sigma$ conversion will dominate where high precision is needed. In the case of radio, the choice between $\Delta\Sigma$ and video converters should therefore favor $\Delta\Sigma$, since high linearity is needed for controlling intermodulation.

9.6 THE FUTURE OF BANDPASS $\Delta\Sigma$ MODULATION

9.6.1 High-Frequency Converters

Radio systems often have intermediate frequencies about a decade apart, because the IF should be higher than the bandwidth of the preceding stage. With present switched-capacitor technology (CMOS at about 1 μm gate length) an IF of 10.7 MHz is practical and an IF of 21.4 MHz is within reach. Thus, this technology is reasonable for a first IF in a broadcast FM receiver (with carriers around 100 MHz) and as a second IF in cellular and cordless telephones, with carriers of 1–2 GHz. Bandwidths up to about 1 MHz still allow a reasonable amount of oversampling, although 100 kHz would be preferable. The 100–

300 kHz bandwidths needed for a variety of digital cellular and cordless telephone standards have already been demonstrated. An SNR performance in the range of 80 dB seems reasonable.

Using transconductance-capacitor filters makes center frequencies in the 100-MHz range look practical with present technology, but the filters are much less linear than switched-capacitor and SNR performance will probably be significantly lower. This is an interesting technology for a first IF in telephony and would even allow the direct conversion of the radio frequency for broadcast FM (though that would probably interest a marketing department more than the engineering side of the house). Bandwidths of a few megahertz are reasonable, which are enough to cover most current applications except for those using extreme spread-spectrum techniques.

Allowing discrete components, especially inductors, makes another order-of-magnitude increase in speed possible at increased manufacturing cost. Off-chip inductors can also be very linear and can have large energy storage and therefore low kT/L noise, although at high speeds memory effects in the feedback DAC may still limit linearity. On-chip inductors are becoming practical as frequencies go up, so it is possible that these ultrahigh-frequency (UHF) techniques will eventually produce monolithic $\Delta\Sigma$ converters.

9.6.2 Bandpass $\Delta\Sigma$ DACs

Bandpass DACs share much of the theory with ADCs, but the circuit concerns are different. The filtering is analog and could be done in either switched-capacitor or continuous-time technology. Potential applications for bandpass $\Delta\Sigma$ DACs lie in digital radio transmitters, where all processing would be done in digital form and then followed by a conversion to analog prior to the antenna. This method is attractive since it allows the output transistors to be operated in a power-efficient switching mode and thus may result in an efficient RF amplifier.

9.7 CONCLUSION

The theory presented in this chapter predicts the operation and usefulness of bandpass $\Delta\Sigma$ converters, and the several working implementations to date confirm this theory. Bandpass $\Delta\Sigma$ converters are very similar to low-pass converters, as demonstrated by the fact that one can be modified to make the other. The three principal differences are as follows:

1. Bandpass converters are immune to $1/f$ noise.
2. Discrete bandpass modulators sample fast-slewing signals and therefore require a S/H circuit with low aperture error. Continuous-time designs are not limited by S/H errors.
3. Bandpass modulators avoid the need for matching between the analog I and Q channels, which is a requirement in zero-IF radios.

REFERENCES

[1] T. H. Pearce and A. C. Baker, "Analogue to digital conversion requirements for HF radio receivers," Proceedings of the IEE Colloquium on system aspects and

applications of ADCs for radar, sonar and communications, London, Nov. 1987, Digest No. 1987/92.

[2] P. H. Gailus, W. J. Turney, and F. R. Yester Jr., "Method and arrangement for a sigma delta converter for bandpass signals," U.S. Patent 4,857,928, filed Jan. 28 1988, issued Aug. 15 1989.

[3] R. Schreier and W. M. Snelgrove, "Bandpass sigma-delta modulation," *Electron. Lett.*, vol. 25, no. 23, pp. 1560–1561, Nov. 1989.

[4] W. L. Lee, "A novel higher order interpolative modulator topology for high resolution oversampling A/D converters," Master's Thesis, Massachusetts Institute of Technology, Cambridge, June 1987.

[5] L. Longo and B.-R. Horng, "A 15b 30kHz bandpass sigma-delta modulator," 1993 *IEEE International Solid-State Circuits Conference Digest of Technical Papers*, pp. 226–227, Feb. 1993.

[6] S. Jantzi, C. Ouslis, and A. S. Sedra, "The design of transfer functions for delta-sigma converters," *Proc. 1994 IEEE Int. Symp. Circuits Syst.*, vol. 5, pp. 433–436, 1994.

[7] C. Ouslis, W. M. Snelgrove, and A. S. Sedra, "filterX: an interactive design language for filters," *Advances in Electrical Engineering Software, Proceedings of the First International Conference on Electrical Engineering Analysis and Design, Computational Mechanics/Springer-Verlag*, New York, pp. 227–240, Aug. 1990.

[8] J. C. Candy and G. C. Temes, "Oversampling methods for A/D and D/A conversion," in J. C. Candy and G. C. Temes, eds., *Oversampling Delta-Sigma Data Converters, Theory, Design, and Simulation*, IEEE Press, New York, pp. 1–25, 1991.

[9] O. Feely and L. O. Chua, "The effect of integrator leak in $\Sigma\text{--}\Delta$ modulation," *IEEE Trans. Circuits Syst.*, vol. CAS-38, no. 11, pp. 1293–1305, Nov. 1991.

[10] J. C. Candy and O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-29, pp. 1316–1323, Sept. 1981.

[11] S. A. Jantzi, R. Schreier, and M. Snelgrove, "A bandpass $\Sigma\Delta$ A/D converter for a digital AM receiver," *Proceedings of the IEE International Conference on Analogue-to-Digital and Digital-to-Analogue Conversion*, Swansea, U.K., pp. 75–80, Sept. 17–19, 1991.

[12] R. W. Adams, P. F. Ferguson, Jr., A. Ganesan, S. Vincelette, A. Volpe, and R. Libert, "Theory and practical implementation of a fifth-order sigma-delta A/D converter," *J. Audio Eng. Soc.*, vol. 39, no. 7/8, pp. 515–528, July/Aug. 1991.

[13] S. A. Jantzi, M. Snelgrove, and P. F. Ferguson, Jr., "A 4th-order bandpass sigma-delta modulator," *Proceedings of the IEEE 1992 Custom Integrated Circuits Conference*, pp. 16.5.1–16.5.4, May 1992.

[14] L. T. Bruton, "Low-sensitivity digital ladder filters," *IEEE Trans. Circuits Syst.*, vol. CAS-22, no. 3, pp. 168–176, March 1975.

[15] S. A. Jantzi, "Bandpass sigma-delta analog-to-digital conversion," M.A.Sc. Thesis, University of Toronto, 1992.

[16] R. Schreier, "Noise-shaped coding," Ph.D. Dissertation, University of Toronto, 1991.

[17] R. Schreier, G. C. Temes, A. G. Yesilyurt, Z. X. Zhang, Z. Czarnul, and A. Hairapetian, "Multibit bandpass delta-sigma modulators using N -path structures," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 593–596.

[18] G. Palmisano and F. Montecchi, "Simplified pseudo- N -path cells for z to $-z^N$ transformed SC active filters," *IEEE Trans. Circuits Syst.*, vol. CAS-35, no. 4, pp. 409–415, April 1988.

- [19] A. Baschiroto, R. Castello, and F. Montecchi, "Finite gain compensation techniques for high-Q bandpass SC filters," *Pro. IEEE Int. Symp. Circuits Syst.*, vol. 3, pp. 2813–2816.
- [20] A. K. Betts and J. T. Taylor, "Finite-gain-insensitive circulating-delay type pseudo-N-path filter," *Electron. Lett.*, vol. 26, no. 23, pp. 1941–1942, 1990.
- [21] H.-J. Dressler, "Interpolative bandpass A/D conversion—experimental results," *Electron. Lett.*, vol. 26, no. 20, pp. 1652–1653, Sept. 1990.
- [22] A. M. Thurston, T. H. Pearce, and M. J. Hawksford, "Bandpass implementation of the sigma-delta A-D conversion technique," *Proceedings of the IEE International Conference on Analogue-to-Digital and Digital-to-Analogue Conversion*, Swansea, U.K., Sept. 17–19, 1991, pp. 81–86.
- [23] O. Shoaie and M. Snelgrove, "Optimal (bandpass) continuous-time sigma-delta modulator," *Proc. 1994 IEEE Int. Symp. Circuits Syst.*, vol. 5, pp. 489–492, 1994.
- [24] The Math Works, Inc., *Matlab*, Version 4.0, Math Works, Natick, MA, 1993.
- [25] R. Schreier and W. M. Snelgrove, "Decimation for bandpass sigma-delta analog-to-digital conversion," *Proc. 1990 IEEE Int. Symp. Circuits Syst.*, vol. 3, pp. 1801–1804, May 1990.
- [26] S. A. Jantzi, W. M. Snelgrove, and P. F. Ferguson, Jr., "A fourth-order bandpass sigma-delta modulator," *IEEE J. Solid-State Circuits*, vol. 28, no. 3, pp. 282–291, March 1993.
- [27] G. Tröster, P. Sieber, K. Schoppe, A. Wedel, E. Zocher, J. Arndt, H.-J. Dressler, H.-J. Golberg, and W. Schardein, "An interpolative bandpass converter on a 1.2 μ m BiCMOS analog/digital array," *1992 Symposium on VLSI Circuits Digest of Technical Papers*, Seattle, WA, pp. 102–103, June 1992.
- [28] G. Tröster et al., "An interpolative bandpass converter on a 1.2 μ m BiCMOS analog/digital array," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 471–477, April 1993.
- [29] F. W. Singor and M. Snelgrove, "10.7 MHz bandpass delta-sigma A/D modulators," *Proceedings of the IEEE 1994 Custom Integrated Circuits Conference*, San Diego, CA, pp. 163–166, May 1–4 1994.

Gabor C. Temes
Shaofeng Shu
Richard Schreier

Chapter 10

Architectures for $\Delta\Sigma$ DACs

10.1 INTRODUCTION

With the exception of some brief discussions at the end of Chapter 1, the preceding text concentrated on the properties and design of $\Delta\Sigma$ ADCs and ignored DACs. In a similar pattern, the papers published on $\Delta\Sigma$ ADCs outnumber those on DACs by a factor of 10 or more. This imbalance does not indicate that DACs are less important commercially than ADCs (the opposite is true) or that they are easier to design and build (their difficulties are comparable). The reasons lie more in the historical circumstances of their development.

This chapter will give a brief summary of the reasons for introducing oversampling as a method of improving the accuracy of DACs, describe the basic structures of $\Delta\Sigma$ DACs, and present the architectures available for the realization of digital noise-shaping loops. The design issues arising in the practical design of $\Delta\Sigma$ DACs and some examples of integrated $\Delta\Sigma$ DACs will be discussed in Chapter 12.

The motivation for introducing the $\Delta\Sigma$ system for D/A conversion is readily understood if we analyze the conditions required for a DAC with, say, 16 bits of accuracy using 3 V reference voltage (an increasingly common design task these days). The voltage corresponding to the permissible one-half least-significant-bit (LSB) error is then $2^{-17} \cdot 3\text{V}$, or about 23 μV . This is a very small value; it is the voltage generated by about a dozen electrons stored in a 0.1-pF capacitor. It is also comparable to the thermal noise present at the input of a typical MOS op-amp. The direct design and fabrication of a DAC with such accuracy normally requires expensive trimming and/or calibration procedures. As will be shown in this chapter, the use of oversampling techniques overcomes the analog accuracy problem by trading digital complexity and speed for the desired insensitivity to analog nonidealities. Since state-of-the-art technologies offer fast and dense digital circuit realizations, this is a worthwhile trade-off.

The basic system diagram of an oversampled noise-shaping DAC is shown in Figure 10.1. The input x to the converter is a multibit digital signal, that is, a stream of digital words with a word length N_0 and a data rate f_N . Usually, f_N is slightly larger than the Nyquist rate of the signal; that is, it is slightly greater than $2B$, where B is the signal bandwidth. This signal is processed by the interpolation filter (IF), which changes the data rate to the oversampled value Rf_N (where R is the oversampling ratio) and also suppresses the spectral replicas centered at $f_N, 2f_N, \dots, (R-1)f_N$. The first two curves of Figure 10.2 illustrate the spectra at the input and output of IF. The oversampled output data has a word length N_1 , which is the same or slightly smaller than N_0 . This signal is then entered into

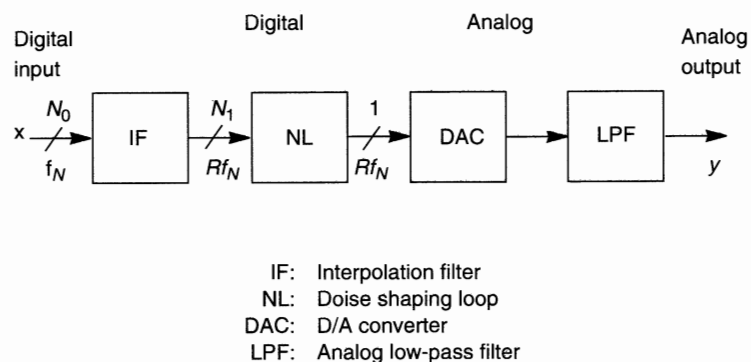


Figure 10.1 Block diagram of a $\Delta\Sigma$ DAC.

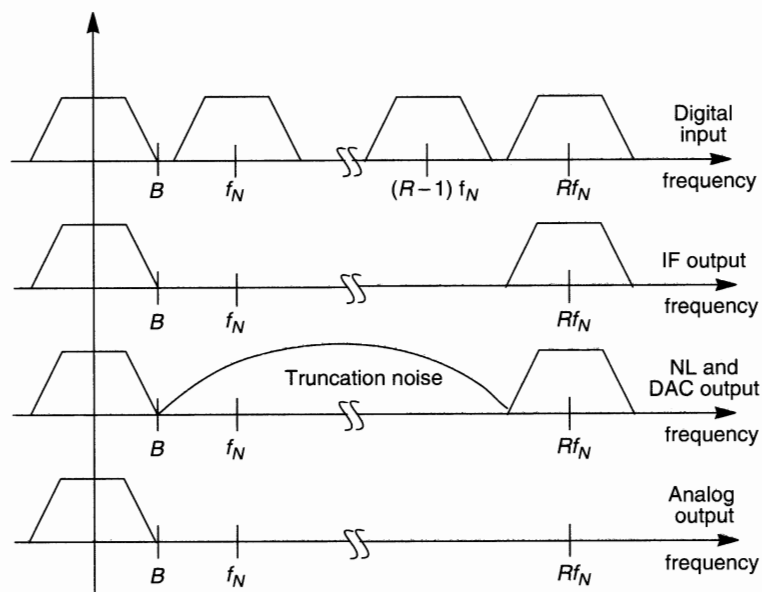


Figure 10.2 Signal and noise spectra in a $\Delta\Sigma$ DAC.

the noise-shaping loop NL (also called the *modulator stage*), which shortens the word length drastically, typically to a single bit. This is done in such a way that most of the large amount of quantization noise power, introduced by the truncation, lies outside of the baseband, at frequencies above B . Next, the truncated output signal y of the NL is converted by the internal DAC, which is the first analog stage in the system. Since it is typically a 1-bit DAC, its realization is conceptually simple, and (as discussed earlier in connection with the 1-bit DAC used in $\Delta\Sigma$ ADCs) it is inherently linear.

The analog output of the internal DAC contains a linear replica of the digital input signal x plus a large amount of noise due to the quantization error e introduced in the NL. Since, as discussed above, most of this noise lies outside of the signal band, it can be almost completely suppressed by the analog low-pass filter (LPF) following the DAC. The last two lines in Figure 10.2 illustrate the spectra of the input and output signals of the LPF.

The theory and design of digital interpolation (as well as decimation) filters will be discussed in detail in Chapter 13. Also, some of the design issues specific to $\Delta\Sigma$ DAC application will be covered in Chapter 12, along with the design problems presented by the analog smoothing filter following the noise-shaping loop. Hence, we shall not discuss these topics at this point.

The overall performance of the complete $\Delta\Sigma$ converter, for ideal operation, is determined by the noise-shaping loop. Its in-band signal gain characteristics and out-of-band quantization noise spectrum should be known before the digital and analog filters (IF and LPF) can be designed. Specifically, the in-band gain response of the loop must be known in advance, since often it has to be equalized by the digital filter. Likewise, the out-of-band noise of the loop must be known before the digital filter can be designed since it limits the amount of minimum stopband loss one should specify for the digital IF. Also, the required stopband loss of the analog filter depends on this noise. Hence, the design of the NL block is the crucial part in the design of the DAC system. In Section 10.2, we shall discuss in detail some of the conventional NL structures, as well as some novel configurations for the NL stage. The design issues of the interpolation filter, modulator, and analog postfilter will then be analyzed in Chapters 12 and 13, which will also describe some published examples of integrated $\Delta\Sigma$ DACs.

10.2 ARCHITECTURES FOR THE NOISE-SHAPING LOOP

Next, we shall describe and discuss some configurations for the noise-shaping loop. The purpose of this stage is very similar to that of the analog noise-shaping loop of a $\Delta\Sigma$ ADC: A finely quantized input signal (in the case of the analog loop, an analog or infinitely finely quantized signal) must be truncated to a very small word length without introducing too much in-band truncation noise.

10.2.1 Delta-Sigma Loop

The idea of using a digital structure similar to that of the $\Delta\Sigma$ modulator is immediately obvious. The general form of this loop is shown in Figure 10.3. In the simplest case, the low-pass filter may be simply an accumulator, resulting in first-order noise shaping

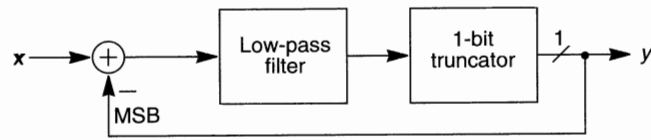


Figure 10.3 General block diagram for a $\Delta\Sigma$ noise-shaping loop.

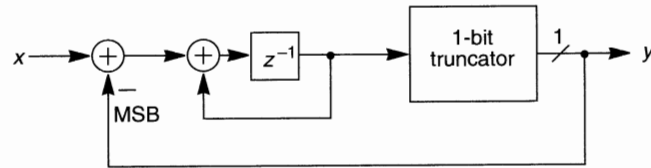


Figure 10.4 A first-order $\Delta\Sigma$ noise-shaping loop.

(see Figure 10.4). For the design of higher order systems, many of the configurations discussed earlier for analog $\Delta\Sigma$ modulator loops can be adapted for operation in digital modulators. For example, Figure 10.5 shows a third-order $\Delta\Sigma$ modulator loop containing a cascade of an accumulator and a second-order resonator, with the MSB output fed back to each stage, while Figure 10.6 illustrates a fifth-order loop, containing two resonators and one accumulator, also with MSB feedback. A similar modulator [1], in which the outputs of all stages are fed forward to the input of the quantizer while the MSB output is fed back

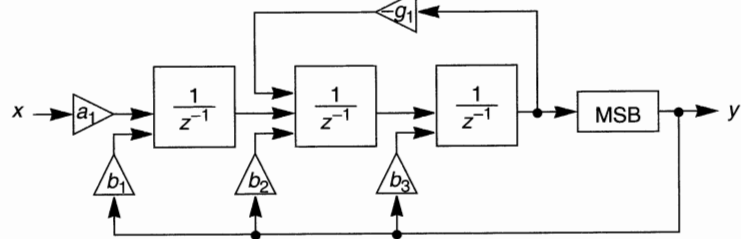


Figure 10.5 A third-order noise-shaping loop using only feedback branches.

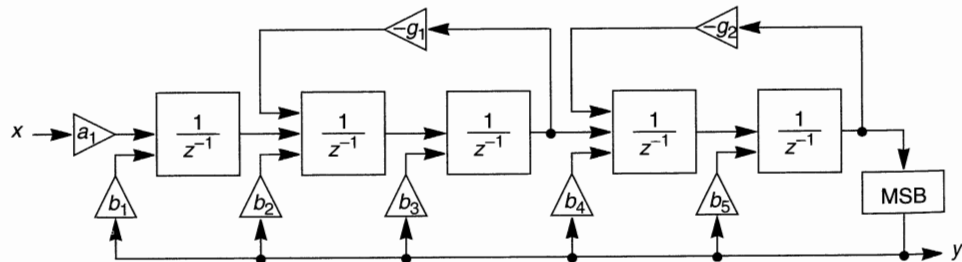


Figure 10.6 A fifth-order noise-shaping loop using only feedback branches.

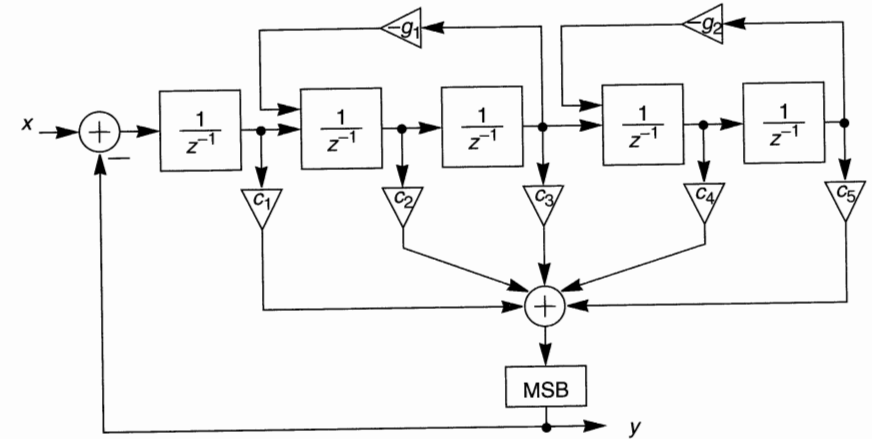


Figure 10.7 A fifth-order $\Delta\Sigma$ noise-shaping loop using both feedforward and feedback branches.

only to the input node of the loop, is shown in Figure 10.7. In these structures, hardware can be saved if the constant factors (e.g., a_1 , b_1 , b_2 , etc., in Figure 10.5) are chosen as sums of a few terms that are integer powers of 2. Such quantization of the coefficients may introduce a slope or even a peak into the nominally flat signal frequency response, but this amplitude distortion can be compensated in the digital filter characteristics. Hardware can also be saved by reducing the number of bits carried in the loop as the signal propagates from left to right. The analysis and design of these structures is similar to their analog counterparts, and therefore will not be discussed here.

10.2.2 Error Feedback Structure

A useful alternative configuration for $\Delta\Sigma$ modulators is schematically illustrated in Figure 10.8. In this structure, instead of the MSB output, the negative of the truncation error (consisting of the neglected LSBs) is fed back to the input through a filter with a transfer function $H_1(z)$. Simple analysis shows that in the z -domain the output of the loop is given by

$$Y(z) = X(z) + [1 - H_1(z)]E(z) \quad (10.1)$$

where Y , X , and E are the z -transforms of the output y , the input x , and the truncation error e of the loop, respectively. This so-called error feedback structure usually leads to a

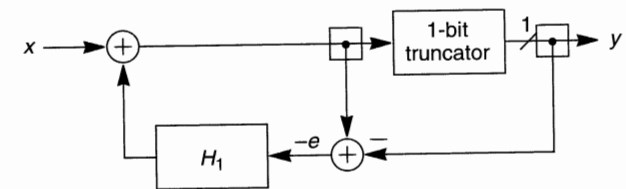


Figure 10.8 Error feedback scheme.

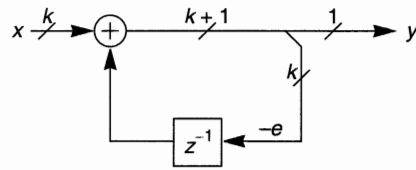


Figure 10.9 First-order error feedback noise-shaping loop.

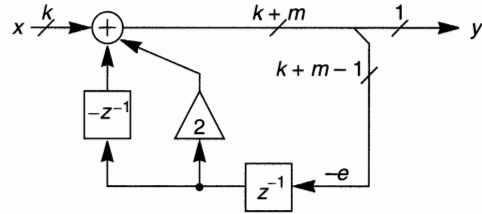


Figure 10.10 Second-order error feedback noise-shaping loop.

simpler realization than the one illustrated in Figure 10.3, discussed above, and is hence often used for digital loops. For analog loops, the high sensitivity of the circuit to any imperfections in either the loop filter or the adders makes this structure impractical.

In the simplest case, $H_1(z) = z^{-1}$ can be chosen as the transfer function of the feedback filter (see Figure 10.9). Then, the NTF becomes

$$H(z) = 1 - H_1(z) = 1 - z^{-1} \quad (10.2)$$

and a first-order noise shaping is achieved. To obtain second-order noise shaping, we can select the feedback filter transfer function as

$$H_1(z) = 1 - (1 - z^{-1})^2 = z^{-1}(2 - z^{-1}) \quad (10.3)$$

Figure 10.10 shows the corresponding noise-shaping loop. Note that neither of these structures requires multiplication of multibit words; only delays, additions, and shifts of the binary point are needed. To prevent overflow, a limiter may be added to the circuit [2], as shown in Figure 10.11. Alternatively, as shown in Figure 10.12, the circuit can be operated with 2-bit (rather than single-bit) output, and a second first-order loop can be added to generate the final 1-bit output data [3]. The overall performance of the system is slightly worse than if the input x would be directly fed to the input of the second loop; however,

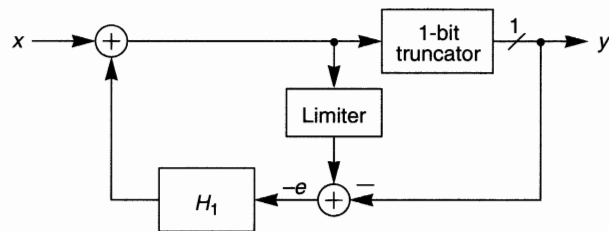


Figure 10.11 Error feedback loop with limiter.

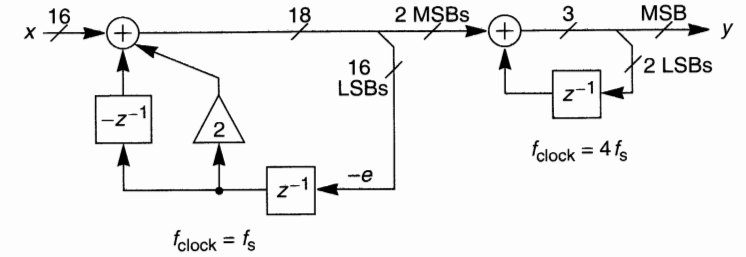


Figure 10.12 A first-order $\Delta\Sigma$ noise-shaping structure in which the fast stage has a 2-bit input signal.

the complexity of the fast ($f_{\text{clock}} = 4f_s$) digital hardware is greatly reduced at the cost of added slow ($f_{\text{clock}} = f_s$) components. In fact, the fast stage can be realized by only a small RAM and a shift register [3].

10.2.3 Cascade Structure

As was the case for analog modulator loops, the cascade (MASH) realization of digital noise-shaping loops is another option. Figure 10.13 shows a second-order modulator realized as a cascade of two first-order loops [4]. Unfortunately, the output of the digital circuit is then the sum of two single-bit words, and hence a 2-bit DAC is needed with the same linearity requirement as that of the overall system. To avoid this problem, it is possible to convert the outputs of the two loops individually into analog form using two 1-bit DACs and combine their analog outputs in a summing stage. Imperfect gain matching between the two paths introduces first-order-filtered noise leakage from the first stage, but at least the signal will not be distorted since it passes through only a single-bit D/A conversion. Alternatively, another simple $\Delta\Sigma$ stage with a higher sampling frequency can be added to reduce the word length to 1 bit, as in Figure 10.14.

A structure in which both the quantization error and the single-bit output data are fed back (and which thus combines single-bit and multibit feedback) is shown in Figure 10.14

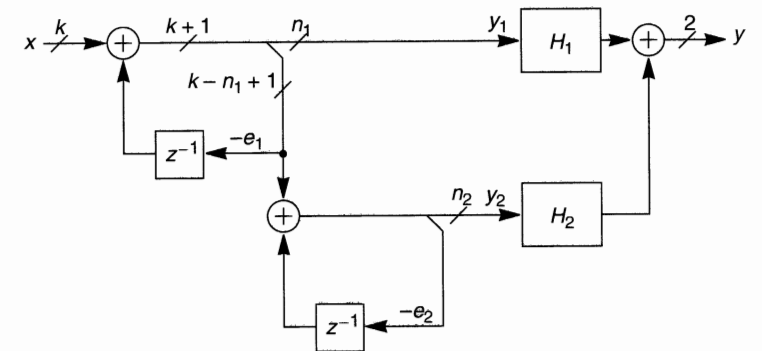


Figure 10.13 Cascade structure for a second-order noise-shaping loop.

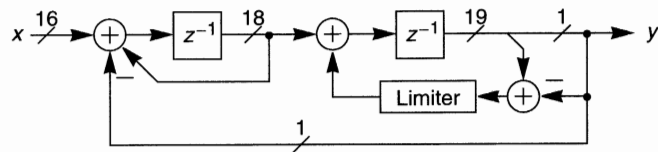


Figure 10.14 A second-order noise-shaping loop with both multibit and single-bit feedback paths.

[5]. Analysis gives the output signal in the z -domain:

$$Y(z) = z^{-2}X(z) + (1 - z^{-1})^2 E(z) \quad (10.4)$$

Thus, the circuit functions as a truncation loop with second-order noise shaping.

10.2.4 Multibit Quantizer Loops

In Chapter 8, the advantages of incorporating multibit, rather than single-bit, quantizers in $\Delta\Sigma$ ADCs were discussed. These included improved stability properties, lower speed requirements, and reduced demands for the postfilter. The generation of tones in the loop is also easier to prevent in such structures. As explained there, however, in order to obtain these improvements the detrimental effects of the inherent nonlinearities of the internal DAC had to be reduced at the cost of some additional hardware. The situation is very similar for $\Delta\Sigma$ DACs. As shown in Figure 10.1, the internal DAC is directly in the signal path, and hence the errors caused by its nonlinearities affect the signal without any noise filtering or attenuation. Hence, steps must be taken to reduce or cancel the distortion caused by these DAC nonlinearities.

One option available is to use one of the “element-swapping” methods discussed in Section 8.3. As explained there, some of the techniques, such as the randomizing approach, convert the errors caused by the nonlinearities into random (or pseudorandom) noise. Others, such as the barrel-shifting or individual-level-averaging technique, achieve a reduction of in-band noise by modulating the nonlinearity-caused noise to a higher out-of-band frequency. The reader is referred to Section 8.3 for details. These techniques remain applicable in the present application as well. Dithering can also be used to reduce harmonic distortion caused by DAC nonlinearity.

An alternative way of reducing the in-band noise and harmonic distortion caused by the nonlinearity of the internal DAC is to use continuously calibrated current copiers [6] or capacitor averaging [7]. An in-band noise reduction corresponding to a linearity of 15–16 bits has been achieved this way.

The digital correction technique described in Section 8.5 can also be applied in $\Delta\Sigma$ DAC design. Figure 10.15 shows the block diagram of a digitally corrected DAC with a multibit internal DAC [8]. In this system, the RAM contains multibit words representing very accurately the actual output levels of the N -bit internal DAC. Since the internal DAC and the RAM have the same inputs, their outputs (analog and digital, respectively) are also corresponding values. Also, since the low-frequency (baseband) components of the RAM output data track the digital input x very accurately due to the feedback loop, the same

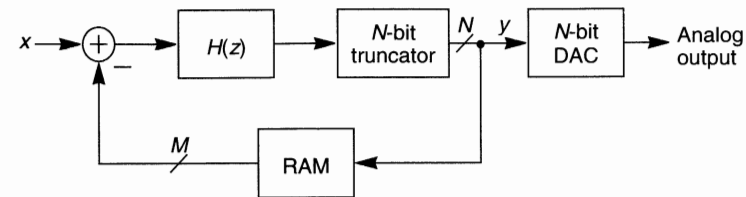


Figure 10.15 A digitally corrected N -bit loop.

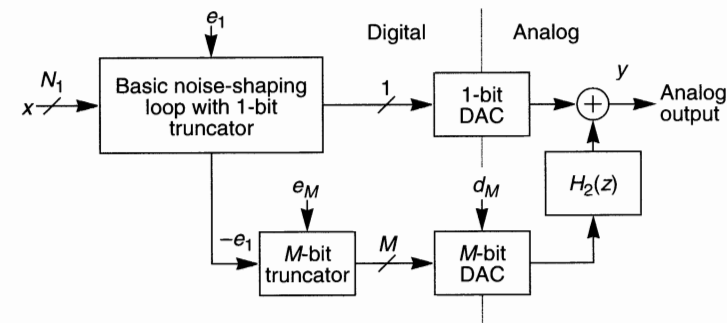


Figure 10.16 A dual-truncation $\Delta\Sigma$ noise-shaping loop.

conclusion holds for the DAC output. In conclusion, the output y of the truncator is predistorted by the inclusion of the RAM in the feedback loop in such a way that the low-frequency spectrum of the DAC output is an accurate analog replica of the digital input x .

The data stored in the RAM in the system of Figure 10.15 can be acquired using the same calibration process as was used in the case of the digitally calibrated multibit ADC, described in Section 8.5. To save some digital hardware, it is also possible to store the digital equivalents of the errors in the DAC levels, rather than the DAC levels themselves, and to combine the RAM's input and output signals [9] via an additional noise-shaping loop.

The dual-quantization techniques discussed in Section 8.6 also have equivalents for $\Delta\Sigma$ DACs. Figure 10.16 shows a dual-truncation DAC structure [10] somewhat similar to the Leslie-Singh ADC architecture [11]. In this system, the negative of the large quantization error e_1 due to the single-bit noise-shaping loop is fed into a correction path, where it is truncated to M bits ($M > 1$). The truncated error is then converted into analog form, high-pass filtered in the analog filter $H_2(z)$, and added to the analog output of the 1-bit DAC to cancel (in practice, to reduce) the noise e_1 in the output signal y . It is replaced in y by the truncation error of the M -bit truncator and the nonlinearity error of the M -bit DAC, filtered by $H_2(z)$.

One important advantage that this scheme (as well as the dual-truncation schemes described in the next paragraphs) has over the corresponding ADC architectures is that for accurate cancellation of e_1 the DAC schemes require the accurate low-frequency realization of analog *differentiators* in the correction filter $H_2(z)$, while the ADC systems require

accurate analog *integrators*. An analog differentiator must realize a transmission zero at dc, which a series branch containing an unswitched capacitor can do very precisely, while the integrator needs a pole at dc, which cannot be realized accurately with practical finite-gain op-amps. Under ideal conditions, the corrected output y of the noise-shaping loop no longer contains the noise-shaped e_1 , but rather the M -bit truncation error e_M shaped the same way as e_1 was. This results in an improvement of nearly M bits in the SNR. In practice, a leakage of filtered e_1 noise into y will always occur. However, due to the zeros at dc in the noise filter function, this effect need not be significant if the system is carefully designed. Similarly, the nonlinearity error d_M of the M -bit DAC (which is usually fairly small to start with) is high-pass filtered and thus further reduced by the analog filter $H_2(z)$.

Figure 10.17 illustrates a DAC system similar to the dual-quantizer MASH structure proposed for ADCs by Brandt and Wooley [12]. In this system, instead of simply truncating e_1 to M bits, a second noise-shaping loop is incorporated into the correction path that shapes the multibit truncation error e_M . As a result, the error due to e_M is even more reduced; the order of the overall noise shaping is now the sum of the orders of the two noise-shaping loops. Figure 10.18 shows the block diagram of a third-order DAC based on this scheme. In the diagram, C_1 and the associated switches realize the 1-bit DAC, while C_2 , C_3 , and C_4 and their switches perform the function of the $H_3(z)$ filter. Note that the transfer function of this filter will have a double zero at $z = 0$ regardless of capacitor matching or finite op-amp gain. Note also that the pole of the loop filter at $z = 0.5$ helps to stabilize the loop and enhances its dynamic range.

An alternative dual-truncation multibit MASH noise-shaping stage [13] is shown in Figure 10.19. Both stages use two accumulators; to enhance stability, both incorporate a scaling by $\frac{1}{4}$. The truncator Q_1 in the first loop has three possible output levels: -1 , 0 , or 1 . Here, Q_2 is a two-level truncator: its outputs can be ± 1 . As a result, the path outputs D_{out1} and D_{out2} each have three possible levels: $D_{out1} = 0, \pm 1$ and $D_{out2} = 0, \pm 2$. The overall output $D_{out1} + D_{out2}$ can be $0, \pm 1, \pm 2$, or ± 3 .

The input to the second stage is $z^{-1}X_1 - D_{out1}$, where X_1 is the output signal of the first accumulator of the first stage. Detailed analysis [13] indicates that the system nearly completely cancels the truncation error of Q_1 and provides a third-order noise shaping for the truncation noise generated by Q_2 . To prevent harmonic signal distortion, it is preferable to use two separate DACs to convert D_{out1} and D_{out2} . A possible implementation [13]

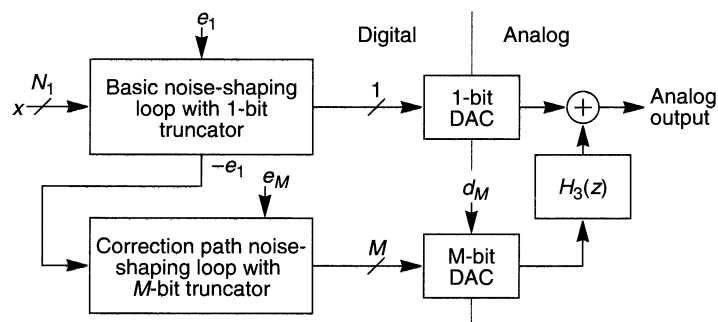


Figure 10.17 A dual-truncation MASH structure.

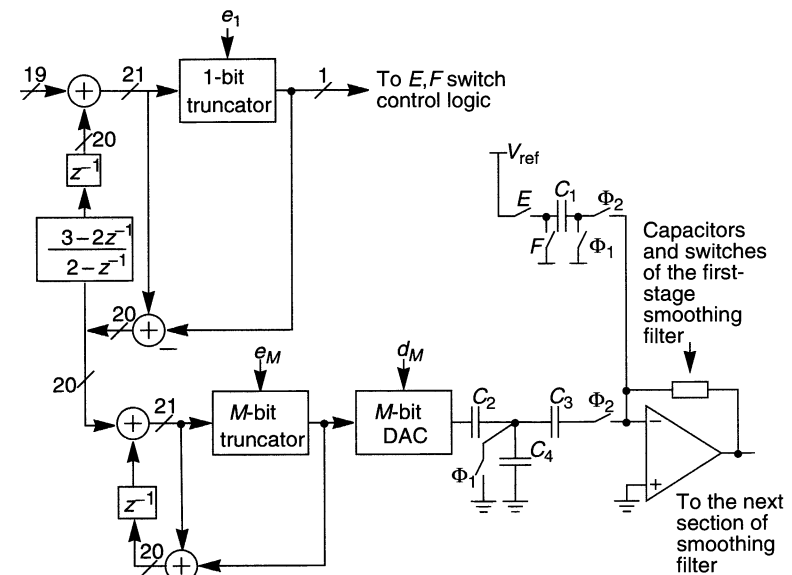


Figure 10.18 A third-order dual-truncation MASH noise-shaping stage.

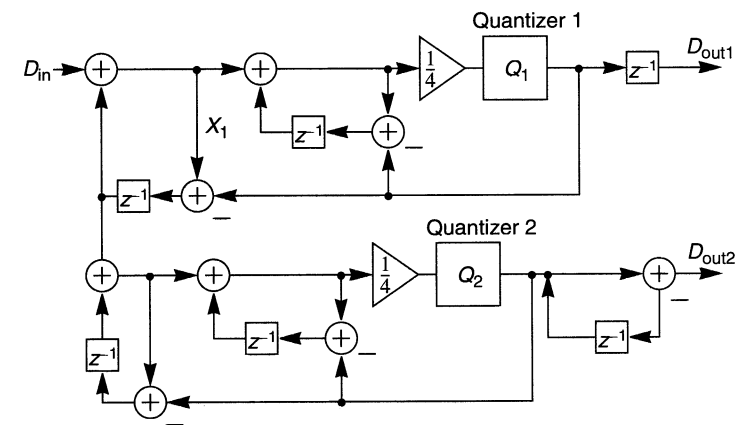


Figure 10.19 Two-stage multibit MASH stage.

is shown in Figure 10.20, where CD_1 and CD_2 perform the digital-data-to-analog-charge conversion. Note that the three possible analog input charges generated by CD_1 (CD_2) must accurately satisfy a linear relation to the values of D_{out1} (D_{out2}). This may be difficult to achieve in a practical circuit.

A different dual-truncation architecture, where the 1-bit and M -bit outputs are both fed back within a single loop, is shown in Figure 10.21. It is equivalent to the dual-feedback single-path ADC system of Hairapetian [14, 15]. Here the $H_5(z)$ block is an

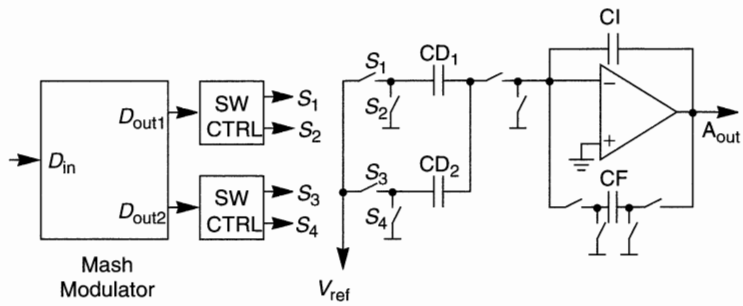


Figure 10.20 DAC implementation.

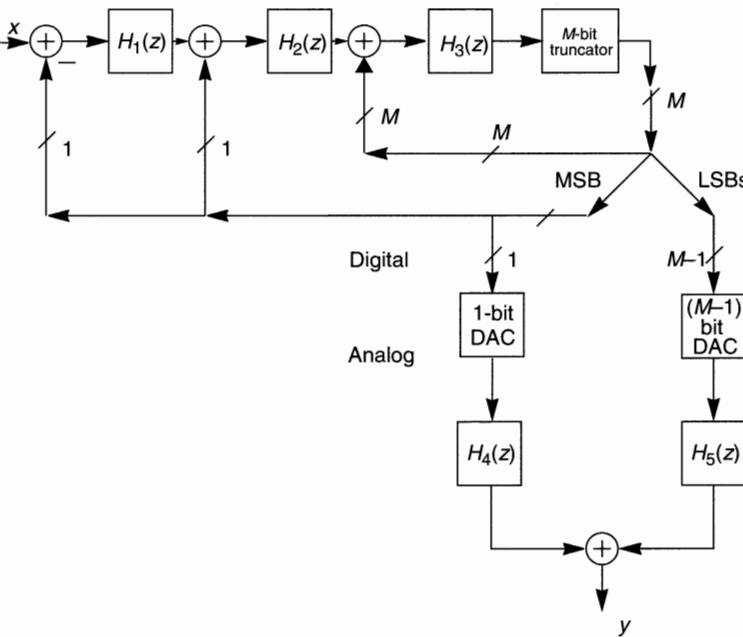


Figure 10.21 A single-stage dual-truncation D/A loop.

analog high-pass filter, and hence the nonlinearity error of the $(M - 1)$ -bit DAC is suppressed in the baseband.

It was pointed out recently [16] that if a linear multibit internal DAC can be realized (using one of the methods described in Chapter 8), then one can partition the N -bit input words into M -bit MSBs and $(N - M)$ -bit LSBs, use a digital noise-shaping loop to truncate the LSBs only, and recombine the MSBs with the truncated LSBs. The resulting data contain only slightly more quantization noise than that obtained directly from an N -bit noise-shaping loop, and the hardware needed for the loop is much less elaborate.

A problem common to all $\Delta\Sigma$ DACs is the generation of idle channel tones, as covered in Chapter 3. This is particularly likely to occur in DACs, where the modulator sig-

nals are all digital and the arithmetic is purely rational. Until recently, it was widely believed that high-order single-path systems, such as those shown in Figure 10.6 or Figure 10.7, were not tonal. However, tonal behavior in such structures has clearly been demonstrated in simulation and in practice, as covered in Section 3.4.4 and 3.6.2 of Chapter 3. In a two-path structure, such as that shown in Figure 10.16 or 10.17, the correction path processes the tone generated in the signal loop along with e_1 and subtracts a quantized version of it from the 1-bit output. The tonal information contained in the output that remains after subtraction is reduced in accordance with number of quantization levels in the M -bit DAC.

It is possible to use digital dithering to prevent the generation of tones in any of these architectures. An extensive treatment of this subject can be found in Chapter 3, beginning with Section 3.8.

10.3 DESIGN EXAMPLE 1: A FIFTH-ORDER SINGLE-BIT NOISE-SHAPING LOOP

This design example will explore the use of a high-order single-bit architecture to achieve, in theory, 110 dB SNR at an oversampling ratio of 64.

10.3.1 The Noise Transfer Function

Using the design procedure of Section 4.4, a fifth-order noise transfer function with optimized zeros and a maximum out-of-band gain of 1.45 was found. It can be realized by a stable modulator (with inputs up to one-half of full scale) having sufficient noise suppression to meet the 110-dB-SNR objective. Figure 10.22 shows the pole-zero configuration of the NTF, while Figure 10.23 gives the magnitude response of the STF and NTF. The rms gain of the NTF is -91 dB in the band of interest and the maximum out-of-band gain is approximately 3 dB. Figure 10.22 also lists the actual values used, which satisfy the practical constraints discussed in the next section.

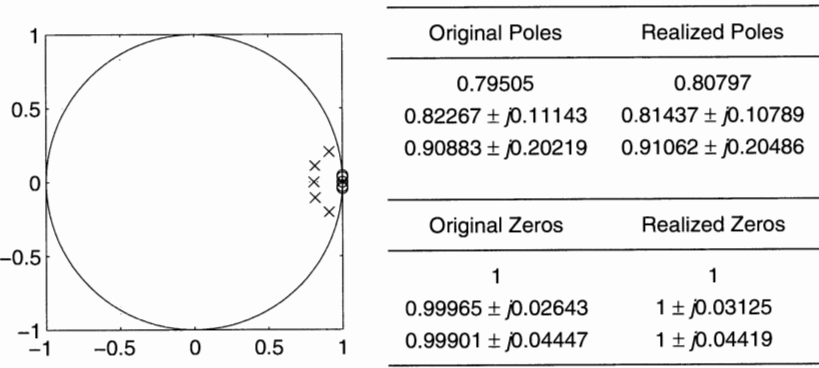


Figure 10.22 Poles and zeros for the noise transfer function of the example fifth-order modulator.

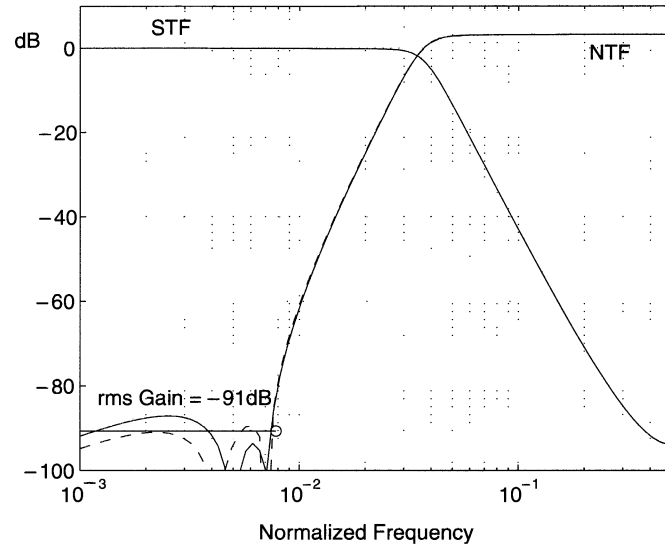


Figure 10.23 Signal and noise transfer functions for the example fifth-order modulator. The dashed line is the NTF response in the absence of errors in the zero placement caused by coefficient quantization.

10.3.2 The Modulator Structure

Figure 10.6 shows the topology of the loop filter we shall use to realize the above NTF. The advantage of this topology is that the adders used in each integrator have a full clock period to compute their outputs since each integrator is of the delaying variety. However, the double delay in the resonator loops causes the resonator transfer functions to be of the form

$$R(z) = \frac{1}{(z-1)^2 + g} \quad (10.5)$$

These have poles at

$$z = 1 \pm j\sqrt{g} \quad (10.6)$$

rather than on the unit circle. Thus, the NTF zeros need to be shifted onto the line $\text{Re}(z) = 1$ in order for the NTF to be realizable with the structure of Figure 10.6. To avoid the use of multipliers, the g coefficients need to be sums of a few powers-of-2 terms. Using a single power of 2 shifts the NTF zeros noticeably, but as Figure 10.23 shows, the modulator performance is not significantly degraded.

To implement the modulator with integer arithmetic, the tolerable round-off error in each stage needs to be calculated, and the dynamic range of the integrator outputs determined through simulation. These results then allow the number of bits required in each stage to be computed and also give the interstage scaling coefficients, which had been

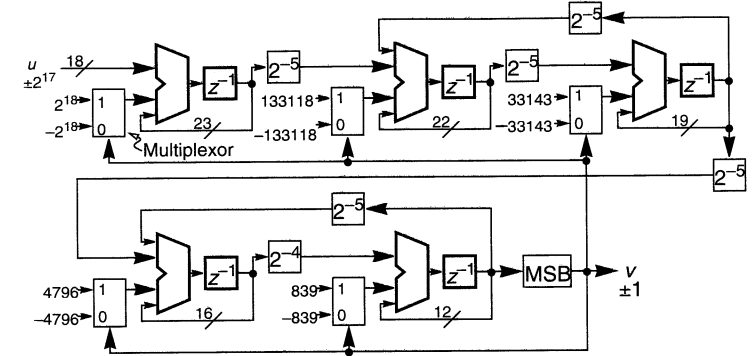


Figure 10.24 Structure of the fifth-order modulator.

omitted from Figure 10.6 for simplicity. The complete modulator, including word lengths and coefficients, is depicted in Figure 10.24. In all, the modulator requires approximately 100 bits of storage and 200 1-bit adders. This amount of hardware fits comfortably into a few square millimeters of silicon. The adder count can be further reduced by quantizing the feedback coefficients such that only the high-order bits are nonzero or by multiplexing a smaller number of adders. Figures 10.22 and 10.23 compare the realized NTF's poles and zeros and their magnitude response with those of the original. As Figure 10.23 also shows, the STF of the modulator is very flat, and thus no correction by the interpolation filter is needed.

Figure 10.25 plots the simulated SNR versus input level curve for the modulator illustrated in Figure 10.24. Note that the modulator has been scaled such that the

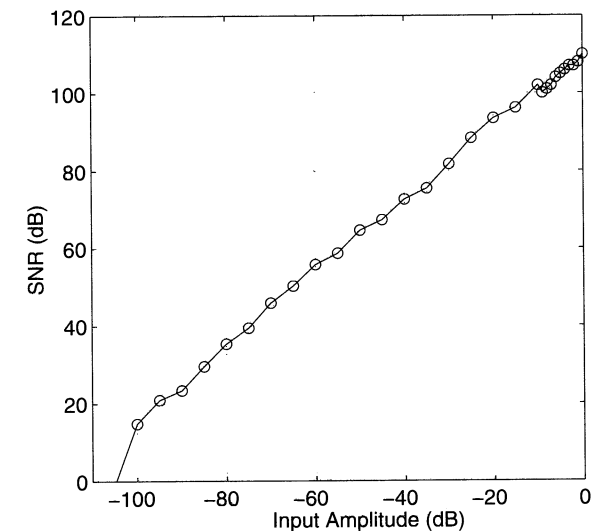


Figure 10.25 SNR versus input amplitude for the fifth-order modulator shown in Figure 10.24.

maximum possible input is one-half of full scale. The simulations show that the modulator meets the 110-dB-SNR objective.

The design of interpolation filters is described in Chapter 13, while Chapter 12 deals with the design of the analog postfilter which must process the modulator output.

10.4 DESIGN EXAMPLE 2: A THIRD-ORDER (2+1) MULTIBIT CASCADE NOISE-SHAPING LOOP

As discussed in the introduction to this chapter, a complete oversampling DAC system consists of three basic functional blocks performing interpolation, noise shaping, and analog low-pass filtering. The three blocks cannot be treated separately in the design process since their specifications affect each other and involve a trade-off between power consumption, chip area, and overall system performance. However, the design of the interpolation filters and the analog low-pass filters will be treated as separate topics from the design of the noise-shaping loops, although their implementation impacts the choice of the noise-shaping topology.

In this section, we will focus the discussion on the design and implementation of a third-order dual-quantization noise-shaping loop (Figure 10.17) and the required analog differentiator. We shall again assume that our objective is to have at least 110 dB in-band signal-to-quantization-noise ratio for an oversampling ratio of 64. The high signal-to-quantization-noise ratio ensures that the quantization noise is much less than the electronic noise introduced in the subsequent analog low-pass filter at the DAC output.

To obtain a third-order noise transfer function for the system shown in Figure 10.17, we can either place all the noise-shaping circuitry into the basic noise-shaping loop or we can split it between the basic and the correction path noise-shaping loops. Various considerations and restrictions, including the digital circuitry complexity, the analog differentiator complexity and realizability, the noise cancellation requirement, the device matching accuracy requirement, and so on, lead to the choice of a second-order basic noise-shaping loop combined with a first-order correction path noise-shaping loop. Choosing an error feedback noise-shaping topology, the complete block diagram of the third-order implementation of the dual-quantization noise-shaping system becomes that shown in Figure 10.26.

The z -transform of the signal y_1 at the output of the 1-bit DAC shown in Figure 10.26 is given by

$$Y_1 = X_1 + \frac{(1-z^{-1})^2}{1-\frac{1}{2}z^{-1}} \cdot E_1 \quad (10.7)$$

As indicated in the above equation, the noise transfer function from the basic noise-shaping loop has a pole at $z = 0.5$ in addition to the two desired zeros. The pole helps to increase the loop stability and reduces the internal signal amplitude for a given range of the input signal amplitude. This reduction in turn reduces the bit-width requirement for the internal digital adders and shift registers. The penalty is that the noise gain at low frequencies is increased by a factor of 2 compared to a second-order noise-shaping loop with a double pole at $z = 0$.

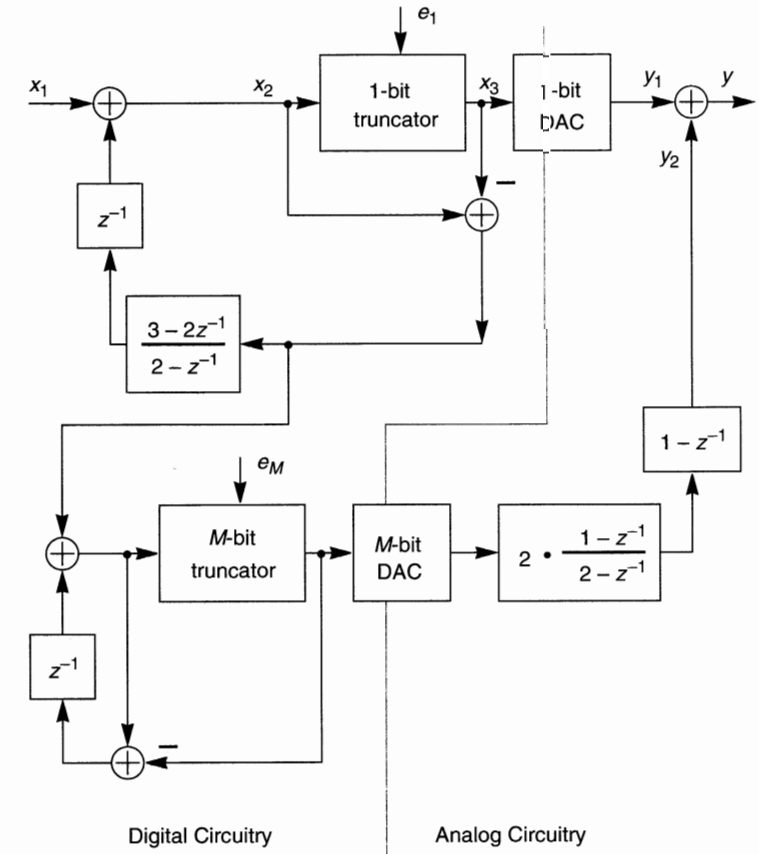


Figure 10.26 Block diagram of the completed third-order dual-quantization noise-shaping system with analog differentiator.

Under ideal conditions, the z -transform of the output signal y_2 from the correction path is given as

$$Y_2 = (1-z^{-1}) \cdot \frac{(1-z^{-1})^2}{1-\frac{1}{2}z^{-1}} \cdot E_N - \frac{(1-z^{-1})^2}{1-\frac{1}{2}z^{-1}} \cdot E_1 \quad (10.8)$$

where E_1 (E_N) is the z -transform of the single-bit (multibit) quantization error e_1 (e_N). The sum of Y_1 and Y_2 , which is the z -transform of the output signal from the dual-quantization noise-shaping system, is then given by

$$Y = X_1 + \frac{(1-z^{-1})^3}{1-\frac{1}{2}z^{-1}} \cdot E_N \quad (10.9)$$

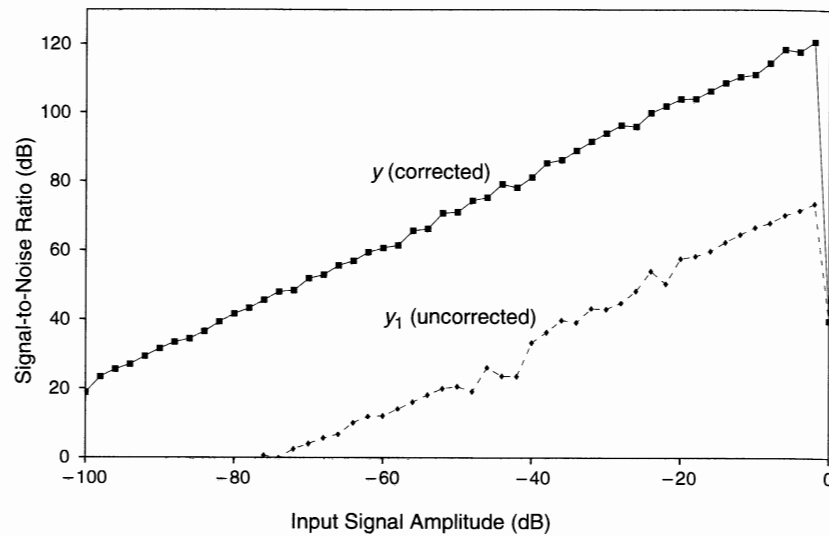


Figure 10.27 Signal-to-quantization noise ratio as a function of the input sinusoidal signal amplitude; 0 dB corresponds to the 1-bit quantizer output level.

Thus, the output $y(n)$ consists of the input signal x_1 plus the multibit quantization error e_N shaped by a third-order noise transfer function. The large error e_1 is canceled in y .

To achieve the specified signal-to-quantization-noise ratio of 110 dB, the number of bits needed in the multibit quantizer is $N = 4$. Figure 10.27 shows the simulated signal-to-quantization-noise ratio as a function of input sinusoidal signal amplitude for signals y_1

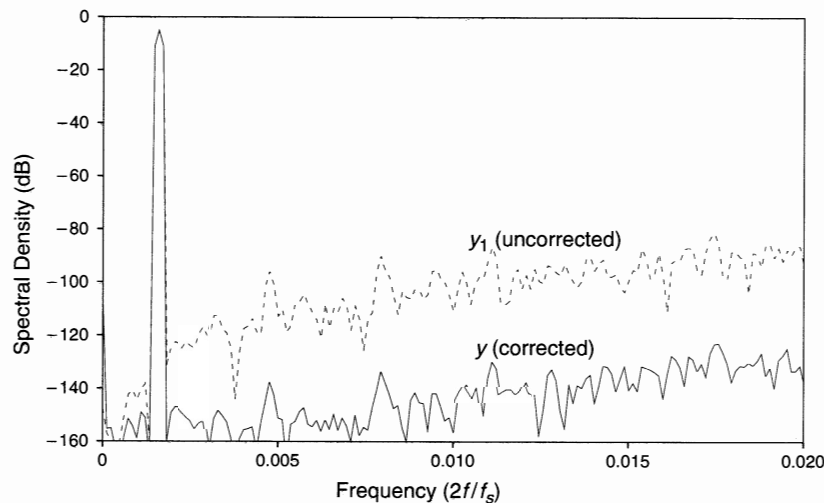


Figure 10.28 Baseband spectrum of the corrected and the uncorrected outputs for a sinusoidal input signal.

and y , the uncorrected and the corrected outputs. The baseband spectra of these two signals with a sinusoidal input are shown in Figure 10.28.

The simulations for these plots were performed with some of the analog circuitry nonidealities taken into account. Specifically, it was assumed that the 4-bit DAC had only an 8-bit linearity and that a 0.5% gain mismatch existed between the signal path and the correction path.

The maximum (minimum) output of the 4-bit quantizer was set to be the same as the maximum (minimum) output of the 1-bit quantizer. The sinusoidal test signal amplitude was normalized to the maximum output value of the quantizers.

Figure 10.29 shows a specific implementation of the third-order noise-shaping loop system. The word lengths of the various adders and shift registers were optimized using simulations with various input signals. The simulations were run for a long time, and the maximum signal amplitude at each node was then used to determine the corresponding word lengths of the adders and shift registers in the system.

Using integer numbers to represent the digital signals, the maximum input signal amplitude for the input 16-bit digital signal is $\pm 2^{15}$. To utilize the dynamic range of the

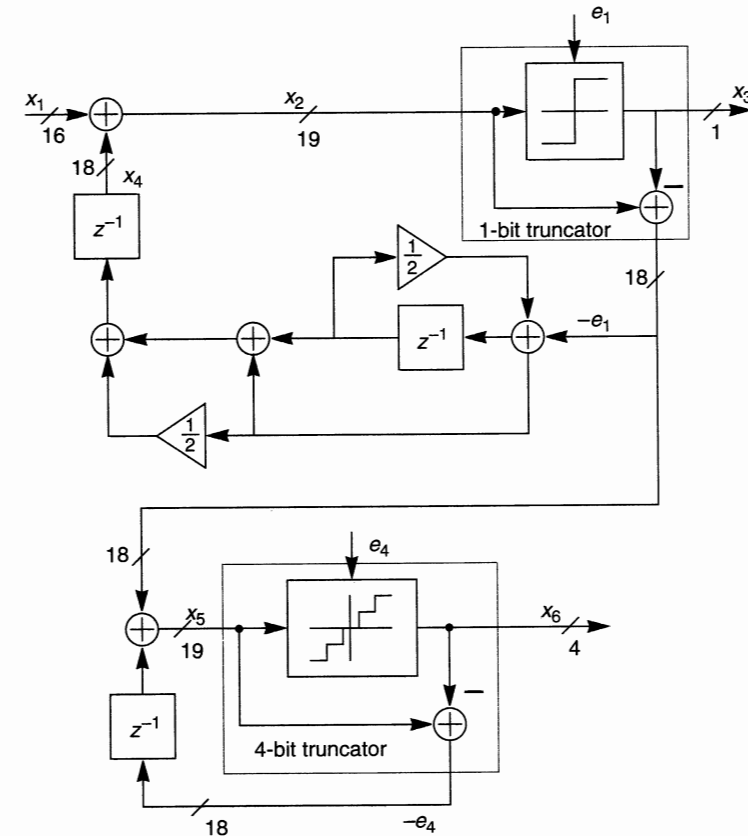


Figure 10.29 Circuit implementation of the third-order dual-quantization noise-shaping loop.

noise-shaping loop fully, the 1-bit quantizer output was set to be $\pm 2^{16}$, which is twice the maximum input signal amplitude. This choice can be understood from Figure 10.27. Under various input conditions, limited simulations indicated that the internal signal amplitudes (e_1 , x_4) are always less than $\pm 2^{17}$ in Figure 10.29. The word lengths of the adders and shift registers in the noise-shaping loop were therefore chosen to be 18 bits. A reset mechanism or digital limiting can be used to prevent quantizer overload or data overflow in the unlikely case that some internal signal amplitude exceeds $\pm 2^{17}$.

The peak signal-to-quantization-noise ratio, 119 dB, is achieved when the input signal amplitude is 2^{15} (Figure 10.27), that is, when the input signal amplitude is half of the 1-bit quantizer output level. One may argue from Figure 10.27 that the peak signal-to-quantization-noise ratio can be further increased (or the noise-shaping loop dynamic range can be better utilized) by setting the peak input level to about -3 dB relative to the 1-bit quantizer output. However, Figure 10.27 was obtained with sinusoidal input signals, which gave an optimistic estimate of the dynamic range of the noise-shaping loop. The loop dynamic range is usually less than that shown in Figure 10.27 for more general input signals.

The most area-consuming blocks in the noise-shaping loops are the adders. Figure 10.29 shows that there are five adders in the basic noise-shaping loop and two adders in the correction path. However, the word lengths of the adders associated with the quantizers can be much less than 18 bits, and they may be realized by only a few logic gates, as will be shown next. This results in a total of only five 18-bit adders needed for the whole noise-shaping system.

In Figure 10.29, if we consider the simple adder associated with each quantizer as part of the quantizer, then each quantizer has two outputs. The output x_3 of the 1-bit quantizer is simply the MSB of the signal x_2 ; its integer value is 2^{16} or -2^{16} , as shown in Figure 10.30. The other output $-e_1$ is the difference between signals x_2 and x_3 and is 18 bits wide. Using 2's complement notation, the 16 LSBs of the error signal $-e_1$ are the same as those of the signal x_2 . The two MSBs of the error signal $-e_1$ are determined by the three MSBs of the signal x_2 . The truth table is shown in Table 10.1. An implementation of this 1-bit quantizer in 2's complement notation is shown in Figure 10.31. The 4-bit quantizer functions both as a digital quantizer and as an amplitude limiter. Its maximum (minimum) output value is chosen to be the same as that of the 1-bit quantizer. The implementation is somewhat more complicated than that of the 1-bit quantizer since the output is not the same as the four MSBs of the input, due to the limiter function. However, the circuit implementation can still be very simple, as shown below.

Refer back to Figure 10.29. The output $-e_4$ is an 18-bit signal and its 13 LSBs are the same as the 13 LSBs of the signal x_5 . A 5-bit adder, instead of an 18-bit adder, can therefore be used to determine the five MSBs of the error $-e_4$. The algorithm is similar to what was discussed for $-e_1$, and an implementation is shown in Figure 10.32.

The key subsystem in the dual-quantization D/A conversion system is the analog differentiator, since it determines the effectiveness of the error cancellation. The ideal transfer function of the analog differentiator as shown in Figure 10.26 is

$$H_D(z) = (1 - z^{-1}) \cdot \frac{(1 - z^{-1})}{1 - \frac{1}{2} \cdot z^{-1}} \quad (10.10)$$

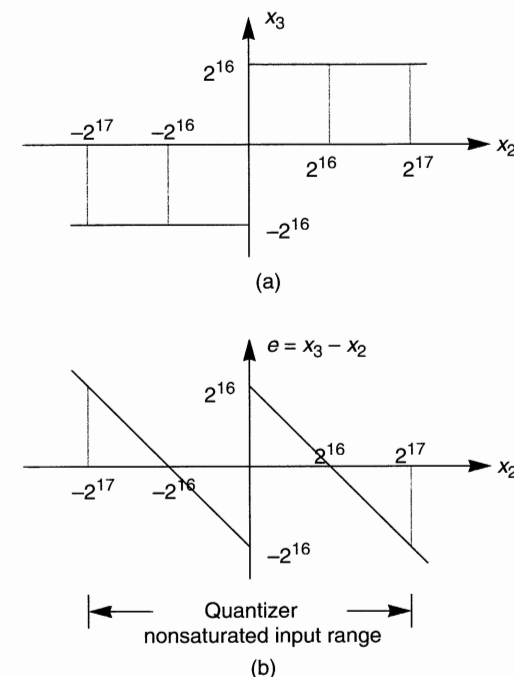


Figure 10.30 One-bit quantizer: (a) input-output relationship; (b) quantization error.

TABLE 10.1 TRUTH TABLE FOR THE TWO MSBs OF THE ERROR SIGNAL $-e_1$ IN 2'S COMPLEMENT NOTATION

$x_2[19]$	$x_2[18]$	$x_2[17]$	$-e_1[18]$	$-e_1[17]$	Notes
0	1	1	Impossible (2)		1. The truth table is obtained from the equation $-e_1 = x_2 - x_3$ with 2's complement notation.
0	1	0	0	1	
0	0	1	0	0	
0	0	0	1	1	2. Impossible since x_2 is the sum of an 18-bit signal and a 16-bit one. Hence, the three MSBs cannot equal to 011 or 100 in 2's complement notation.
1	1	1	0	0	
1	1	0	1	1	
1	0	1	1	0	
1	0	0	Impossible (2)		

It was shown in reference [10] that in order to cancel the large error e_1 with sufficient accuracy, the two transmission zeros have to be accurately realized, while pole and gain errors are much less critical.

The second-order analog differentiator can be implemented by cascading two first-order switched-capacitor differentiator stages, as shown in Figure 10.33. A single-ended version of the circuit is shown for clarity, although in practical implementation a fully

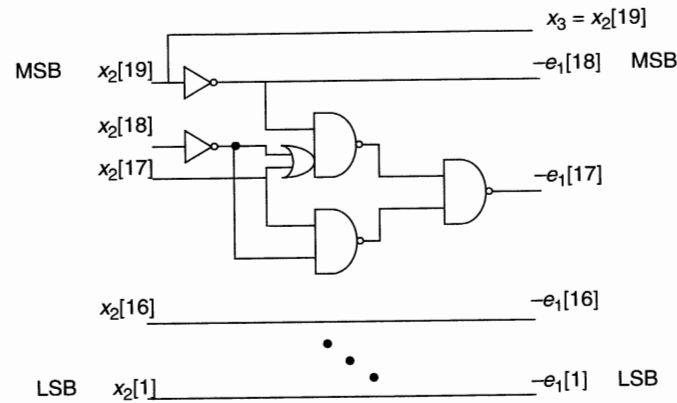


Figure 10.31 Circuit implementation of the 1-bit quantizer with its associated adder.

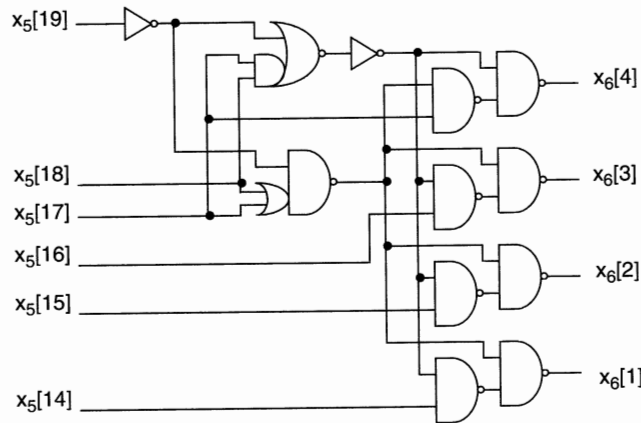


Figure 10.32 Circuit implementation of the 4-bit quantizer.

differential configuration is preferred. The transfer function from v_1 to v_2 is

$$H_1(z) = \frac{C_1}{C_3} \cdot \frac{1 - z^{-1}}{1 + C_2/C_3 + \alpha(1 + C_1/C_3 + C_2/C_3) - [C_2/C_3 + \alpha(C_1/C_3 + C_2/C_3)]z^{-1}} \quad (10.11)$$

where $\alpha = 1/A_1$, and A_1 is the op-amp dc gain. Thus, the transmission zero is realized accurately. (It will be affected by leakage in C_1 and C_4 , but that is usually negligible in a CMOS technology for reasonable clock rates and temperatures.) The finite op-amp gain results in a gain error at low frequencies, approximately equal to $1/A_1$, and an error in the pole location. Both of these errors are small and have a very small effect on the error cancellation if the op-amp gain is at least 60 dB.

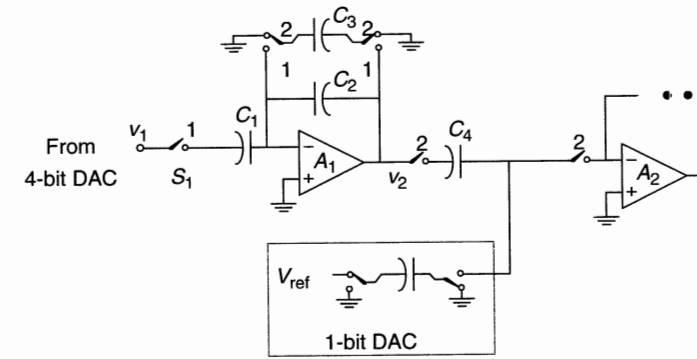


Figure 10.33 A cascade second-order analog differentiator.

The parasitic capacitances at the top and bottom plates of the capacitor C_1 do not affect the zero location. The signal-dependent portion of the charge feedthrough from the switch S_1 can be treated as a part of the nonlinearity error of the 4-bit DAC, which is not important either, as discussed in Section 10.2. The main error source affecting the transmission zero in the first differentiator is the op-amp settling error. The output signal v_2 of the op-amp A_1 is a high-pass-shaped error signal. This signal has a large sample-to-sample voltage swing. The settling error has to be small in order to reduce the effect of any nonlinear settling behavior of the op amp. For high-speed applications, this can be a major limitation for using this differentiator. For audio applications, however, this should not be a problem.

The second differentiator is realized using a single capacitor C_4 . The differentiated signal is combined with the charge from the 1-bit DAC output before it enters the next stage of the low-pass filter. Thus, the differentiation is nearly ideal if the current leakage can be neglected. Note also that any noise from the 4-bit DAC and the first-stage differentiator (including settling errors) will be first-order noise-shaped by the second differentiator stage.

10.5 CONCLUSION

In this chapter, the system-level design of $\Delta\Sigma$ DACs was discussed. It was shown that because of the different signal representations (analog vs. digital) of the various stages, the realization constraints for $\Delta\Sigma$ DACs differ from those valid for $\Delta\Sigma$ ADCs. Particular attention was paid to the various architectures realizing the noise-shaping loops used in these converters. Special configurations, such as error feedback structures and dual-truncation schemes, were introduced. Two numerical examples, one describing a practical fifth-order single-bit noise-shaping loop and the other discussing in detail the design of a third-order multibit cascade loop, were included to illustrate the actual design process.

The practical design issues for the remaining blocks (interpolation filter and analog postfilter) of the $\Delta\Sigma$ DAC system will be discussed later, in Chapters 12 and 13.

REFERENCES

- [1] N. S. Sooch et al., "18-bit stereo D/A converter with integrated digital and analog filters," 91st Convention of the Audio Engineering Society, Audio Eng. Soc., 60 East 42nd Str., New York, NY 10165, Preprint No. 3113, Oct. 1991.
- [2] P. J. Naus et al., "A CMOS stereo 16-bit D/A converter for digital audio," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 390–395, June 1987.
- [3] H. G. Musmann and W. Korte, "Generalized interpolative method for digital/analog conversion of PCM signals," U.S. Patent 4,467,316, Aug. 21, 1984.
- [4] J. C. Candy and A. N. Huynh, "Double interpolation for digital-to-analog conversion," *IEEE Trans. Commun.*, vol. COM-34, pp. 77–81, Jan. 1986.
- [5] J. C. Candy and G. C. Temes, "Oversampling methods for data conversion," in *Oversampling Delta-Sigma Data Converters*, IEEE Press, New York, 1992.
- [6] H. Schouwenaars et al., "An oversampling multi-bit CMOS DAC for digital audio with 115 dB dynamic range," *1991 IEEE Int. Solid-State Circuits Conf. Dig.*, vol. 34 pp. 72–73, Feb. 1991.
- [7] B. Leung, "Pipelined multibit oversampled D/A converters with capacitor averaging," *Analog Int. Circuits Signal Proc.*, vol. 2, pp. 139–156, April 1992.
- [8] T. Cataltepe et al., "Digitally corrected multi-bit sigma-delta data converters," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 647–650, May 1989.
- [9] M. Sarhang-Nejad and G. C. Temes, "A high-resolution multibit sigma-delta ADC with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, pp. 648–660, June 1993.
- [10] X. F. Xu et al., "The implementation of dual-truncation sigma-delta D/A converters," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 597–600, May 1992.
- [11] T. C. Leslie and B. Singh, "An improved sigma-delta modulator architecture," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 372–375, May 1990.
- [12] B. P. Brandt and B. A. Wooley, "A 50-MHz multibit sigma-delta modulator for 12-b 2-MHz A/D conversion," *IEEE J. Solid-State Circuits*, vol. 21, pp. 1746–1756, Dec. 1991.
- [13] K. Uchimura et al., "Oversampling A-to-D and D-to-A converters with multistage noise shaping modulators," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. AASP-36, pp. 1899–1905, Dec. 1988.
- [14] A. Hairapetian et al., "Multibit sigma-delta modulator with reduced sensitivity to DAC nonlinearity," *Electron. Lett.*, vol. 27, pp. 990–991, May 1991.
- [15] A. Hairapetian et al., "A dual-quantization multi-bit sigma delta A/D converter," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 437–440, May 1994.
- [16] S. R. Norsworthy et al., "A minimal multi-bit digital noise shaping architecture," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. I-5-I-8, May 1996.

Analog Circuit Design for $\Delta\Sigma$ ADCs

Brian Brandt
Paul F. Ferguson
Mike Rebeschini

11.1 INTRODUCTION

Much of the appeal of $\Delta\Sigma$ ADCs stems from their tolerance for component mismatch and circuit nonidealities. This tolerance can eliminate the need for component trimming or calibration and allows the implementation of these converters in VLSI technologies. Oversampled ADCs efficiently exchange speed for resolution and thereby exploit the high sampling rates afforded by the low parasitic capacitances and small feature sizes characteristic of VLSI technologies.

This chapter will explore the unique aspects of analog circuit design for oversampled ADCs. As will be seen, these robust converters considerably reduce the precision required of the analog circuits as compared to Nyquist-rate converters. They also simplify system integration by reducing the burden on the supporting analog circuitry. Specifically, they do not require precision sample-and-hold circuitry and they relax the performance requirements on the front-end analog antialias filter.

Section 11.2 discusses the implications of the choice of architecture on analog circuit design. It also distinguishes the critical and noncritical nodes and components of $\Delta\Sigma$ modulators and presents several general design considerations. Section 11.3 focuses on the design of key subcircuits commonly used to implement $\Delta\Sigma$ modulators. Section 11.4 describes the effects of some key circuit nonidealities, while Section 11.5 addresses important considerations for the design of the component blocks of $\Delta\Sigma$ modulators. Section 11.6 discusses some system considerations, such as dynamic range, clock jitter, and input impedance. Section 11.7 discusses the proper layout techniques for $\Delta\Sigma$ ADCs. The chapter concludes with two design examples of modulators that place different demands on the analog circuitry used for their implementation.

In an effort to limit the scope of the chapter, an emphasis has been placed on switched-capacitor implementations in CMOS VLSI technologies because of the relatively low cost of these technologies and their widespread use in digital signal processing applications. While most of the $\Delta\Sigma$ modulator implementations reported to date have employed established switched-capacitor design techniques, several continuous-time implementations have also been reported [1–4].

11.2 ARCHITECTURAL CONSIDERATIONS

Several $\Delta\Sigma$ modulator architectures have been presented in the preceding chapters of this text. These architectures can broadly be separated into two categories with respect to the requirements they place on the analog circuits used for their implementation. Single-stage modulators such as those presented in Chapters 1 and 5 are generally more tolerant of circuit nonidealities than the cascaded modulators presented in Chapters 4 and 6. For example, single-stage modulators are quite tolerant of gain errors that result from capacitor mismatch in switched-capacitor integrators. Gain variations of as much as 20% from the nominal value have only a minor impact on the performance of second-order $\Delta\Sigma$ modulators [5]. However, large integrator gain errors can lead to instability, especially in modulators of order greater than 2. Tolerance for gain error also translates into tolerance for incomplete settling of the integrator outputs if the settling process is linear.

Single-stage modulators are also quite tolerant of integrator leakage that results from the finite dc gain of nonideal operational amplifiers. While leakage reduces the attenuation of the quantization noise at low frequencies, a dc gain approximately equal to the oversampling ratio is sufficient to limit the baseband noise increase to less than 1 dB in second-order modulators [5]. In practice, larger dc gains may be needed to suppress harmonic distortion. Also, in first- and second-order modulators, large integrator leakage can result in a “dead zone” near the zero input in which small changes in the input to the modulator do not change the modulator output [6].

In contrast to single-stage modulators, the performance of cascaded modulators may be quite sensitive to circuit nonidealities. The performance of these modulators depends on the accurate matching between the analog noise shaping performed in the first stage and the digital noise shaping provided by the logic that combines the outputs of the individual stages. The combination of gain error from capacitor mismatch and leakage from finite operational amplifier dc gain prevents switched-capacitor integrators from achieving their ideal transfer function. As a result, first-order shaped quantization noise proportional to the gain error is leaked through to the combined output of a modulator composed of two or three cascaded first-order stages. In addition, *unshaped* quantization noise that is inversely proportional to the dc gain appears in the combined modulator output. As discussed in Chapters 6 (Section 6.3.2) and 8, tolerance for integrator gain error and leakage is improved significantly by employing a second-order modulator as the first stage in the cascade.

Both single-stage and cascaded modulators are sensitive to circuit nonidealities at their inputs. For example, electronic noise produced in the sampling switches and the operational amplifier of the first integrator ultimately limits the achievable dynamic range as the oversampling ratio is increased. Also, the sampling of the input to the modulator

and of the output of the feedback DAC, as well as the summing of these two signals, must be performed to the overall precision of the conversion. Offset dc voltages at the input to the first integrator appear directly in the transfer characteristic of the oversampled ADC but are only a minor concern in many signal acquisition systems. Nonlinearity in the first integrator, on the other hand, represents the most significant source of harmonic distortion in modulators employing 1-bit quantizers. As a result, operational amplifiers with large dc gains and capacitors with low-voltage coefficients may be required to implement the first integrator. Also, slewing in the output of the first integrator must be avoided unless sufficiently accurate settling (determined by the required accuracy of the overall ADC) is allowed under all conditions.

Circuit imperfections at other points in $\Delta\Sigma$ modulators are less critical than those at the input. The effects of electronic noise, offsets, and nonlinearities in the second and succeeding integrators are attenuated by the gain of the preceding integrators. As a consequence, it is often possible to reduce the size and power consumption of these integrators. The performance of $\Delta\Sigma$ modulators is also relatively insensitive to electronic noise and offsets in the comparators, since the effects of these impairments are attenuated by the same noise shaping that attenuates the large quantization noise. While comparator hysteresis as large as 10% of the modulator input range is tolerable in second-order modulators [5], it may be problematic in higher order modulators, as discussed in Section 11.5.2.

The choice of oversampling ratio has an influence on the design of the circuits used to implement a $\Delta\Sigma$ modulator. For a given baseband frequency, the use of a higher oversampling ratio generally increases the operational amplifier's bandwidth and slew rate requirements. However, these increased performance requirements are mitigated to a large degree by the reduction in the size of the sampling and load capacitances that may accompany the increase in the oversampling ratio. Thus, modulators that operate at higher oversampling ratios do not necessarily place more stringent demands on the analog circuits used for their implementation. For example, if the oversampling ratio is doubled, the size of the sampling and integrating capacitors may be reduced by a factor of 2 in order to double the bandwidth and slew rate of a single-stage amplifier. A factor-of-2 increase in thermal noise power results from reducing the sampling capacitors and increasing the amplifier bandwidth. However, because of the increased oversampling ratio, the thermal noise is distributed over twice the bandwidth, yielding the same baseband noise power. The speed of the amplifier may be increased in this fashion until the unity-gain bandwidth approaches the nondominant pole frequencies, at which point the phase margin of the amplifier degrades. The use of a larger oversampling ratio also relaxes the performance requirements on the analog antialias filter preceding the modulator.

Because of the mixed-signal nature of oversampled ADCs, the use of differential modulator topologies to reject noise from digital circuitry is especially important. The use of a fully differential configuration attenuates power supply and substrate noise, clock feedthrough, switch charge injection errors, and even-order harmonic distortion. Moreover, in some operational amplifier topologies, such as the folded cascode [7], the use of a differential configuration removes current mirrors and their associated poles from the signal path. A differential architecture also doubles the available signal range of the modulator, which is especially important as supply levels migrate from 5 to 3 V and lower. Alternatively, the increased signal range may be leveraged to allow the use of smaller sampling capacitors so that the total capacitance of a differential integrator equals that of a

single-ended implementation. The minimum allowable capacitance is, however, also limited by mismatch and/or linearity considerations. Thus, the principal area and power penalty for the use of differential circuitry results from the feedback circuitry required to set the common-mode output level of the integrators. Depending on the application, single-ended to differential conversion circuitry may also be required.

11.3 BUILDING BLOCKS

11.3.1 Input Integrator

In discussing the basic building blocks that make up a $\Delta\Sigma$ modulator, it is appropriate to discuss the input integrator first. This is not only because of the location of the input integrator as the first block in the $\Delta\Sigma$ system, but also because of the importance of the input integrator in determining the overall performance of the modulator.

The reason that the first integrator is so important to the overall performance of a $\Delta\Sigma$ ADC has to do with noise shaping. The easiest way to determine the effect of a particular circuit impairment on the performance of a $\Delta\Sigma$ modulator is to refer it back to the input of the modulator. Referring a signal back through an integrator is equivalent to dividing it by the transfer function of the integrator. If the integrator has a transfer function

$$H(z) = 1/(1 - z^{-1})$$

then the unity-gain frequency f_1 of the integrator will occur when

$$|1 - e^{-j2\pi(f_1/f_s)}| = 1$$

that is, when $f_1 = f_s/6$. Since $\Delta\Sigma$ modulators employ a reasonable amount of oversampling, usually the oversampling ratio satisfies $R = f_s/2B \geq 32$. Hence, the integrator will have a considerable amount of gain at frequencies in the baseband. Therefore, noise and distortion in the baseband due to circuit nonidealities occurring after the first integrator will be greatly attenuated when referred back through the integrator. The same reasoning applies to succeeding integrators. Baseband noise and distortion due to circuit nonidealities occurring after the second integrator are attenuated even more by virtue of being referred back through two integrators. Therefore, each succeeding integrator becomes less of a design challenge. However, enough benefit is gained from one order of noise shaping to make the first integrator the only one that requires special design consideration. Besides, if you can design the first integrator to full specifications, you know you can design the other integrators.

In a single-loop modulator, the noise and distortion performance of the modulator will be determined primarily by the noise and distortion performance of the first integrator. Single-loop modulators are relatively insensitive to integrator gain and pole errors. In a cascaded arrangement, the noise and distortion performance is most important in the first integrator of the first modulator in the cascade. Integrator gain and pole errors, which do have a significant effect on the performance of a cascaded $\Delta\Sigma$ modulator, are most important in the first integrator of the first modulator. If the order of the first modulator in the cascade is greater than 1, then gain and pole errors in the other integrators in the first modulator may be as important as errors in the first integrator, depending on the signal that is fed from the first modulator to the second modulator (Chapter 6).

In Sections 11.3.2–11.3.4 and 11.4.1–11.4.3, the effects of nonideal components on the characteristics of a switched-capacitor integrator are derived. First, a specification sheet for an SC integrator is created. Then a switch-level schematic of a fully differential SC integrator is presented, and the differential-mode and common-mode transfer functions are derived. It is then shown how such an integrator would actually be realized with MOSFET switches, capacitors, and an op-amp. A folded-cascode op-amp design is presented and analyzed. Sections 11.4.1 and 11.4.2 contain an analysis of the effect of nonideal switches, capacitors, and op-amps on the characteristics of the integrator. Thermal and flicker noise are covered in greater detail in Section 11.4.3.

11.3.2 Specifications

The schematic for a switched-capacitor integrator is shown in Figure 11.1. The ideal transfer function is given by

$$\frac{V_o}{V_i} = \frac{C_1}{C_2} \frac{z^{-1/2}}{1 - z^{-1}} \quad (11.1)$$

In reality, however, Eq. (11.1) will be insufficient to describe completely the characteristics of the integrator. Because the integrator is constructed from nonideal analog components, a complete description of the characteristics of the integrator should also take into account the noise and distortion generated in the integrator and the deviation of the transfer function from the ideal expression given in Eq. (11.1).

A data sheet is a convenient method for specifying the characteristics of a switched-capacitor integrator. Such a data sheet is shown in Table 11.1.

The mechanisms that determine the values of the parameters listed in Table 11.1 are examined in Sections 11.3.3–11.3.4, and in Sections 11.4.1–11.4.3.

11.3.3 Fully Differential SC Integrator

A switch-level diagram of a fully differential switched-capacitor integrator is shown in Figure 11.2. Fully differential circuitry has superior power supply noise rejection, as compared to a single-ended design, and also provides twice the output swing for a given supply voltage. In addition, the symmetry of a fully differential circuit provides for cancellation of even-order distortion components, regardless of their cause. For these reasons, fully differential circuits are recommended for high-performance switched-capacitor designs.

Analyzing fully-differential circuits directly can be both complicated and redundant. The best way to analyze these circuits is to take advantage of the symmetry of the circuit and use Bartlett's bisection theorem (see, e.g., G. Temes and J. LaPatra, *Introduction to*

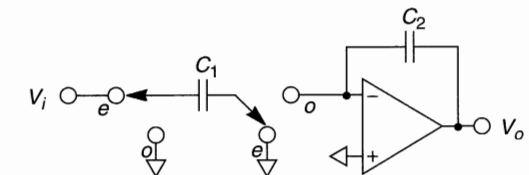


Figure 11.1 Single-ended switched-capacitor integrator.

TABLE 11.1 DATA SHEET FOR SWITCHED-CAPACITOR INTEGRATOR

Parameter	Symbol	Definition	Value
Maximum clock frequency	f_s, max	Maximum frequency the integrator can be clocked at and still meet specifications	
Input impedance	Z_i	See Section 11.6.3	
Gain error	α	Actual $H(z) = \frac{C_1}{C_2} \frac{(1-\alpha)z^{-1/2}}{1-(1-\beta)z^{-1}}$	
Pole error	β	See above	
Input referred thermal noise voltage	e_T		
Input referred flicker noise voltage	$e_{1/f}$		
Maximum output signal swing	V_o, max		
Signal-to-distortion ratio	S/D	Signal-to-distortion ratio measured at a specified signal output level; may be frequency dependent	

Circuit Synthesis and Design, McGraw-Hill, New York, 1977, p. 235) to create single-ended circuits for the differential- and common-mode responses. The following analysis will take into account the input signal v_i and the charge injected into the summing junction per clock cycle, Q_{inj} .

First, the positive and negative inputs are defined in terms of common-mode and differential-mode components:

$$V_i^+ = V_{\text{cm}} + V_d \quad V_i^- = V_{\text{cm}} - V_d \quad (11.2)$$

$$Q_{\text{inj}}^+ = Q_{\text{inj}}^{\text{cm}} + Q_{\text{inj}}^d \quad Q_{\text{inj}}^- = Q_{\text{inj}}^{\text{cm}} - Q_{\text{inj}}^d \quad (11.3)$$

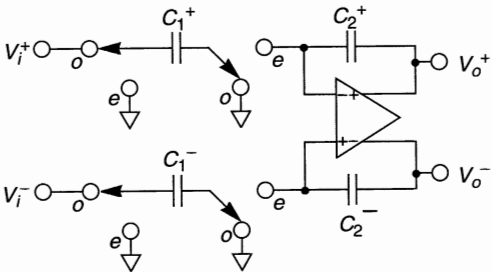


Figure 11.2 Fully differential switched-capacitor integrator.

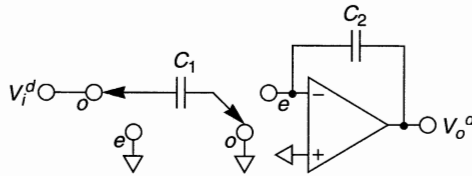


Figure 11.3 Differential-mode half-circuit.

From Eqs. (11.2) and (11.3), the common-mode and differential-mode components are given by

$$V_{\text{cm}} = \frac{1}{2}(V_i^+ + V_i^-) \quad V_d = \frac{1}{2}(V_i^+ - V_i^-) \quad (11.4)$$

$$Q_{\text{inj}}^{\text{cm}} = \frac{1}{2}(Q_{\text{inj}}^+ + Q_{\text{inj}}^-) \quad Q_{\text{inj}}^d = \frac{1}{2}(Q_{\text{inj}}^+ - Q_{\text{inj}}^-) \quad (11.5)$$

The schematic for the differential-mode circuit is shown in Figure 11.3. In the differential-mode circuit, the op-amp input terminal is held at a fixed voltage. The response to the input voltage is given by

$$V_o^d = V_i^d \frac{C_1}{C_2} \frac{1}{1-z^{-1}} \quad (11.6)$$

The response to Q_{inj}^d (differential charge injected from switches per cycle) is given by

$$V_o^d = -Q_{\text{inj}}^d \frac{1}{C_2} \frac{1}{1-z^{-1}} \quad (11.7)$$

Note that Q_{inj}^d represents one-half of the difference between the switch charge injection on the positive and negative halves of the fully differential circuit. This is one of the major advantages of fully differential circuitry: If everything matches, the switch charge injection will not show up in your signal, regardless of the charge cancellation scheme. This is only true if the switch charge injection is independent of the input signal. Signal-dependent charge injection will give rise to odd-order distortion even in a perfectly matched fully differential configuration.

The schematic for the common-mode circuit is shown in Figure 11.4. Here, V_{AG} is the analog ground voltage.¹ The common-mode output is held to V_{ref} by the common-mode feedback circuit. The response to the input signal is given by

$$V_{\text{cm}} = \left(2V_{\text{AG}} - V_i^{\text{cm}} z^{-1/2} \right) \frac{C_1/C_2}{1-z^{-1} + C_1/C_2} \quad (11.8)$$

The response to the common-mode channel charge injected per cycle, $Q_{\text{inj}}^{\text{cm}}$, is given by

$$V_{\text{cm}} = \frac{Q_{\text{inj}}^{\text{cm}}/C_2}{1-z^{-1} + C_1/C_2} \quad (11.9)$$

1. The circuits discussed in this section are all assumed to operate between 0 V and V_{DD} (5 V) supply voltages. The analog ground voltage V_{AG} is in between the voltages. It is denoted by an inverted triangle to distinguish it from $V_{ss} = 0$, which is denoted by the conventional ground symbol.

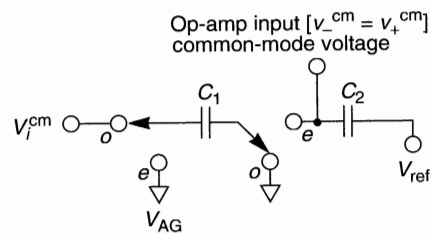


Figure 11.4 Common-mode half-circuit.

Thus, the common-mode charge injection does not cause an offset in the signal, but instead changes the steady-state bias voltage of the op-amp input terminals.

Note that V_{AG} and V_{ref} do not have to be equal. The common-mode output voltage of a fully differential amplifier can be different from the common-mode input voltage.

Having derived the differential-mode and common-mode transfer functions of the integrator from switch-level diagrams, the next step is to present a more detailed schematic for the switched-capacitor integrator. Figure 11.5 is a schematic of a fully differential switched-capacitor integrator that shows the implementation of the switches at a transistor level, along with the waveforms used to clock the switches. The bottom plate of the capacitors is designated by a curved line. This distinction is important because integrated capacitors are not generally symmetrical, and there is a larger parasitic capacitance to the substrate from the bottom plate than from the top plate. The capacitors should be connected such that the bottom plate is driven either directly or through a switch by a volt-

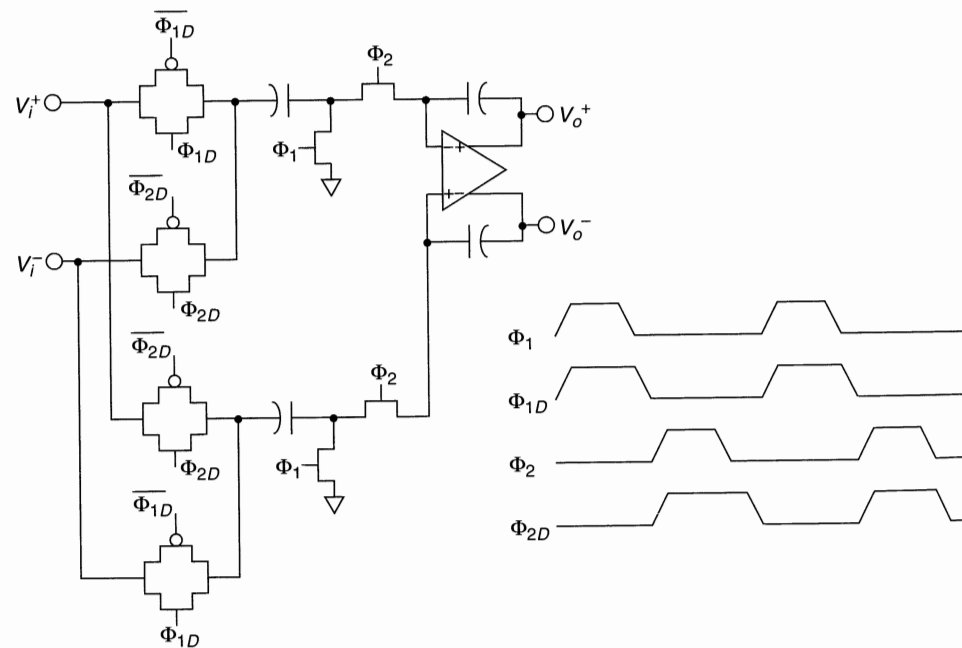


Figure 11.5 Detailed schematic of a fully differential switched-capacitor integrator.

age source or the output of an op-amp. This arrangement causes the parasitic capacitances to have the least effect on the operation of the circuit. Also, substrate noise coupling is reduced by this arrangement.

The switches are implemented with MOSFETs. In the arrangement shown in Figure 11.5, only N -channel devices are used for the summing junction switches, while full CMOS transmission gates are used for switches that are connected to a signal source or an op-amp output. In any switching scheme, it is desirable to minimize the number and sizes of the switch devices. This will minimize parasitics such as junction capacitances to the well and substrate and channel charge that degrade circuit performance. This would indicate the use of minimum-gate-length N -channel devices as switches whenever possible, since this would yield the minimum device size for a given on resistance. Switches that are coupled to the summing junction have one side connected to analog ground or to the summing junction of the op-amp. With the proper switching arrangement, the op-amp summing junction will also be at analog ground, independent of the input signal. By choosing analog ground to be a low enough voltage, it is only necessary to use N -channel devices for the summing junction switches. This is not the case for the switches coupled to the signal input or the output of op-amps. Because these switches must conduct over a wide range of voltages, it is necessary to use a full CMOS transmission gate. At lower supply voltages, even a full CMOS transmission gate may not be turned on over the full signal range. In this case, it may be necessary to make the device in the well a switched-tub switch, where the well of the device is connected to the source during the “on” phase (Figure 11.6). Further improvement may be achieved by charging up a capacitor to the supply voltage during the “off” phase and connecting the capacitor between the gate and source of the device during the “on” phase (Figure 11.7). This will result in a relatively constant gate-to-source voltage and will greatly extend the operating region of the device. It will also make the charge injection nearly signal independent.

The clocking scheme used in the circuit shown in Figure 11.5 uses a basic two-phase nonoverlapping clock (Φ_1 and Φ_2), along with the additional clocks Φ_{1D} and Φ_{2D} . These clocks are identical to Φ_1 and Φ_2 , except for a delay in the falling edge. The summing junction switches are clocked by Φ_1 and Φ_2 , while the switches that operate over the signal range are clocked by Φ_{1D} and Φ_{2D} (or their inversions if P -channel devices are used). This will cause the switches coupled to the summing junction to turn off slightly before the signal conducting switches. Once the summing junction switches have turned off, there is no

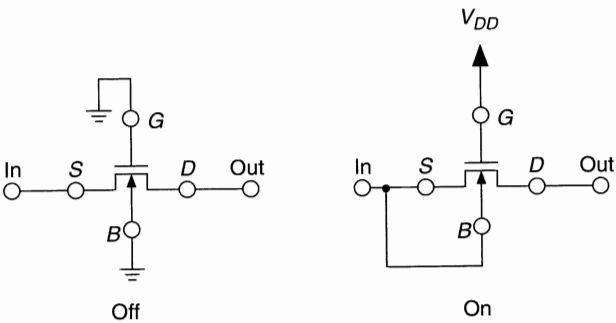


Figure 11.6 N -channel switch linearization through well biasing.

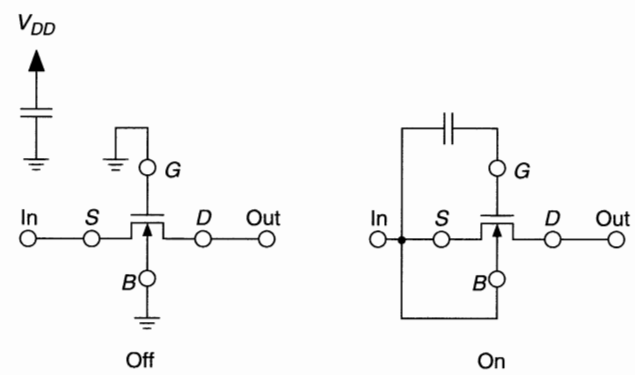


Figure 11.7 *N*-channel switch linearization through well biasing and constant gate-to-source drive.

conductive path to the top plate of the switched capacitor. The total charge on this node cannot change when the summing junction switches are open. Therefore, once the summing junction switches have turned off, the circuit may be disturbed by charge injection from other switches or interference of any kind without affecting the signal transmission. When the disturbances subside and the switches are closed for the next phase, these disturbances will not have affected the state of the circuit.

With this clocking arrangement, the charge injection from the signal conducting switches will (to a first approximation) not affect the circuit. The charge injected from the summing junction switches will be signal independent (to a first approximation), since both the source and drain of a summing junction switch will always be at analog ground when turn-off occurs. Therefore, charge injected from the summing junction switches will cause only a dc offset, not harmonic distortion. Offset is far less of a concern in $\Delta\Sigma$ modulators than harmonic distortion. Also, because of the fully differential architecture, the offset will be determined, not by the charge injection, but by the difference in charge injection between the negative and positive halves of the circuit.

It was mentioned earlier that the charge injection due to the summing junction switches is signal independent to a first-order approximation. A second-order effect that may cause distortion is that, even though the total channel charge of a turned-on summing junction switch is signal independent, the manner in which the charge divides when the switch turns off depends on the impedance on either side of the switch. Since the impedance of a conducting CMOS transmission gate is signal dependent, charge injection from summing junction switches can still give rise to some harmonic distortion. One possible solution to this problem is to place a shunt capacitor from the switched-capacitor side of the transmission gate to analog ground. If the RC time constant of the transmission gate and the shunt capacitor is much longer than the fall time of the clock that turns off the summing junction switch, then the impedance seen by the switch as it turns off will be determined by the capacitance of the shunt capacitor, which is voltage invariant, and no distortion will result. Another solution is to place a linear resistor (polysilicon, for example) in series with the CMOS transmission gate. The impedance of the series combination of a CMOS transmission gate and a resistor will exhibit a lower relative voltage coefficient than the impedance of a CMOS transmission gate alone.

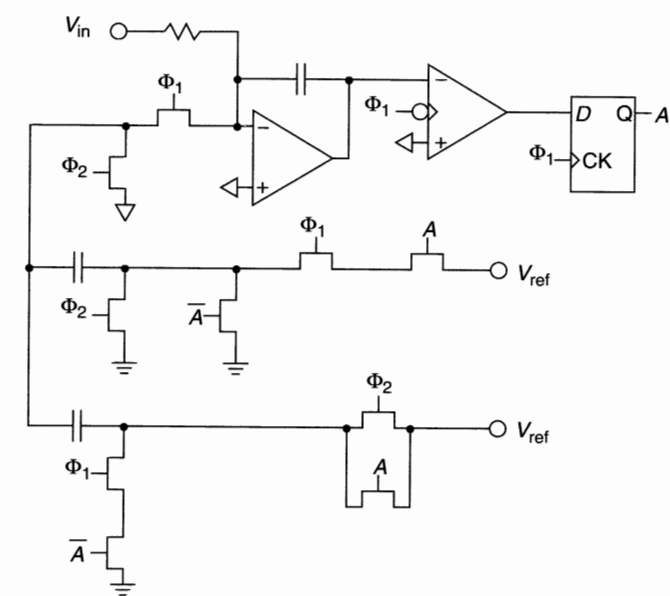


Figure 11.8 $\Delta\Sigma$ modulator with switched input capacitor replaced by resistor.

As an alternative, the input switched capacitor of the $\Delta\Sigma$ modulator can be replaced by its equivalent valued resistor, as is shown in Figure 11.8. The impedance seen by the summing junction switches is now independent of the signal. If the resistor is external, then signals that exceed the supply rails of the chip can be converted.

11.3.4 Op-Amp

Having derived a fully differential switched-capacitor integrator architecture consisting of MOSFET switches, integrated capacitors, and an op-amp, the next step is to discuss the design of the fully differential op-amp. Since the SNR of any given order modulator can be increased by increasing the oversampling ratio, speed is of primary interest. Another motivation for designing a fast op-amp is that in a switched-capacitor implementation, if the op-amp is fast enough to allow the circuit to settle completely, the exact nature of the settling is unimportant. Thus, if the amplifier is fast enough, slewing during the settling time and voltage-dependent settling will not impact the overall circuit performance. The amplifier should have a fairly linear open-loop transfer function and possess a reasonable amount of low-frequency open-loop gain, so the distortion of the amplifier will be reduced even further when placed in a closed-loop configuration. The op-amp will be driving capacitive loads, the size of which will be determined by the level of in-band thermal noise required, so a high output impedance is acceptable.

Given these design criteria, a good choice for the op-amp in the switched-capacitor integrators of a $\Delta\Sigma$ modulator is a folded-cascode op-amp (Figure 11.9). The cascode arrangement yields a high-speed op-amp that has the low-frequency open-loop gain of a two-stage amplifier. At frequencies above the dominant pole frequency, the sources of P_3 and P_4 are at virtual ground, eliminating Miller multiplication of the gate drain

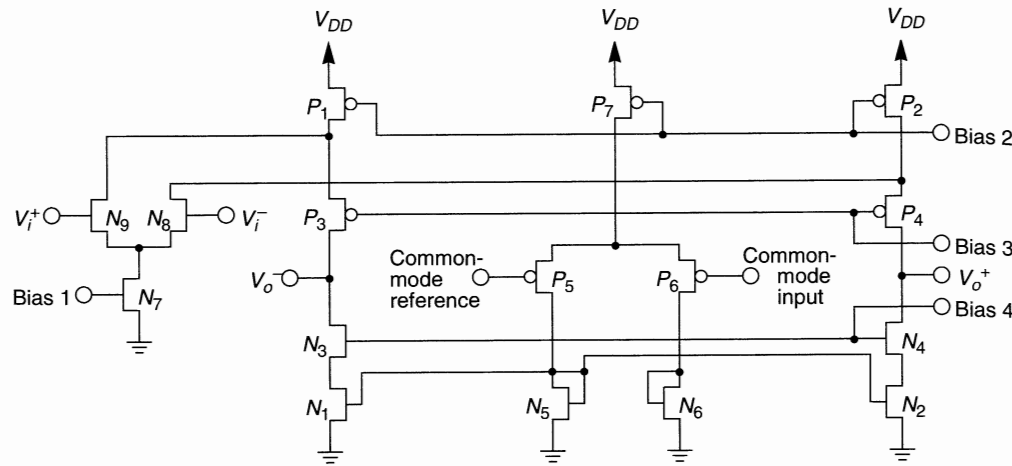


Figure 11.9 Fully differential folded-cascode op-amp.

capacitances of N_8 and N_9 . The folded-cascode circuit has a greater output swing than a direct stack of transistors.

The output voltage swing of the op-amp is determined by the cascode P devices, P_3 and P_4 , in the positive direction, and by the cascode N devices, N_3 and N_4 , in the negative direction. For proper operation, all transistors should be operating in the saturation region. Since the voltage at the sources of the cascode devices is held constant, they will come out of saturation when their drain voltage changes such that the drain-to-source voltage falls below $V_{DS,sat}$, the minimum drain-to-source voltage required for a device to remain in saturation. From simple MOS device theory

$$V_{DS,sat} = \sqrt{\frac{2I_D L}{\mu W C_0}} \quad (11.10)$$

Two possible biasing schemes for cascode devices are shown in Figure 11.10. To obtain the greatest output swing, the drain of the current source device, 3, should be biased at the minimum voltage required for it to remain in saturation. If the bias is generated from two diode-connected devices in series, then the drain of 3 is biased at $V_{DS,sat} + V_T$. Deriving

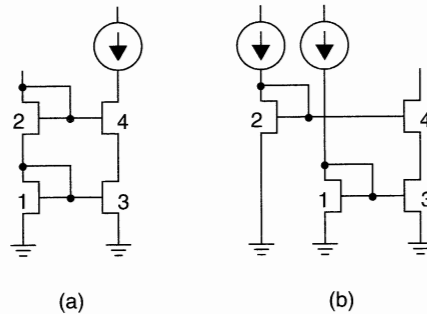


Figure 11.10 Current biasing schemes for folded-cascode op-amps: (a) cascode current mirror; (b) current mirror with improved swing.

the bias voltages from two separate diode-connected devices allows the drain of 3 to be biased at $V_{DS,sat}$. Device 2 is sized empirically to obtain the correct bias voltage.

Referring to Figure 11.9, the important small-signal parameters of the op-amp can be derived. The output impedance is given by

$$r_0 = g_{mP3} r_{dsP3} (r_{dsN9} \parallel r_{dsP1}) \parallel g_{mN3} r_{dsN3} r_{dsN1} \quad (11.11)$$

The dc voltage gain is given by

$$A = -g_{mN9} r_0 \quad (11.12)$$

The cascode op-amp is compensated by its load capacitance. The load capacitance will create a dominant pole at a frequency

$$\omega_{p1} = \frac{1}{r_0 C_L} \quad (11.13)$$

If a single pole response is assumed, then the unity-gain bandwidth of the amplifier is

$$\omega_1 = \frac{g_{mN9}}{C_L} \quad (11.14)$$

An important point, which is often overlooked, is that the source of cascode device P_3 is a virtual ground only at frequencies past the dominant-pole frequency [8]. A common-gate connected device has an input impedance that is a factor $g_m r_{ds}$ lower than its output impedance. This can still be a fairly high impedance if the output impedance is high enough. At frequencies above the dominant-pole frequency, the output impedance is reduced by the presence of load capacitor C_L . At dc, however, the output impedance is quite high.

The dc gain from the gate of N_9 to the source of P_3 is given by

$$A_1 = -g_{mN9} \left((r_{dsN9} \parallel r_{dsP1}) \parallel \frac{r_{dsN1} g_{mN3} r_{dsN3}}{g_{mP3} r_{dsP3}} \right) \quad (11.15)$$

The gain from the source of P_3 to the output is

$$A_2 = g_{mP3} r_{dsP3} \quad (11.16)$$

Since the source of P_3 is not a virtual ground at low frequency, the parasitic gate-to-drain capacitance (C_{gd}) of an input MOSFET, N_8 or N_9 , shows up as an equivalent fixed-feed-back capacitor around the op-amp of value C_{gd}/A_2 . One method for negating this effect is to place neutralizing capacitors of value C_{gd} from gate N_9 to drain N_8 and from gate N_8 to drain N_9 . Another solution is to place additional N -channel common-gate devices between the drains of the N -channel input devices and the sources of the P -channel cascode devices.

For greatest speed, N -channel devices were chosen for the input pair because of their higher mobility. Since the unity-gain bandwidth of the op-amp is determined by the ratio of the transconductance of an input device to the load capacitance, it would seem that the op-amp could be made arbitrarily fast by increasing the width and bias current of the input devices. However, to maintain stability, all nondominant poles must occur at frequencies higher than the unity-gain bandwidth of the op-amp. Therefore, the speed of the op-amp is limited by the location of the first nondominant pole. For the op-amp shown in Figure 11.9, this pole is given by g_{mP3}/C_{sP3} , the ratio of the transconductance of the cascode

P -channel device to the total capacitance on its source. Besides the parasitic diffusion capacitance on the source node, there is an intrinsic device gate-to-source capacitance. If the capacitance at the source of the device is dominated by the intrinsic gate-to-source capacitance, then using the first-order model for a saturated MOSFET gives the following expression for the nondominant pole:

$$\omega_{p2} \approx \frac{3g_m}{2C_0WL} = \sqrt{\frac{9I_D\mu}{2C_0WL^3}} \quad (11.17)$$

where I_D is the drain current, μ is the hole mobility, and W and L are the device width and length, respectively. Assuming that the minimum allowable device length is used, the non-dominant-pole frequency can be increased by increasing the current density through the device. This is done at the expense of increasing $V_{DS,sat}$, which in turn reduces the output swing of the op-amp. Again from first-order MOS theory, the nondominant pole can be expressed in terms of $V_{DS,sat}$ as

$$\omega_p = \frac{3\mu V_{DS,sat}}{2L^2} \quad (11.18)$$

To perform a noise analysis of the switched-capacitor integrator, it is necessary to have an expression for the input-referred noise of the op-amp. The rms noise voltage of each device will be represented by e_n . For this analysis, it is not necessary to separate e_n into flicker noise, wide-band thermal noise, and dc offset. The first step is to determine which devices in Figure 11.9 contribute to the noise of the op-amp. None of the devices in the common-mode amplifier contribute to the noise of the op-amp, because the output signal is taken differentially. Similarly, current source device N_7 does not contribute, because its noise also shows up as a common-mode signal. Cascode devices N_3 , N_4 , P_3 , and P_4 do not contribute to the input-referred noise of the amplifier either, because the large impedance in the source leg of these devices greatly reduces the gain from the gates of these devices to the output of the amplifier. The input-referred noise contribution of the remaining devices is derived by multiplying the noise power by the square of the ratio of that device's transconductance to the input device's transconductance.

The expression for the input-referred noise is therefore

$$e_n^2 = 2 \left[e_{nN9}^2 + e_{nP1}^2 \left(\frac{g_{mP1}}{g_{mN9}} \right)^2 + e_{nN1}^2 \left(\frac{g_{mN1}}{g_{mN9}} \right)^2 \right] \quad (11.19)$$

The factor of 2 results from the fact that the fully differential circuit consists of two matched halves. The noise of the two half-circuits is uncorrelated, so the total noise power will be twice the noise power of one of the half-circuits.

The transfer function for a fully differential amplifier, $v_{o+} - v_{o-} = A(v_{i+} - v_{i-})$, relates the differential input voltage to the differential output voltage but says nothing about the absolute values of the output voltages. For practical reasons, in actual circuits, the outputs need to be constrained to be centered about some voltage. This function is accomplished by the common-mode feedback circuitry. The performance requirements on the common-mode feedback circuitry are not nearly as stringent as for the main op-amp, because the signal of interest is the difference between the main op-amp outputs. If a single-ended output is obtained from a fully differential circuit by using only one of the fully differential outputs, then the degradation in the power supply rejection and harmonic

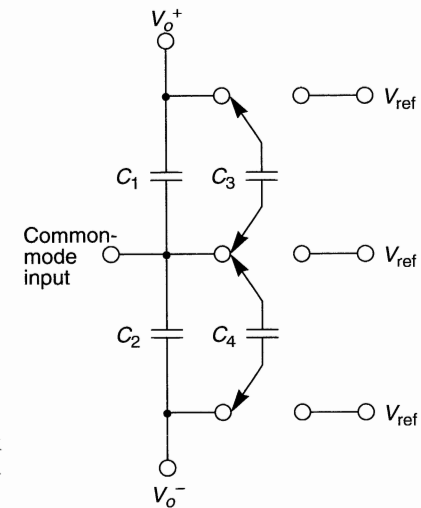


Figure 11.11 Common-mode feedback scheme for a fully differential folded-cascode op-amp.

distortion performance compared to the fully differential output case will be determined by the performance of the common-mode feedback circuitry of the final stage.

The common-mode feedback circuitry in Figure 11.9 changes the common-mode output voltage by controlling the gates of N_1 and N_2 . The common-mode op-amp compares the average of the op-amp outputs to the desired centering point. If the op-amp outputs are too high, more of I_{P7} will flow through P_5 rather than P_6 . This will increase the gate voltage of N_1 and N_2 , pulling the outputs down. The average of v_{o+} and v_{o-} may be derived from a resistive divider, with switched capacitors functioning as resistors, as shown in Figure 11.11. The fixed capacitors pass high frequencies and perform a hold function [9].

Another common-mode feedback circuit, which does not require a common-mode op-amp, is shown in Figure 11.12 [10]. Capacitors C_1 and C_2 perform the holding function and provide high-frequency feedback as before. Switched capacitors C_3 and

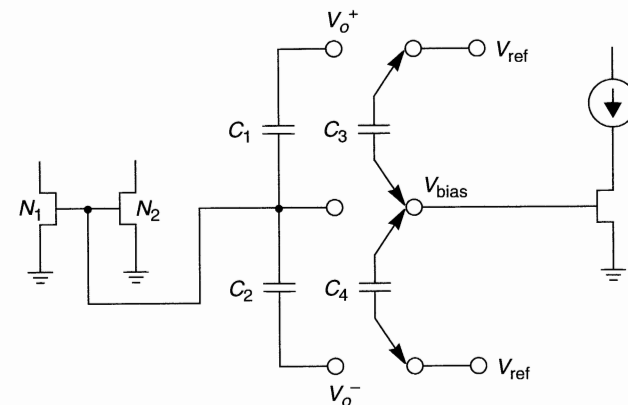


Figure 11.12 Common-mode feedback scheme that does not require a common-mode op-amp.

C_4 function as the resistive divider. V_{bias} is the bias voltage that, when applied to the gates of N_1 and N_2 (see Figure 11.9), places the outputs of the op-amp in the active operating region. If v_{o+} and v_{o-} are not centered about V_{ref} , then the switched capacitors will transfer charge onto the gates of N_1 and N_2 to provide the correction. Note that capacitors from either op-amp output to a fixed voltage (such as ground) act as compensation capacitors for both the differential- and common-mode circuits, but capacitors connected between the op-amp outputs (such as C_1 , C_2 in Figure 11.12) will not act as compensation for the common-mode circuit.

4 CIRCUIT NONIDEALITIES

11.4.1 Effects of Component Nonidealities on the Integrator Performance

A fully differential switched-capacitor integrator implemented with integrated capacitors, MOSFET switches, and an op-amp was presented in Section 11.3.3. In Section 11.3.4, the op-amp and the associated common-mode feedback circuitry were analyzed. In this section, the effects of nonidealities on the performance of the integrator will be analyzed, and the implications of using a nonideal integrator as the first integrator in a $\Delta\Sigma$ modulator are discussed. Referring back to the data sheet presented in Section 11.3.2, the factors determining the maximum operating frequency of the circuit will be discussed. The effects of nonideal components on the transfer function of the integrator, expressed in terms of gain and pole error, are analyzed. Expressions for the input-referred thermal noise are derived. The effects of nonlinear components on the performance of the integrator are discussed in Section 11.4.2. Thermal and flicker noise are discussed in more detail in Section 11.4.3, and the input impedance of the modulator is discussed in Section 11.6.3.

As mentioned previously, the maximum speed at which the integrator can be clocked is of particular interest in the design of $\Delta\Sigma$ modulators, because of their oversampled nature and because of the dramatic reduction of in-band quantization noise achieved with increased oversampling ratios. Switched-capacitor circuits operate on the assumption that complete settling occurs during each time slot. Therefore, unless the signal is used as a continuous-time signal, the manner in which the circuit settles is unimportant. Since the output of an ADC is digital, the only place the continuous-time properties of the signal are important is at the input sampler. The only time the value of the voltages at the integrator outputs matters is at the end of each time slot.

The circuit has to settle completely in one time slot. In the clocking scheme shown in Figure 11.5, the length of a time slot is equal to one-half of a clock period minus the delays put in for nonoverlap and charge injection reduction purposes. There are basically two factors that determine the settling time of the integrator: the RC time constants of the MOSFET switches in series with the switched capacitors and the response time of the operational amplifier. The response time of the operational amplifier is determined by the slew rate, the unity-gain bandwidth, and the configuration in which it is placed. A very important distinction needs to be made between settling that is signal independent (linear settling) and settling that is signal dependent (nonlinear settling). Nonlinear settling may be due to the op-amp slewing, or the fact that the characteristics of the MOSFET devices

will be operating-point dependent (cf. the discussions in Section 11.8.1). If the integrator does not settle fully but the settling process is linear, then the resultant error shows up as a gain error in the integrator (to be discussed in more detail later). If, however, the settling process is nonlinear, the resultant error will show up as distortion (also to be discussed in more detail later). Of the two factors, nonlinear settling is by far the more serious problem. A linear settling error has very little effect on the performance of a single-loop modulator, and a linear settling to within 0.1% is more than adequate for most cascaded modulators. A nonlinear settling error, on the other hand, will degrade the performance of both single-loop and cascaded modulators. If the input capacitor of the modulator does not charge completely and the degree of charging is signal dependent, then the effect is the same as if the input capacitor were nonlinear. The error will show up as harmonic distortion and is not affected by the oversampling ratio.

The most straightforward solution for eliminating the effects of nonlinear settling is to make sure the circuit settles completely enough so that no matter what the signal dependence may be, the dynamic range requirements are still satisfied. For example, if the circuit settles to within 0.001% of final value, then a 100-dB dynamic range is guaranteed regardless of the nature of the settling. Settling problems due to the RC time constant of the MOS switch and the switched capacitor can usually be solved simply by increasing the width of the switch, the limitation being that as the switch becomes larger in comparison to the switched capacitor, the effects due to channel charge injection and the parasitic junction capacitances also increase. The settling of the op-amp is a bit more involved since it depends on several factors: slew rate, unity-gain bandwidth, and gain configuration. The slew rate of the folded-cascode amplifier is determined by the ratio of the available output current to the load capacitance, so it can be increased by increasing the bias current through the amplifier. Unity-gain bandwidth can be improved at the expense of output swing, as pointed out in connection with Eq. (11.18). The configuration of the op-amp is also important in determining the charging time of the circuit. If the op-amp is assumed to have a single dominant pole and is not slewing, then the response of the integrator to a new input sample will be exponential with a time constant given by [11]

$$\tau \approx \frac{1}{\omega_1} \left(1 + \frac{\sum_m C_{im}}{\sum_m C_{fn}} \right) \quad (11.20)$$

where ω_1 is the unity-gain bandwidth, C_{im} is the m th input capacitor, and C_{fn} is the n th feedback capacitor. When using Eq. (11.20), it should be kept in mind that “input capacitor” also includes parasitic capacitance from the op-amp input to ac ground, such as the gate-to-source capacitance of the amplifier input device. From the above discussion, it can be seen that for the fastest op-amp settling the integrator gain (i.e., the ratio of switched input capacitor to integrating capacitor) should be made as small as possible. There is a trade-off here, because the smaller the gain of the first integrator, the greater the effect that noise sources following the first integrator will have on the overall modulator performance.

In the discussion of gain and pole errors that follows, it is helpful to derive first the transfer functions for a switched-capacitor integrator when effects such as nonzero switch

resistance, finite op-amp bandwidth, and finite op-amp gain are taken into account. The equations are derived for the single-ended integrator shown in Figure 11.1, but can easily be extended to the fully differential case, since the single-ended integrator is the differential-mode half-circuit of the fully differential integrator.

The integrator transfer function will differ from the ideal because of nonzero switch resistance, finite op-amp bandwidth, and finite op-amp gain. Considering these effects separately gives the following transfer functions. For finite op-amp gain A ,

$$H(z) = \frac{(C_1/C_2)z^{-1/2}(1 - 1/A - C_1/(AC_2))}{1 - (1 - C_1/(AC_2))z^{-1}} \quad (11.21)$$

For finite op-amp bandwidth B (in hertz)

$$H(z) = \frac{(C_1/C_2)z^{-1/2}[(1 - \varepsilon) + z^{-1}\varepsilon C_2/(C_1 + C_2)]}{1 - z^{-1}} \quad (11.22)$$

where $\varepsilon = e^{-\pi BT_s}$. For nonzero switch resistance R_{on}

$$H(z) = \frac{(C_1/C_2)z^{-1/2}(1 - 2e^{-T_s/4R_{on}C_1})}{1 - z^{-1}} \quad (11.23)$$

When considered together, these nonidealities will interact with each other. For a complete analysis see [12]. A word of caution: When analyzing circuits under the assumption that the op-amp gain is finite, the inverting op-amp terminal is no longer a virtual ground. Therefore, the effect of all switched and fixed capacitors connected to the summing junction, intentional or parasitic, must be considered.

The gain error is defined in Table 11.1 as the relative error in the gain of the integrator and is due to a number of different causes. Capacitor mismatch causes gain error in a straightforward manner, since the gain of the integrator is determined by the ratio of the switched input capacitor to the integration capacitor. As can be seen from Eqs. (11.21), (11.22), and (11.23), gain error is also caused by finite op-amp gain and incomplete linear settling due to switch resistance or finite op-amp bandwidth. Gain error has little effect on the performance of a single-loop modulator. In a cascaded modulator, gain error will cause a leakage of noise to the overall modulator output that is shaped to the same order as the order of the first modulator in the cascade. Integrator pole error is caused by finite op-amp gain [cf. Eq. (11.21)]. Note that finite settling time does not give rise to a pole error. A pole error has little effect on the performance of a single-loop modulator. In a cascaded modulator, however, a pole error will cause a leakage of noise to the overall modulator output that is shaped to one order less than the order of the first modulator in the cascade.

11.4.2 Nonlinear Effects

Nonlinear effects are in general much more difficult to treat analytically than linear effects. The approach taken here is to treat the nonlinearity as a small perturbation to the ideal system response. The effect of nonlinearities on the switched-capacitor integrator output is discussed, as well as their effect on the performance of the overall $\Delta\Sigma$ modulator.

The nonlinearities are due to voltage-dependent capacitors, signal-dependent switch charge injection, and the fact that the device parameters of the MOS transistors are operating-point dependent. The errors due to nonlinear capacitors, signal-dependent charge injection, or a nonlinear op-amp dc transfer function will occur even with infinite settling time. Some nonideal effects such as the settling of the op-amp dependent on its operating point and the nonlinear resistance of the MOS switches will only cause nonlinearities if the settling is incomplete. In the following analysis, only the effects of nonlinear capacitors and a nonlinear op-amp dc transfer function will be considered. Signal-dependent switch-charge injection and incomplete nonlinear settling due to finite switch on-resistance have the same effect as a nonlinear switched capacitor. Nonlinearities due to incomplete op-amp settling have a similar effect to a nonlinear op-amp dc transfer function but are more complex, since they are dependent on the prior state of the circuit as well as the present. System-level simulation based on a table look-up method [13] may be the best way to determine the combined effect of all the different causes of nonlinearity on your system performance.

In the following analysis, it will be assumed that the modulator is constructed with fully differential circuitry. This will result in the cancellation of all even-order distortion terms (harmonic and intermodulation), regardless of the cause of the distortion. This is one of the main advantages in using fully differential circuits for high-performance switched-capacitor circuits. It is also assumed, to simplify the analysis, that the power of a harmonic decreases with increasing harmonic order. Therefore, in this analysis, only third-order nonlinear terms are considered.

The charge–voltage relationship for the nonlinear capacitor is then given by

$$q = C_1 V + C_3 V^3 \quad (11.24)$$

The nonlinear op-amp dc transfer function is given by

$$V_o = a_1 V_i + a_3 V_i^3 \quad (11.25)$$

The effect that a nonlinear capacitor has on modulator performance depends on the placement of the capacitor within the modulator. If a 1-bit DAC is employed, then a nonlinearity in the switched input capacitor implementing the D/A feedback does not cause any distortion at the output of the modulator. This is because the feedback capacitor is charged to only two distinct voltage levels, so the feedback will be inherently linear. A nonlinearity in the switched input capacitor that carries the signal does cause distortion in the output of the modulator. The amount of distortion is easy to calculate, since the capacitor only passes the input signal. If the capacitor is described by Eq. (11.24), then for a sinusoidal input of amplitude A , the signal-to-third-harmonic distortion is given by

$$S/D = \frac{C_3}{4C_1} A^2 \quad (11.26)$$

The effect of a nonlinear integrating capacitor is more difficult to calculate. The output of the first integrator will contain shaped noise in addition to the input signal. The effect of a nonlinearity in the integrating capacitor is harmonic distortion, plus a rise in the noise floor due to the intermodulation of the shaped broadband quantization noise. To see the effect this will have on the modulator output, the noise and distortion should be referred to

the input by dividing these terms by the transfer function of the integrator. This “noise-shaping” effect will cause a nonlinearity in the integrating capacitor to be much less of a problem than a nonlinearity in the switched input capacitor that carries the signal. The effect is difficult to handle analytically and is dependent on the structure and order of the modulator, so a single example will be given to demonstrate the magnitude of the problem. From computer simulation, a second-order single-loop modulator with an oversampling ratio of 128 has a $S/(N+D)$ of 84.4 dB with a 1-V peak-to-peak (p-p) sinusoidal input and a feedback of ± 1 V. If the switched input capacitor is equal to the integrating capacitor and the parameters for the integrating capacitor as specified in Eq. (11.24) are $C_1 = 1$ and $C_3 = 0.035$, the $S/(N+D)$ will degrade to 81.2 dB due to the nonlinearity. Note that the same degree of nonlinearity in the switched input capacitor would yield a third harmonic only 53.2 dB down from the fundamental.

The effect of a nonlinear op-amp dc transfer characteristic on the distortion performance of the integrator is next calculated. For this calculation, the transfer characteristic of Eq. (11.25) is assumed. The nonlinearity is assumed to be a small perturbation, and only first-order terms in the perturbation are considered. The circuit is first solved for the fundamental, with $a_3 = 0$. Then the third-order term is calculated from the fundamental present at the output. Finally, the circuit is analyzed at the third-harmonic frequency to include the effect of feedback on reducing the third-harmonic term. The resultant expression is

$$V_{03} = \frac{(a_3/a_1^3)V_0^3}{1 + [a_1k(1 - z^{-1})]/(1 - kz^{-1})} \quad (11.27)$$

Here, V_{03} is the third-order output term, V_0 is the fundamental output term, $k = C_2/(C_1 + C_2)$, C_1 is the value of the switched input capacitor, C_2 is the integrating capacitor, a_1 and a_3 are defined in Eq. (11.25), and $z = e^{j\omega\tau}$, where ω is the radian frequency of the component being evaluated, whether it is a third harmonic or an intermediate-frequency (IM) product.

Several observations can be made from Eq. (11.27). First, it is desirable to have an op-amp with a reasonably linear transfer function to begin with. For distortion components of frequencies less than f_p/a_1 , the op-amp will essentially be operating open loop. The distortion performance will be improved by a factor of approximately the open-loop op-amp gain for distortion components of frequencies greater than f_p , the pole frequency of the switched input capacitor and the integrating capacitor. This indicates that for the same relative nonlinearity (a_3/a_1^3), the higher the open-loop gain of the op-amp, the lower the distortion.

A nonlinear op-amp will have much the same effect as a nonlinear integrating capacitor in a $\Delta\Sigma$ modulator. The op-amp output contains both signal and shaped noise, so the nonlinearity will give rise to harmonics and an overall increase in the noise floor. When referred back to the input, the in-band noise is reduced by a factor $1 - z^{-1}$.

In concluding this section on the effect of nonlinear components on the performance of a $\Delta\Sigma$ modulator, the following observations are made:

1. Nonlinear effects are difficult to handle analytically, inaccurate to simulate, and sometimes even difficult to consider physically.

2. Because of noise shaping, nonlinear effects are most important at the input (switched input capacitor). The next most important nonlinearities are at the output of the first integrator (nonlinear integrating capacitor, nonlinear op-amp) and the input to the second integrator. Distortion in succeeding integrators is less important.
3. The effect of a nonlinear element will depend on the structure of the modulator, but in general there is no clear advantage to either single-loop or cascaded structures.
4. If the relative nonlinearity does not change, increasing the open-loop op-amp gain will help reduce distortion due to the op-amp.
5. Decreasing the gain of the integrator (i.e., the ratio of switched capacitor to integrating capacitor) will improve the distortion performance as well as the settling time but will increase the effect of nonidealities of the succeeding integrators.
6. The best approach to dealing with nonlinearities is to have a good qualitative understanding of their causes and effects. Design to minimize distortion without sacrificing too much performance in other areas. Fabricate the chip, analyze it to find out what is limiting your performance, tweak the design, and do a second pass. The only way to get quantitative results on the effect of nonlinearities on modulator performance is through computer simulation. This is not only cumbersome but may also be very inaccurate due to the difficulty of modeling these very small effects.

11.4.3 Intrinsic Noise

Intrinsic noise refers to noise that is generated in the device itself, as opposed to noise that couples in from an external interfering source. Intrinsic noise cannot be eliminated by shielding, filtering, or circuit layout, since it is a property of the device, but its value can be altered by choice of circuit topology and component size. In this section the two most important noise mechanisms in MOS devices, thermal noise and flicker (or $1/f$) noise, will be discussed.

Thermal noise is caused by the random fluctuation of carriers due to thermal energy and is present even at equilibrium (e.g., in a turned-on MOSFET with zero current flow). Because of this, it needs to be taken into account for both the switches and op-amps in a switched-capacitor circuit. Thermal noise has a white spectrum and a wide band, limited only by the time constants of the switched capacitors or the bandwidths of the op-amps. Since aliasing needs to be taken into account, calculations involving thermal noise are often complicated.

Consider a capacitor and a resistor in series with a switch that periodically opens, sampling a noise voltage onto the capacitor (Figure 11.13). If the pole associated with the RC time constant is at a frequency much higher than the sampling frequency f_s (usually a requirement for switched-capacitor circuits), then all the thermal noise power can be considered as being aliased into a band from 0 to $f_s/2$.

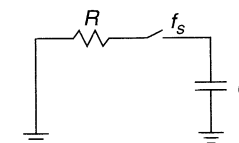


Figure 11.13 Periodically sampled capacitor.

To calculate the total noise power, model the resistor as having a noise source in series with a power source equal to the Johnson noise $4kTR \Delta f$. The total noise power can be found by evaluating the integral

$$e_T^2 = \int_0^\infty \frac{4kTR}{1 + (2\pi fRC)^2} df = \frac{kT}{C} \quad (11.28)$$

It is interesting to note that while the thermal noise is generated in the resistor, the total noise power depends only on the capacitor. Since the noise is aliased down to the band from 0 to $f_s/2$, the final spectrum is white with a spectral density

$$S(f) = \frac{2kT}{f_s C} \quad (11.29)$$

In most switched-capacitor circuits, there are two samplings of thermal noise per clock period, resulting in a spectral density

$$S(f) = \frac{4kT}{f_s C} \quad (11.30)$$

It should be noted that the noise spectral density of a switched capacitor is the same as that of its equivalent resistor, $R_{eq} = 1/f_s C$, in the band from 0 to $f_s/2$; that is,

$$S_T(f) = 4kTR_{eq} = \frac{4kT}{f_s C} \quad (11.31)$$

The thermal noise of an op-amp can be modeled as a noise voltage source in series with one of the input terminals. The gate-referred thermal noise spectral density of a MOSFET in saturation is white and is equal to $S(f) = 8kT/3g_m$. The value of the noise voltage source e_T can be found for a particular op-amp by input referring the thermal noise of the individual transistors [see, e.g., Eq. (11.19)]. When the noise is sampled, the total noise power is aliased into the band from 0 to $f_s/2$. The total noise power is $e_T^2 = S_T B$, where B is the noise bandwidth of the amplifier in a given configuration and S_T is the input-referred spectral density. It is interesting to note that if the thermal noise of the amplifier is dominated by the input devices, then the aliased white-noise density will depend only on the compensation capacitor. This is because S_T will be inversely proportional to the g_m of the input devices, and the bandwidth will be proportional to g_m . Because of the gain provided by the first integrator at the lower in-band frequencies, the thermal noise performance of a $\Delta\Sigma$ modulator is determined mainly by the switch noise and the op-amp noise in the first integrator. The spectral density due to the switch thermal noise is

$$S_T(f) = \frac{4kT}{f_s} \sum_i \frac{1}{C_i} \quad (11.32)$$

where the C_i are the switched input capacitors.

Typically there will be two switched input capacitors, one carrying the signal and one providing the feedback from the modulator output. To calculate the in-band thermal noise power, the spectral density must be multiplied by the baseband cutoff frequency f_c . It is then seen that the in-band thermal noise power due to the switches will decrease by 3 dB per octave of oversampling. If the modulator is implemented with fully differential cir-

cuitry, an additional 3-dB improvement in the signal-to-thermal noise ratio will be realized. This is due to the fact that the number of switched input capacitors doubles for a fully differential circuit, but the noise is uncorrelated, so the thermal noise power doubles, or increases by 3 dB. On the other hand, a fully differential circuit also doubles the signal swing, which represents an increase in signal power of 6 dB.

The input-referred thermal noise spectral density of the op-amp is given by the total thermal noise power, $S_T B$, divided by $f_s/2$. The total thermal noise power of the op-amp depends on the relative transconductances of the noise-contributing devices that make up the amplifier and is inversely proportional to the value of the compensation capacitor. To determine the in-band thermal noise power, the power spectral density is multiplied by the baseband cut-off frequency f_c . Again, because the noise is unshaped, there is a 3-dB improvement per octave of oversampling. If the modulator is implemented as a fully differential circuit, then the signal power will increase by 6 dB, but the thermal noise power of the op-amp will not change. Therefore, using fully differential circuitry results in a 6-dB improvement over the single-ended case.

Flicker noise is also called $1/f$ noise because it has a spectral density that varies approximately inversely with frequency. This noise in MOS devices is attributed to the variation in channel charge caused by charges being captured and released by traps with a distribution of time constants. The $1/f$ noise in a MOSFET can be modeled by a voltage source in series with the gate. The spectral power density of the source is given by

$$S_{1/f}(f) = \frac{K}{WL} \frac{1}{f} \quad (11.33)$$

where K is an empirically determined constant. The equivalent $1/f$ gate noise is usually modeled as being independent of bias conditions. The empirical constant K is dependent on the process and whether the device is a P - or N -channel one. It is recommended that the $1/f$ noise be measured on a device with the same gate length and bias conditions and in the same process as the actual device that will be used in the circuit, since the data do not always fit the model very closely.

Since $1/f$ noise is caused by a fluctuation in the number of carriers in the channel, a device with no current flowing through it has no $1/f$ current noise. Another way to think of this is that the gate noise voltage is independent of bias current, and a device with no current flowing through it has a g_m equal to zero. This implies that switches that are fully “on,” with $V_{DS} = 0$, or fully “off” at the time of sampling contribute no $1/f$ noise. Therefore, in a switched-capacitor circuit, $1/f$ noise is only of concern in the op-amps. Flicker noise is easier to handle computationally than thermal noise, since it is predominant at low frequencies, and aliasing does not have to be taken into account.

The input-referred $1/f$ noise of a $\Delta\Sigma$ modulator is very nearly equal to the input-referred $1/f$ noise of the op-amp in the first integrator. Because of noise shaping, the contributions of op-amps in succeeding integrators to the overall $1/f$ noise of the $\Delta\Sigma$ modulator can be neglected. Equation (11.19) can be used to calculate the input-referred $1/f$ noise of the op-amp from the $1/f$ noise contributions of the devices that make up the op-amp.

Since $1/f$ noise predominates at low frequencies, it cannot be reduced substantially by increasing the oversampling ratio. Equation (11.33) suggests one method for decreasing the $1/f$ noise, namely by increasing the gate area of the MOSFETs that contribute to the op-amp noise. This technique will yield a 3-dB improvement for every doubling of

MOSFET gate area. A far more efficient method for reducing $1/f$ noise, which lends itself nicely to switched-capacitor circuits, is to employ an autozeroed integrator as the first integrator in the $\Delta\Sigma$ modulator. An example of such an integrator is described in Chapter 6, Section 6.4.1. The effect of the autozeroing operation is to store the value of $1/f$ noise present at the end of the previous half-period on an input capacitor. This value is then subtracted from the $1/f$ noise present at the end of the current period. This is equivalent to multiplying the $1/f$ noise voltage by a factor of $1 - z^{-1/2}$. Since a $\Delta\Sigma$ modulator employs a high degree of oversampling, the autozeroing operation usually causes the amplifier $1/f$ noise to no longer be of any concern in the design of the modulator.

11.5 MODULATOR COMPONENT DESIGN CONSIDERATIONS

11.5.1 The Feedback DAC

The purpose of a feedback DAC in a closed-loop ADC is to convert the ADC's digital output back into an analog form to be compared to its analog input. In essence, the DAC controls the mapping between the analog and digital domains. Therefore, the performance of the closed-loop ADC is completely dependent on the accuracy of its feedback DAC and the way in which its output is compared to the ADC input [14].

One of the primary features of $\Delta\Sigma$ converters is that they can use low-resolution DACs in their feedback paths. Low resolution, however, is not the same as low accuracy. In fact, if a multibit DAC is used, the linearity of that DAC must match the overall linearity requirements of the system. For example, if a four-level DAC is used in a 16-bit linear system, the matching requirements between any two steps in the DAC are as stiff as those of a 16-bit linear DAC. This is usually not possible without calibration or trimming, although dynamic matching may also be used.

It is for this reason that 1-bit DACs are typically used in $\Delta\Sigma$ converters. Since they only have two levels, these DACs are guaranteed to be free from differential nonlinearity, since all step sizes are identical (there is only one step). There are, however, other mechanisms for nonidealities in the 1-bit DAC. These mechanisms can be separated into three categories: nonidealities associated with the voltage reference, with the way the charge is taken from the reference, and with the way charge is delivered to the integrator.

11.5.1.1 Reference Nonidealities. The output of the DAC is the product of the digital input and the voltage reference. This time-domain multiplication results in a convolution in the frequency domain. Therefore, if the reference has any signals on it (other than dc), their spectra will be convolved with the $\Delta\Sigma$ converter's bit stream. This can be particularly troublesome since the spectrum of the bit stream is far from white and, in fact, usually contains spectral lines (sticks) at high frequencies that are not reduced by the loop's noise-shaping properties. Signals on the reference with spectral components near those of the spectral sticks in the bit stream will cause those sticks to be modulated and convolved, transferring them down to audible frequencies near dc. This transference of frequencies results in unwanted tones in the passband of the converter, known as idle tones. A particularly troublesome reference interference frequency occurs at half of the modulator sampling frequency ($f_s/2$), since most of the spectral sticks in the bit stream occur near this frequency, and this frequency is common in most A/D conversion systems.

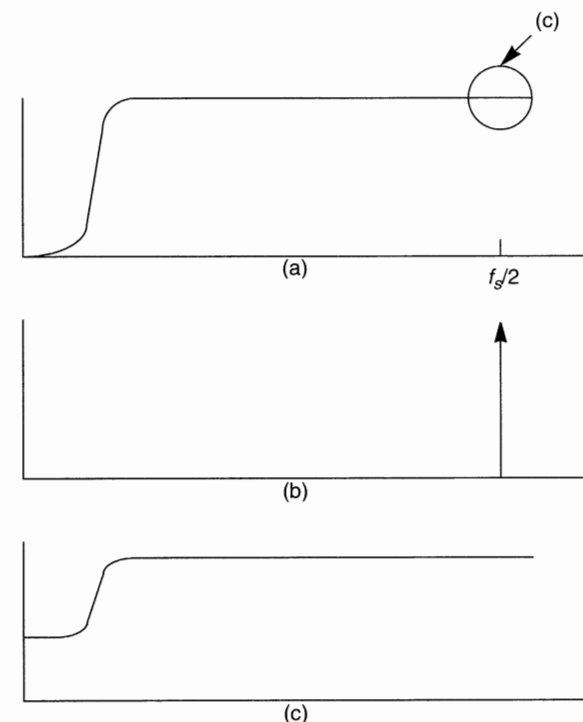


Figure 11.14 Effects of $f_s/2$ on the reference voltage: (a) original noise-shaping spectrum; (b) spectrum of the reference with $f_s/2$ pickup; (c) new spectrum with excess noise.

Figure 11.14 shows the effects of $f_s/2$ interference on the reference. The top graph shows a noise-shaped spectrum, the second graph shows the spectrum of the interfering noise, and the third the convolution of the voltage reference spectrum and the noise-shaped spectrum. Note how the high-frequency noise gets folded down into the low-frequency region to create increased in-band noise.

11.5.1.2 Charge-Taking Nonidealities. In a switched-capacitor $\Delta\Sigma$ ADC, charge is taken from the reference and delivered to integrator(s) by the feedback DAC. The amount of charge and the instant in time at which it is taken from the reference must be identical for each clock cycle. If it is not, then the reference will be at a different potential each time a sample is taken from it. This will have the effect of putting unwanted signals on the reference.

There are two causes of charge-taking nonidealities: state-dependent reference loading and transition-dependent reference loading.

In state-dependent reference loading, a "1" in the bit stream will cause the feedback DAC to draw a different amount of charge from the reference than a "0." This will cause the reference voltage to vary with the 1's density in the bit stream, which will create a gain error in the converter. [The 1's density is defined as the ratio of the number of 1's to the number of possible 1's (or the number of data points) in the bit stream in a given time interval.]

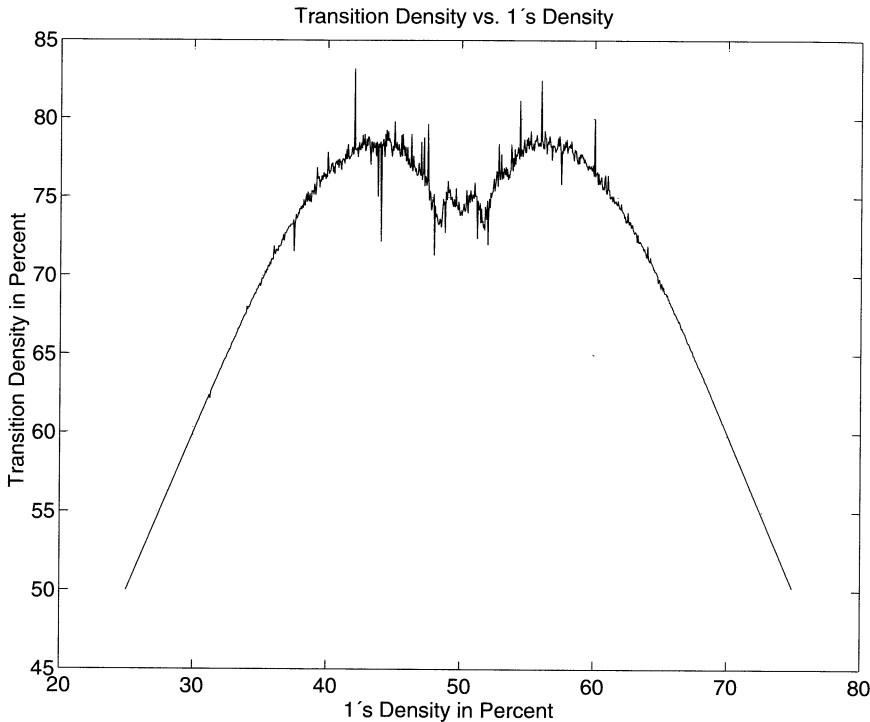


Figure 11.15 Transition density versus 1's density for a fourth-order modulator.

The second most common form of charge-taking nonideality occurs with transition-density-dependent DAC schemes. Transition density, like 1's density, is defined as the ratio of the number of transitions to the number of possible transitions (or the number of data points) in the bit stream in a given time interval. A transition occurs when the bit stream changes state, from a 0 to a 1 or from a 1 to a 0. For example, for an alternating 1–0–1–0 pattern, there is one transition per clock cycle and the transition density is 1. Therefore, near midscale, or a 50% 1's density, the transition density is highest. As the 1's density approaches 0 or 100%, the transition density approaches 0. A plot of transition density versus 1's density for a fourth-order modulator appears in Figure 11.15.

If the load on the reference depends on the transition density, the reference voltage may vary nonlinearly with 1's density. This can cause a bow in the input/output transfer function of the converter, which will cause integral nonlinearity and second-harmonic distortion.

11.5.1.3 Charge-Delivery Nonidealities. Nonidealities can also occur when the charge is being delivered to the integrator. If the charge stored on the feedback capacitors in the DAC is not fully delivered to the integrator each clock cycle, then a gain error can result. If the time allowed to deliver the charge is also dependent on when the comparator makes its decision, then the amount of charge delivered could depend on the comparator's input voltage. This could create a signal dependence of the reference charge

delivered, which will create distortion and possibly a noise-folding problem. Therefore, it is important that the clock generation circuitry be designed in such a way as to ensure that the timing of charge delivery be independent of the timing of the comparator decision.

It is also important that the full logic-level signal be delivered to the switches, regardless of the comparator's decision. This requires full settling of the comparator.

Other charge-delivering nonidealities can occur due to parasitic switched capacitances in the circuit. Unintentional parasitic switched capacitances can cause input signals to be delivered to the integrator in a bit-stream-dependent way, which will also cause distortion.

We see from the discussion in this section that the design of a simple 1-bit DAC is really a very complex task. An example of a 1-bit DAC design that avoids the problems mentioned appears in Figure 11.16 [15]. The figure shows the first-stage integrator of a $\Delta\Sigma$ modulator with an input path and a 1-bit feedback path. During phase Φ_2 , the top capacitor charges to reference while the lower capacitor charges to ground. During phase Φ_1 , the

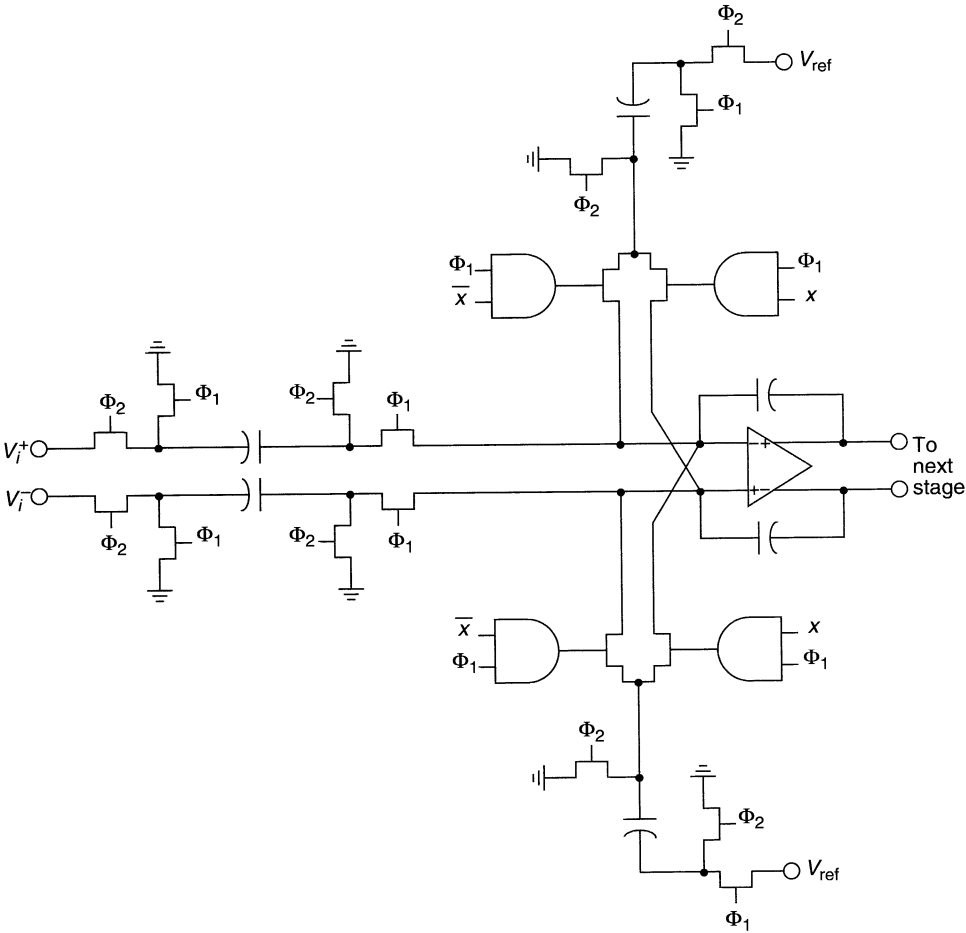


Figure 11.16 Commercially implemented differential 1-bit DAC.

comparator decision is used to decide which summing junction is to receive a positive reference charge and which is to receive a negative reference charge. If x is high (and $\bar{x}$ is low), then a reference charge is taken from the lower summing junction, while a reference charge is delivered to the upper summing junction. If x is low, the charge delivery is swapped.

This circuit guarantees that the reference sees the same load, independent of transition density, since both capacitors are charged between reference and ground every clock cycle. Some other schemes do not share this property, creating a transition-dependent reference load and second-harmonic distortion.

11.5.2 The Comparator

The purpose of the comparator in a $\Delta\Sigma$ modulator is to quantize a signal in the loop and provide the output of the modulator. This output is fed to the 1-bit DAC, which reconverts the digital signal to an analog one to be used in feedback to close the modulator loop. Since the comparator appears after the loop gain block and before the output terminal, nonidealities associated with it are shaped by the loop in the same way that the quantization noise it produces is shaped. This means that, at the frequencies of interest, the comparator nonidealities are reduced the most.

This can be most easily shown by considering the offset of the comparator. In a second-order loop, two integrators appear between the input terminal of the loop and the comparator. Therefore, if there is an offset in the comparator $V_{os,comp}$, then the input-referred offset of the loop will be $V_{os,input} = V_{os,comp}/A_{integ}^2$, where A_{integ}^2 is the combined dc gain of both integrators.

Even though the comparator nonidealities are reduced by the noise-shaping properties of the loop, there are still some circuit design issues to watch out for. The most important issue is one of metastability. If the input to the comparator is so small as to cause metastability, then it probably does not matter whether it puts out a 1 or a 0, since the inaccuracy and associated quantization noise from either decision will be nearly identical. It is very important, however, that a decision be made and delivered in time to the feedback DAC. It is also very important that the *same* decision that is delivered to the DAC be also delivered to the modulator output! If this does not happen in an audio converter, a loud “click” will be passed through the decimation filter, which will make the recording sound as if it was made from a scratched record.

Another area to watch out for is comparator hysteresis. Hysteresis in the comparator will cause decisions to be dependent on previous decisions, putting memory into the comparator. This memory can create unwanted system poles that may cause errors in the signal and noise transfer functions.

Figure 11.17 shows a comparator used in a commercial $\Delta\Sigma$ modulator implementation [15]. A dynamic latching comparator structure could be used since only periodic answers are required in all $\Delta\Sigma$ loops. Thus, power and chip area is saved. The EVAL input is used in one clock phase to reset the comparator to remove dependencies on previous decisions, and in the other phase to latch the answer and deliver it to the outputs. The output buffering inverters are sized so that they both put out 0's until the comparator has sufficient overdrive to make one output rise to a 1. A latch is placed at the comparator output and closed after a finite period of time. If the comparator does not make a decision within

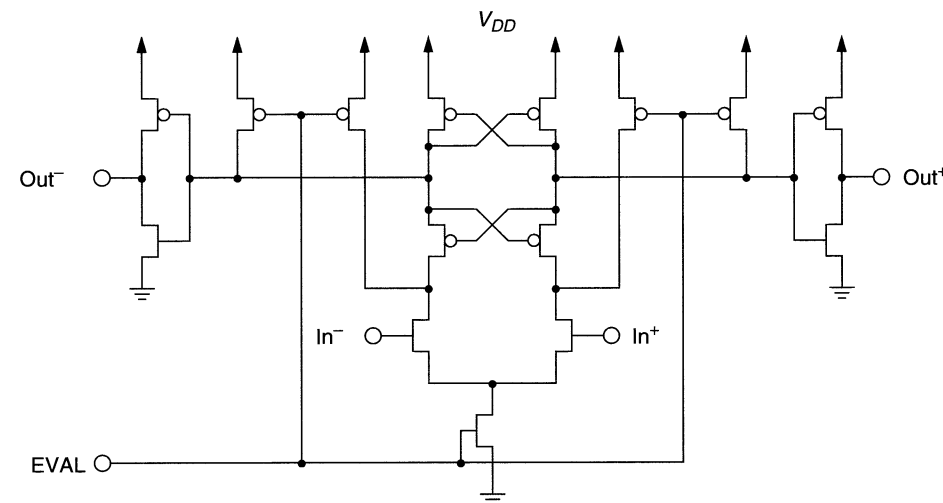


Figure 11.17 Commercially implemented differential comparator.

that period, the latch will receive two 0's and will assign a decision to be passed both to the DAC and the digital filter.

11.5.3 The Clock Generation Circuitry

Clock generation circuitry in a $\Delta\Sigma$ modulator is very similar to clock generation circuitry in a switched-capacitor filter. In their simplest form, the circuits require a two-phase nonoverlapping clocking scheme and data-dependent clocks for the feedback DAC. Improvements such as timing to open the summing junction switches first can be added, and additional phases can also be included for integrator offset correction or chopper stabilization. Clocks for the comparator and its associated logic should be designed so as to ensure that the comparator decision is delivered to the feedback DAC in a manner independent of the decision time. Finally, steps should be taken in the clock generation circuitry to ensure that the bit stream delivered to the feedback DAC is identical to that delivered to the decimation filter.

11.6 SYSTEM-LEVEL CONSIDERATIONS

11.6.1 Dynamic Range Considerations

When transferring the $\Delta\Sigma$ modulator design from a system diagram to an actual switched-capacitor circuit, the dynamic range of the analog components used in the circuit must be taken into account. Unlike the integrator in a software simulation, the value of the output of a physical switched-capacitor integrator will have an upper and lower bound determined by the output range of the operational amplifier. When the integrator output exceeds the output range of the op-amp, the integrator will saturate and become

very nonlinear. As the output of the integrator approaches the supply rails, it will eventually clip, or hard limit. When the integrator clips, an input that would cause the integrator output to move even further toward the supply rail will now have very little effect on the output of the integrator. In clip, the output of the integrator remains fixed, and the op-amp no longer has an effect on the circuit. The summing junction is no longer a virtual ground, but has a value determined by the input signal and by the passive RC equivalent formed by the switched input capacitor and the integrating capacitor.

Dynamic range scaling is accomplished by adjusting the values of both the fixed integrating capacitors and the switched input capacitors. If the integrating capacitor of a switched-capacitor stage is multiplied by a scale factor and every capacitor that the op-amp for that stage drives is multiplied by the same scale factor, then the voltage at the output of that op-amp will be reduced by the same scale factor. However, unless the op-amp output being scaled is the final output, the overall system transfer function remains unchanged. It would seem reasonable to make the signal swing at the op-amp outputs as large as possible in order to minimize the effect of circuit noise; however, scaling up the op-amp outputs will cause the amplifiers to clip at a lower level of input signal. Once the amplifier outputs exceed their normal operating range, the system performance will begin to degrade. The usual method for dynamic range scaling is to scale the amplifier outputs such that all the amplifiers just go into saturation when the input signal reaches its maximum allowable value. This scaling is intended to maximize the dynamic range of the circuit. In cases where circuit noise is not limiting the circuit performance, it may be appropriate to scale differently. As mentioned in Section 11.4.2, it may be desirable to scale down an op-amp output to improve settling time or linearity at the expense of noise performance.

For single-loop higher order modulators, the signal swing at the output of each integrator will depend on the modulator topology and the pole/zero placement. In general, though, the excursions at the integrator outputs will increase at higher signal levels. In these types of modulators, the input range is restricted to be somewhat less than the full range, which (normalized to the reference voltage) is ± 1 , so that the modulator will remain stable. A possible dynamic range scaling scheme is to scale the integrator outputs such that they just saturate at the maximum allowed value of input signal. Simulations should then be performed to verify that the system operates to specification with this scaling. Some designers have even used the saturation and clipping of the op-amps as the mechanism for ensuring that the modulator remains in the stable operating region [16]. Dynamic range scaling for first- and second-order single-loop modulators is somewhat simpler, since the maximum signal swing at the integrator outputs of these structures is easy to predict, even without simulation. Dynamic range scaling for cascaded modulators is also fairly straightforward, since cascaded modulators are composed of first- and second-order single-loop modulators.

The integrator output of a first-order modulator that employs a (normalized) feedback of ± 1 and has an integrator transfer function of $H(z) = 1/(1 - z^{-1})$ is constrained to lie between the values ± 2 . The maximum swing occurs for the case where the integrator output is just below zero and the modulator input is $+1$. The input signal will be added to the integrator output, and the feedback signal, which will be -1 , will be subtracted from the integrator output. Therefore, the integrator output at the end of the next clock period will be just below $+2$. The analysis for a second-order single-loop modulator with a feed-

back of ± 1 and integrator transfer functions of $H(z) = 1/(1 - z^{-1})$ is not as straightforward. It can be shown, however, that the output of the first integrator is constrained between the values ± 3 . The output of the second integrator could be bounded. A commonly used value for bounds on the second integrator is ± 5 . If this value is assumed, then the second integrator will not saturate until the input is greater than 0.64, or 4 dB down from full scale. This loss of dynamic range is generally not of great concern, since the SNR of a second-order single-loop modulator begins to degrade at about 3 dB below full scale anyway due to overload of the quantizer (assumed to be a comparator).

11.6.2 Clock Jitter

Calculating the effect of clock jitter on a $\Delta\Sigma$ ADC implemented in switched-capacitor technology is a fairly simple matter. The operation of switched-capacitor circuits depends on complete charging during each phase of the clock. Once the analog signal has been sampled, the switched-capacitor circuit is similar to an analog sampled-data computer. The lengths of the time slots and the variations in the lengths of the time slots have no direct effect. Therefore, the effect of clock jitter on a switched-capacitor circuit can be analyzed by examining its effect on the sampling of the input signal and on the reconstruction of the output signal. Since the output of an ADC is digital, the effect of clock jitter on the performance of a $\Delta\Sigma$ ADC is completely accounted for by taking into account the effect that it has on the sampling of the input signal. This also implies that the effect of clock jitter on a switched-capacitor $\Delta\Sigma$ modulator is independent of the structure or order of the modulator.

Consider the effect of clock jitter on the sampling of the input signal. A sinusoidal time jitter with amplitude α and frequency ω will cause a sampling of the input signal at time τ to instead occur at time $\tau + \alpha \sin \omega \tau$. The effect is the same as if the input signal $A \cos \omega_0 \tau$ were instead $A \cos[\omega_0(\tau + \alpha \sin \omega \tau)]$. This can also be written in the form $A \cos(\omega_0 \tau + \alpha \omega_0 \sin \omega \tau)$, which is recognizable as the expression for FM modulation. For $\alpha \omega_0 \ll 1$, the jitter will give rise to a pair of sidebands at $\omega_0 - \omega$ and $\omega_0 + \omega$ with an amplitude $A \alpha \omega_0 / 2$. It can be seen that the jitter is modulated by the input signal and its power is scaled by the factor $A^2 \omega_0^2 / 2$. Whether oversampling will help to reduce the output error caused by the jitter depends on the nature of the jitter. If the jitter is assumed to be white [5] and has a power $(\Delta \tau)^2$, then the resultant error will have uniform power spectral density from 0 to $f_s/2$, with a total power of $(A \omega_0 \Delta \tau)^2 / 2$. In this case, the in-band noise power will be reduced by the oversampling ratio. On the other hand, if the clock jitter has a $1/f$ characteristic ("close-in noise"), then the error will have a spectrum that appears as a "skirt" on the spectral line of the fundamental. In this latter case, oversampling will not reduce the in-band noise, and a $\Delta\Sigma$ converter will have the same sensitivity to jitter as a Nyquist-rate converter. Note that for continuous-time loop filters the effects are much more complex and harmful.

11.6.3 Input Impedance

In the final analysis, it is the performance of the entire A/D system that must meet specification, not just that of the $\Delta\Sigma$ modulator. Therefore, it is important to also consider the external circuitry that interfaces to the $\Delta\Sigma$ modulator. The output of the modulator is

digital, so interface concerns will consist mainly of clock timing and set-up and hold times. However, the input to the modulator is analog, and the input characteristics of the modulator must be understood to ensure that the external analog circuitry does not cause a degradation in the system performance.

The diagram used as the basis of the following discussion regarding the input impedance of the modulator is depicted in Figure 11.18. The diagram is drawn for a single-ended modulator, although the results can be extended to the fully differential case.

The external components R_x and C_x form an antialiasing filter for the switched-capacitor circuitry. Resistance R_x consists of the output resistance of the signal source, plus any additional external resistance. The varactor C_{J1} is a voltage-variable capacitor representing any unswitched junction capacitance on the input to the chip. Most likely C_{J1} will be dominated by the diodes in the electrostatic input protection network. Capacitor C_1 represents the switched-capacitor input and is a voltage-invariant capacitor that is alternately charged to the input voltage and discharged to analog ground at a frequency equal to the sampling rate f_s . Capacitor C_{J2} represents the voltage-variable capacitance associated with the source-drain junctions on the capacitor side of the input switches. Capacitor C_{J2} will be switched in a similar manner to C_1 .

The effect of the input impedance of the analog modulator on the system performance depends on the output impedance of the signal source. Resistor R_x , in conjunction with the parallel combination of C_x and C_{J1} , forms a low-pass filter. Since C_{J1} is voltage variable, this filter can cause distortion. The amount of distortion will depend on the signal frequency, on the nonlinearity of C_{J1} , and on the relative values of R_x , C_x , and C_{J1} . For a fixed pole frequency, the magnitude of the distortion will vary inversely with the size of C_x . The distortion will also increase with increasing signal frequency at a rate of 20 dB/decade. Usually C_x is of a large enough value to smooth out the current pulses from the switched capacitors C_1 and C_{J2} , so these capacitors appear as resistors for loading calculations. Switched-capacitor C_1 will be equivalent to a resistor of value $1/f_s C_1$ between the input and analog ground and will attenuate the signal to a degree determined by the value of R_x . Switched-capacitor C_{J2} appears as a nonlinear resistor between the input and

analog ground. The resistor divider formed by this nonlinear resistor and R_x can give rise to distortion. This distortion will not vary with the signal frequency but will increase with an increase in the sampling frequency for a fixed value of R_x .

11.7 LAYOUT CONSIDERATIONS

11.7.1 Signal Paths

As in most IC layouts, the signal paths of a switched-capacitor $\Delta\Sigma$ ADC are the most critical. It is extremely important, for example, that noise at multiples of the sampling clock frequency does not get on the input lines or feedback DAC lines. Noise at those frequencies will be aliased by the sampling of the switched capacitors down near dc and will be indistinguishable from low-frequency signals.

On-chip noise coupling can occur even if the user provides proper antialiasing protection off chip. Clock lines going over sensitive input or feedback DAC lines can disturb these lines just before their signals are sampled, which will cause an increase in noise. If these disturbances have any bit-stream information in them, they can also cause distortion and tones. Substrate noise can also present a problem. It is very difficult to shield a circuit entirely from this noise source, so the best weapon against substrate coupling is the use of differential circuitry. If such circuits are employed, it is very important that both sides are disturbed by the substrate in an *identical* fashion. This means that they must be matched in area with each other, even if it means adding extra interconnect to do so. It is also a good idea to put sensitive nodes on the second metal layer and place a low-impedance noise-free signal in the first metal underneath them.

Noise coupling into the input nodes will create noise sources that are indistinguishable from the signal due to the proximity of the coupling in the loop. This is not so for noise coupling into internal nodes of the $\Delta\Sigma$ modulator loop. If noise is coupled into the summing junction at the comparator input, for example, it will receive the same noise shaping as the 1-bit quantization noise does. The effect can be observed in the output spectrum as a noise source that increases with increasing frequency. In another example, if the coupling causes a signal distortion at the input to the second stage, the loop will exhibit a signal distortion that increases with frequency at a rate of 6 dB/octave due to the shaping caused by the first stage.

11.7.2 Busses

In a differential circuit, it is always best to keep things as symmetric as possible. This often points to the use of mirror-symmetric cells. These are cells that are used in pairs like bookends, where one-half is the mirror image of the other and placed side-by-side with its counterpart. This concept can be carried to its extreme, placing a mirror-symmetric op-amp in the center of the layout, capacitors on either side of it, switches further out from center, and power bussing and logic signals on the far outside. This leads to a very inefficient layout, since the area required for the power bus, logic signal busses, and analog signal busses is all doubled for the benefit of exact signal symmetry. This layout practice is shown in Figure 11.19, which represents a stereo $\Delta\Sigma$ ADC modulator chip.

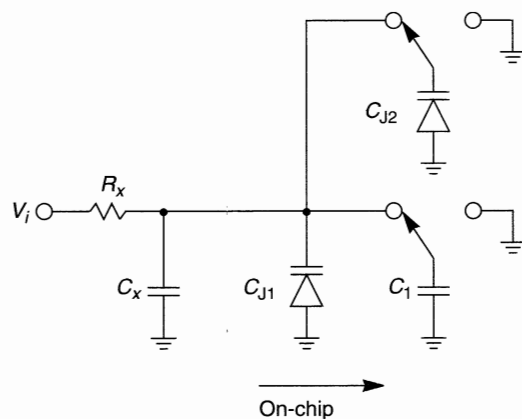


Figure 11.18 Equivalent model for input to the modulator.

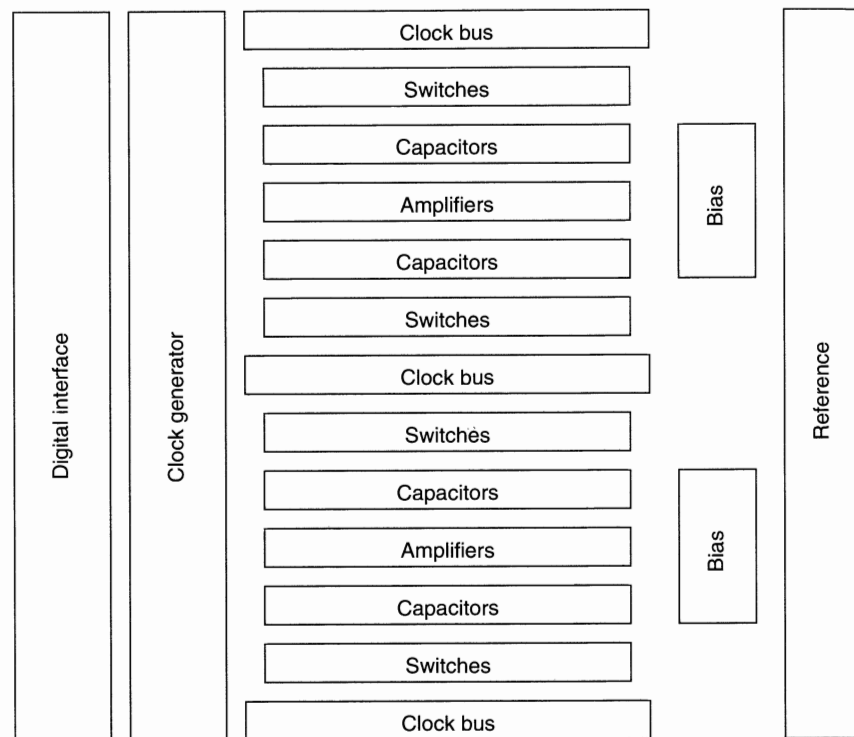


Figure 11.19 Fully differential layout carried to extremes.

It is for this reason that it is usually best to put the power busses on one side of the circuit, the logic and analog signal busses on the other, and the mirrored half-cells rotated 90 degrees between them. This layout practice can save a considerable amount of chip area relative to the first one and is shown for comparison in Figure 11.20. When using this practice, it is important to ensure that symmetrical coupling into the signal paths is maintained by either manually or automatically checking parasitic capacitance and resistance.

11.7.3 RF Coupling

It is very important to avoid RF coupling into the signal and reference paths of the sampling $\Delta\Sigma$ modulator. On-chip clock line coupling has already been mentioned. Coupling between the user-supplied antialiasing filter and the package pin can also cause problems. This can occur simply by using the wrong type of capacitor in the external low-pass filter. Polystyrene capacitors are sometimes used in audio circuits for their excellent linearity. These capacitors are large and hence make excellent antennas for the reception of RF signals, so they should not be used for the input bypassing capacitor on a switched-capacitor circuit. A much better choice would be a ceramic NPO (zero temperature coefficient) capacitor of the same value, and the best choice would be the surface-mount version of the ceramic NPO capacitor. If this capacitor is placed very near the package pin, the amount of wire that the RF noise can couple into can be minimized.

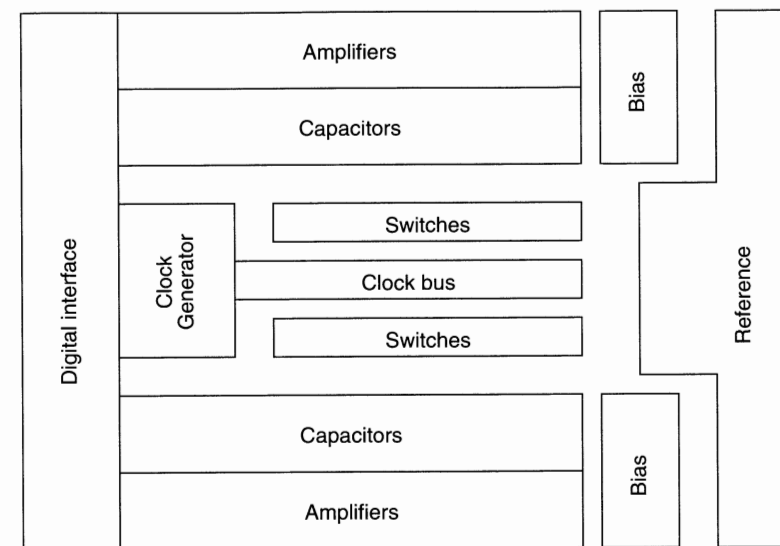


Figure 11.20 A smaller pseudo-differential layout saves area.

Even with these extraordinary measures, the RF coupling can still be picked up by the bond wires inside the package! These are usually long and thin and make excellent antennas for extremely high frequency signals. These signals can be sampled by the input sampling switches of the $\Delta\Sigma$ converter, since the roll-off frequency of these switches is usually 5–10 times the modulator sampling frequency to avoid distortion due to incomplete settling across the nonlinear MOSFET resistance. Some of the high-frequency noise can be reduced by placing a small amount of on-chip RC filtering between the input bond wire and the sampling switch. Another technique is to arrange properly the bonding pads on the IC to place relatively clean signals such as analog ground next to the sensitive input and reference bonding pads.

Finally, it is very important to minimize the amount of RF transmission in and around the IC. This can be accomplished by minimizing the load on the digital output drivers to minimize the current spikes transmitted through the package bonding wires and nearby circuitry. It can also be accomplished by proper timing of the transmission of the digital signals from the chip with respect to the timing of the sampling instants. Figure 11.21 shows improper and proper relationships between logic transitions and sampling instants. The top graph shows the sampling of an input voltage being corrupted by an incompletely settled disturbance coupled to it. The bottom graph shows a proper relationship between the sampling instants and transitions on the serial output. The serial output is potentially particularly offensive, since it contains extremely distorted signal information, which can cause idle tones and noise if it is coupled back to the input and reference signals.

11.7.4 Interfacing to the ADC

The last and perhaps most important part of layout considerations is the way in which the $\Delta\Sigma$ converter is connected in an application. If the $\Delta\Sigma$ converter uses switched-capacitor inputs, it is important that signals have had proper antialiasing filtering at the input to the

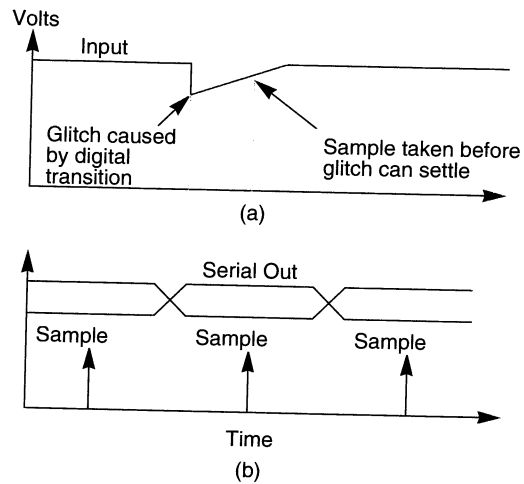


Figure 11.21 Timing of digital interference is important: (a) shows sampling disturbance; (b) shows a proper relationship between output transitions and sampling instants.

switched-capacitor circuit. It is also very important that the action of the switched capacitors taking charge from the input circuitry do not cause a nonlinear event to occur. This means that the preamplifier circuit cannot be put into a situation where it goes into slew limiting.

Slew limiting can be avoided by placing enough series resistance between the preamplifier circuit and the switched-capacitor circuit. However, if this resistance becomes too large, the average current drawn by the switched-capacitor circuitry can create significant voltage drops across it that will cause a gain error in the circuit. A more troublesome effect comes when the input sampling switches are complementary MOS pass gates. The amount of charge injection from the opening of the pass gate MOS switches is a very nonlinear function of the input voltage. This charge injection happens every clock cycle, creating a nonlinear charge per sample period or nonlinear current to flow. If this current creates a large enough drop across the input resistor, a large nonlinear voltage distortion at the input can result. This distortion can be measured at the input to the switched-capacitor circuit with a spectrum analyzer.

The other important node in a $\Delta\Sigma$ converter is the reference input. If signals with frequencies near $f_s/2$ get coupled onto this node, quantization noise and tones will be folded down to the low frequencies where they can affect the performance of the converter. The reference should therefore be bypassed with a large capacitor with excellent high-frequency characteristics, such as a tantalum capacitor. Input signals can also be inductively coupled onto the reference lead if the leads carrying the input signals are near the reference signal leads either inside or outside the package. The current spikes in one wire can induce voltage drops across an adjacent wire in the same fashion as in a transformer. This will place an input signal component on the reference voltage that will cause a second-harmonic distortion at the converter output, as signal on the reference results in a squaring of the input signal in an ADC.

11.8 DESIGN EXAMPLES

11.8.1 Second-Order Single-Stage $\Delta\Sigma$ Modulator

As discussed in Section 11.2, single-stage $\Delta\Sigma$ modulators that employ 1-bit quantizers are quite tolerant of gain errors that result from capacitor mismatch or incomplete linear settling. They are also relatively insensitive to integrator leakage resulting from finite dc gain in operational amplifiers as well as to offset voltages and comparator hysteresis. As a result, the principal circuit design consideration for single-stage modulators is to minimize noise and distortion in the first integrator. As an example, special emphasis has been placed on preventing slewing distortion in the first integrator of the second-order $\Delta\Sigma$ modulator described in this section. The modulator was designed for digital-audio signal acquisition at an oversampling ratio of 256. The performance objective was a signal bandwidth of 20 kHz and a 16-bit resolution (98 dB dynamic range) while operating from a single 5-V power supply.

Figure 11.22 shows a fully differential switched-capacitor implementation of the modulator consisting of two identical parasitic-insensitive switched-capacitor integrators, a comparator that serves as a 1-bit ADC, and a distributed two-level (1-bit) DAC. The modulator operates on a two-phase nonoverlapping clock, similar to that employed in many digital systems. During phase 1, switches S_1 and S_3 conduct so that the differential input to the modulator is sampled onto the left sides of the first integrator's sampling

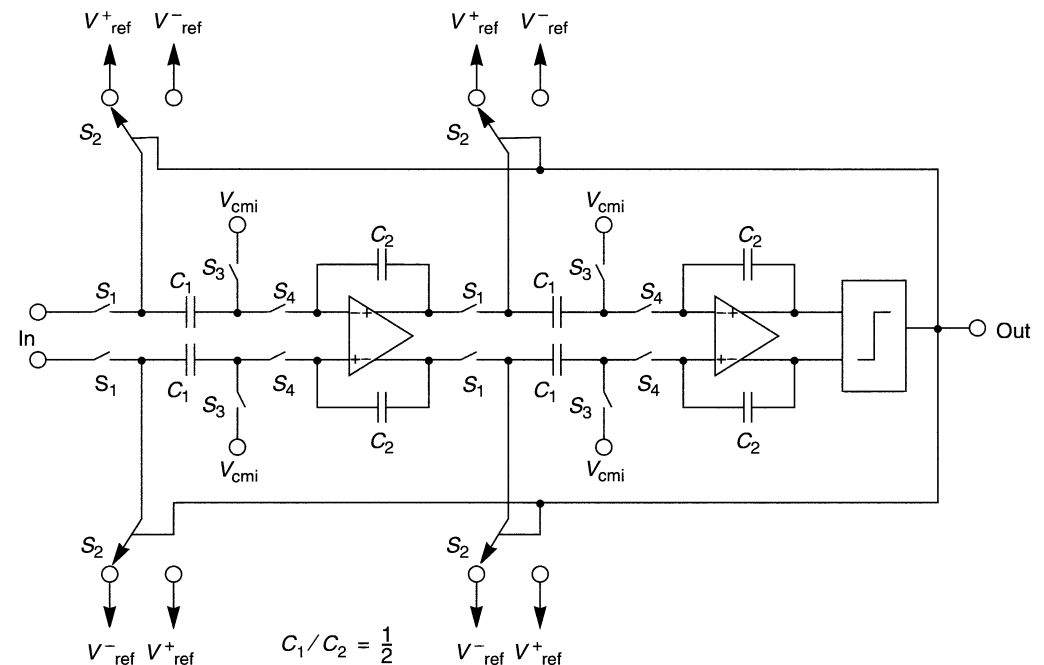


Figure 11.22 Fully differential CMOS implementation of a second-order $\Delta\Sigma$ modulator.

capacitors, C_1 , while the right sides are connected to the desired common-mode input voltage of the operational amplifier, V_{cmi} . Likewise, the differential output of the first integrator is sampled onto capacitors C_1 of the second integrator. The comparator is strobed during phase 1, when the output of the second integrator is not changing.

At the end of phase 1, switches S_3 are opened slightly ahead of switches S_1 to reduce signal-dependent charge injection onto the sampling capacitors C_1 [17]. Since switches S_3 are near the virtual ground of the operational amplifier, and therefore at the same potential, ideally no differential charge is injected onto the sampling capacitors C_1 when they are turned off. When switches S_1 are opened a short time later, the right sides of the sampling capacitors C_1 are floating so that no differential charge from S_1 is injected onto these capacitors.

During phase 2, switches S_4 are closed and the left sides of the sampling capacitors C_1 are connected via the switches S_2 to either V_{ref+} or V_{ref-} , depending on the result of the comparison performed during phase 1. This action performs both the D/A conversion and subtraction functions. As a result, a packet of charge proportional to the difference between each integrator's input and the DAC's output is transferred to the integrating capacitors C_2 . At the end of phase 2, switches S_4 are opened slightly ahead of switches S_2 to isolate the inputs of the operational amplifiers from the differential charge injection introduced by opening S_2 . The comparator is reset during phase 2 in preparation for the next comparison. (Note that this circuit does not use the DAC switching scheme of Figure 11.16, which may be preferable to avoid modulation of the reference voltage.)

According to the switching scheme just described, both integrators in Figure 11.22 include delays of one sample period in their forward paths. Also, both integrators include gain factors of $\frac{1}{2}$ at their inputs that are set by the ratio of their sampling and integrating capacitors, $C_1/C_2 = \frac{1}{2}$. These two modifications of the traditional second-order $\Delta\Sigma$ modulator that is described in [18] reduce the signal range required at the outputs of the integrators to about 1.7 times the modulator's input range [5]. The modifications also allow pipelining in the modulator that reduces the critical path delay to one integrator delay per clock cycle.

While the relative sizes of the sampling and integrating capacitors set the gain factor at the inputs of the integrators, their absolute sizes are chosen on the basis of slew rate and thermal noise considerations. (Matching considerations may also affect this choice.) The size of the capacitors should be minimized so as to reduce the output current that must be provided by the operational amplifier to meet slew rate requirements. Thermal noise introduced by the nonzero resistances of the sampling switches determines the minimum size of the sampling capacitors, which must be large enough to band limit this noise. At an oversampling ratio of 256, 1-pF sampling capacitors (C_1) were found to be large enough to allow greater than 16 bits of resolution in the context of this design. Note that it is possible to use smaller sampling capacitors in the second integrator since noise generated at its input undergoes first-order shaping.

The operational amplifier used in the integrators is the most critical element of the modulator. Incomplete settling of the integrator outputs does not degrade the performance of the modulator, provided that the settling process is linear. However, the settling should not be slew-rate limited. Simulations [19] indicate that a slew rate of 150 V/ μ S is sufficient to meet the performance objectives. Since the comparator can be designed to be quite fast, the settling speed of the integrator ultimately limits the achievable sampling rate of

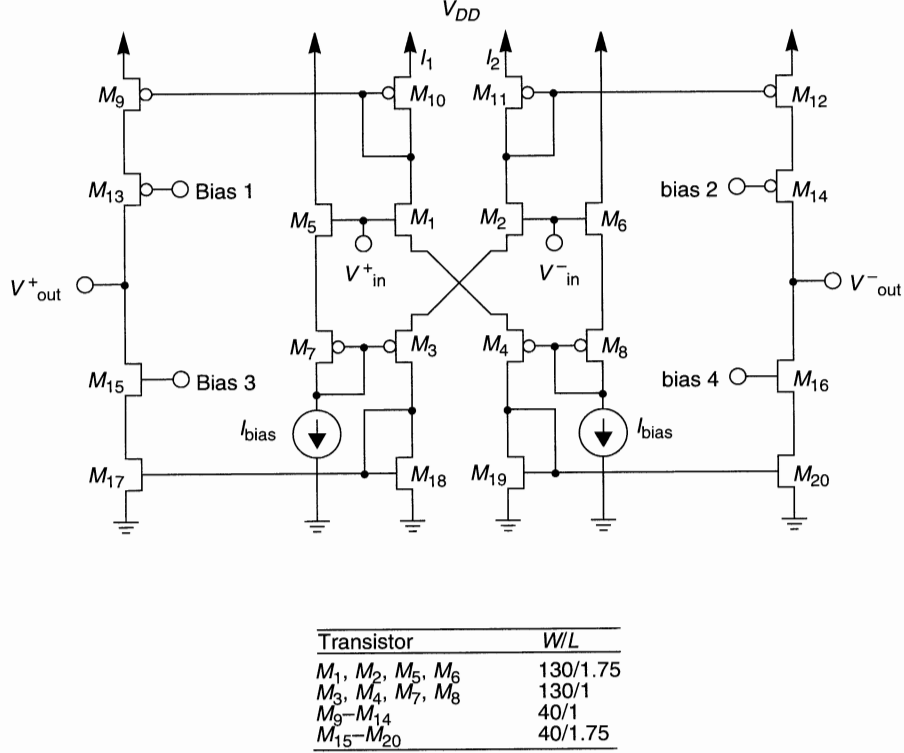


Figure 11.23 Class AB operational amplifier.

the modulator, even if complete settling is not required. The need for high speed, coupled with a relatively modest gain requirement of 60 dB to suppress harmonic distortion, encouraged the use of a single-stage amplifier. The constraint of a single 5-V power supply dictated a low-noise, large-output-swing architecture. To achieve these performance objectives, especially the high slew rate, the class AB operational amplifier shown in Figure 11.23 was chosen here, because of its large output current capability [10]. Unlike amplifiers based on a differential pair, the output current of this amplifier is not limited by a tail current. By dynamically biasing the gates of the cascode output transistors M13–M16, this amplifier provides a large output current and a large output voltage range while maintaining a gain comparable to that of alternative single-stage designs.

In the circuit of Figure 11.23, the four input transistors M1–M4 develop a differential current between the currents I_1 and I_2 . These currents are mirrored to the output branches through the current mirrors M9–M10, M11–M12, M17–M18, and M19–M20. Transistors M5–M8 serve as level shifters and set the quiescent values of I_1 and I_2 equal to I_{bias} . The cascode output transistors M13–M16 increase the output impedance and thus the gain of the amplifier. To meet the conflicting requirements of high output current and large output swing, the gates of these cascode transistors are dynamically biased to ensure that the common-source output transistors M9, M12, M17, and M20 remain saturated. During the initial part of the integration phase (phase 2), the current flowing in the common-source output transistors is necessarily large to provide a high slew rate. The gates of the cascode

transistors must therefore be moved toward $V_{DD}/2$ to keep the common-source transistors saturated. As the integrator outputs settle and currents are reduced toward their quiescent values (thereby increasing the gain of the amplifier), the gates of the cascode transistors are moved away from $V_{DD}/2$ to allow maximum output range. The relatively simple circuitry needed to accomplish this dynamic biasing is presented in [10].

While the use of a class AB amplifier topology is an effective means of preventing slewing distortion, the shift in the amplifier's pole and zero frequencies as currents increase during the initial part of the transient response may limit the settling linearity. In low-power applications, class AB amplifiers are used to achieve low quiescent current levels [10, 20]. In these applications, the ratio of the largest required slewing current to the quiescent current may exceed 45. In the design depicted in Figure 11.23, however, the quiescent bias current I_{bias} was set at a relatively high level of 50 μ A to limit the maximum change in current to a factor of 8.

An additional benefit of the modulator's fully differential architecture is that the common-mode input and output voltages of the operational amplifiers may be set independently to maximize the output current, which is limited by transistors M1–M4 entering the linear region of operation, and by the output range, respectively. The common-mode input voltage is set by the V_{cmi} nodes in Figure 11.22. The common-mode output voltage is set by a switched-capacitor feedback circuitry similar to that presented in Figure 9 of [10].

The second major component of the modulator is the comparator. The performance of the modulator is relatively insensitive to comparator offset and hysteresis since the effects of these impairments are attenuated by the same second-order noise shaping that attenuates the large quantization noise. The regenerative latch shown in Figure 11.24 has been used to implement the comparator [21, 22]. In this latch, the cross-coupled devices M2A–M2B and M3A–M3B are strobed at their drains, rather than sources, to eliminate backgating effects and promote faster regeneration.

The latch is strobed at the beginning of the sampling phase when ϕ_1 transitions from low to high. The latch is reset at the end of the sampling phase and the result of the comparison is stored in an RS latch (not shown). Note that the switching threshold of the output inverters in Figure 11.24 is set low intentionally. In the event that the comparator is unable to come to a decision before ϕ_1 falls, the result of the previous comparison will remain in the RS latch. Also, because no preamplification or offset cancellation circuitry

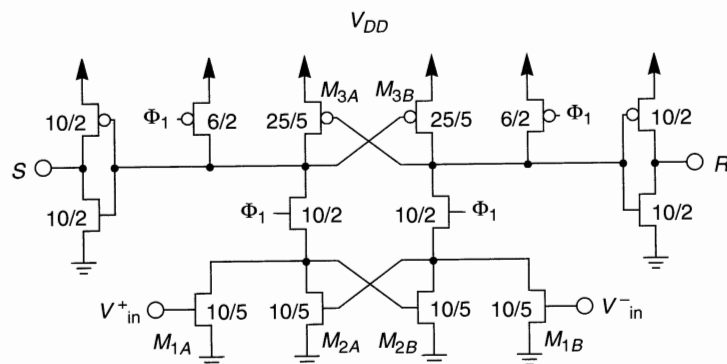


Figure 11.24 Regenerative feedback comparator.

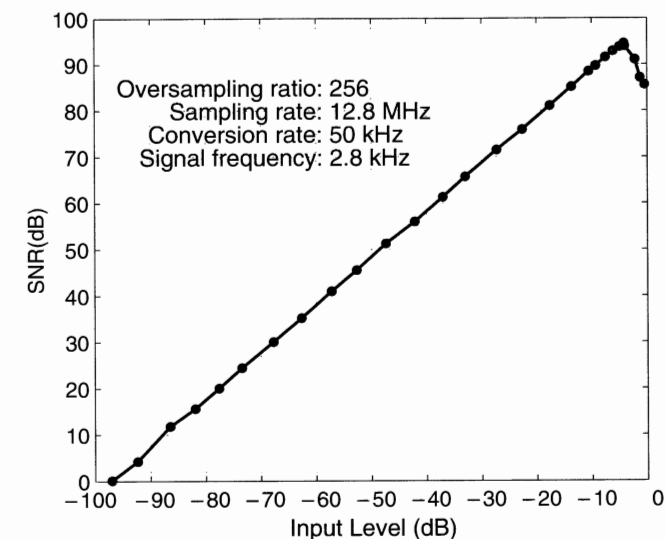


Figure 11.25 Measured signal-to-(noise+distortion) ratio.

precedes this latch, offset voltages were controlled by using nonminimum gate lengths in the input and cross-coupled transistors.

The modulator was fabricated in a 1- μ m CMOS technology with highly linear metal-to-polycide capacitors [23]. The active die area of the modulator was 0.39 mm². Figure 11.25 shows the measured signal-to-(noise+distortion) ratio (SNDR) as a function of the input sine wave amplitude. An input level of 0 dB represents a sine wave whose peak-to-peak amplitude equals the spacing between the two levels of the DAC, which is 4 V (differential) in this implementation. The frequency of the input sine wave was 2.8 kHz and the modulator sampling rate was 12.8 MHz, which produced a 50-kHz Nyquist-rate output and a 23-kHz signal band at an oversampling ratio of 256. At the 12.8-MHz sampling rate, the modulator dissipates 13.8 mW from a single 5-V power supply.

The modulator achieves a 98-dB dynamic range and a peak SNDR of 94 dB. The roll-off in SNDR for input signals larger than -3 dB is due primarily to third-harmonic distortion in the first operational amplifier. As mentioned previously, the settling of the integrator outputs limits the achievable sampling rate of the modulator. Above 12.8 MHz, both the noise floor and distortion increase in the experimental circuit. At 12.8 MHz, the integrators have a period equal to approximately four time constants in which to settle. This represents a need for more complete settling than suggested by simulations based on linear settling and is a result of deviations from ideal exponential settling in the class AB operational amplifiers. The key performance parameters for the subcircuits described in this section are summarized in Table 11.2.

11.8.2 Second-Order Cascaded Modulator (1–1)

As a second design example, a second-order cascaded modulator will be discussed. For the sake of comparison, the design specifications will be the same as for the second-order single-stage modulator presented in Section 11.8.1: a signal bandwidth of 20 kHz

TABLE 11.2 MEASURED SUBCIRCUIT PERFORMANCE SUMMARY

Operational Amplifier	
dc Gain	67 dB
Unity-gain frequency	50 MHz
Slew rate	350 V/ μ s
Linear output range	6 V
Integrator	
Settling time constant	7.25 ns
Comparator	
Offset voltage (rms)	13 mV

and 98 dB dynamic range with single 5-volt supply operation. The theory of cascaded $\Delta\Sigma$ modulators (and some of the design information briefly repeated here) is covered in Chapter 6 of this text, and analog circuit design for $\Delta\Sigma$ ADCs is covered in the preceding part of this chapter. This design example will discuss a second-order cascaded (1–1) modulator employing 1-bit quantizers. This example is presented only for demonstration purposes, since there probably is no good reason for designing such a modulator. The main advantage of using the cascaded structure is that modulators of order greater than 2 can be constructed that are inherently stable. Another advantage of using a cascaded modulator is that multibit quantizers can be used in any loop other than the first without degrading the linearity of the overall converter.

A simple linearized model would predict the same quantization noise at the modulator output for both the second-order single-stage modulator and the second-order cascaded modulator. The output quantization noise will be white noise from the quantizer shaped to second order, or multiplied by $(1 - z^{-1})^2$. In practice, this is generally true, the main deviation from theory occurring at signal levels that are close to full scale. At these signal levels, the behavior of the single-stage and cascaded modulators become markedly different. In the single-stage modulator, the signal at the output of the second integrator becomes very large as the input signal approaches full scale. This will cause the quantizer noise to increase, raising the noise floor. On a plot of SNDR versus signal level, this effect will show up as a degradation in SNDR at higher signal levels. The SNDR will peak at a signal level approximately 3 dB below full scale and then begin to roll off. The second-order cascaded modulator does not exhibit this behavior. The inputs to the quantizers in both loops remain well bounded, even at input signal levels close to full scale. As a result, the in-band quantization noise level is independent of signal, and the SNDR of the converter will increase with increasing signal level all the way to full scale. Therefore, the second-order cascaded modulator will have a theoretical peak SNDR several decibels higher than the equivalent second-order single-stage modulator.

The system block diagram for the proposed second-order cascaded (1–1) modulator is shown in Figure 11.26. Only the quantization error from the first modulator is fed forward and converted by the second modulator. This technique maximizes the dynamic

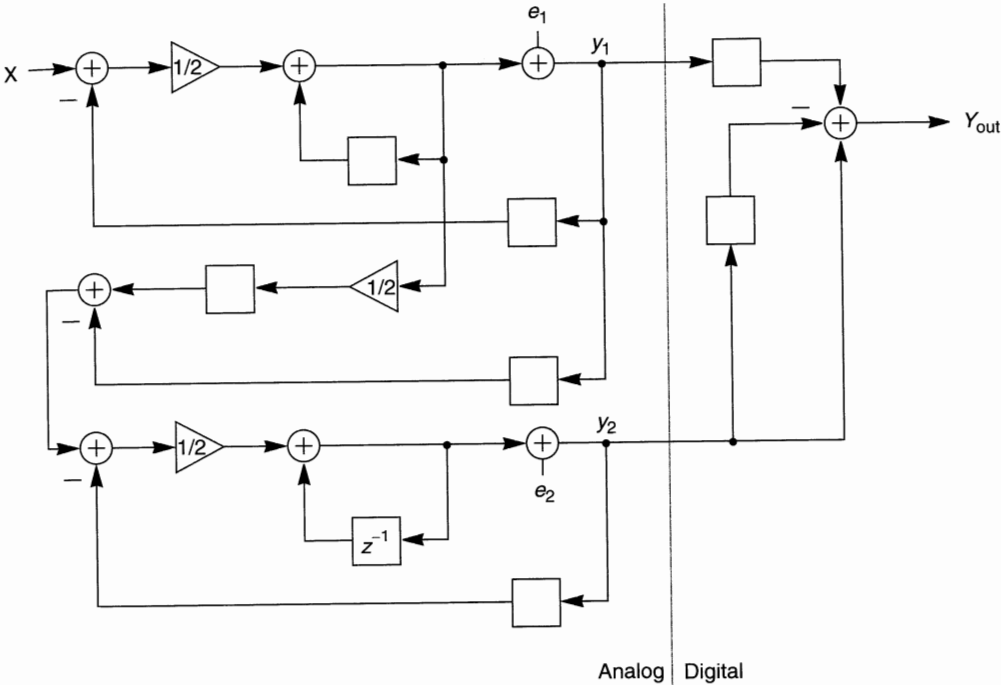


Figure 11.26 System diagram for second-order cascaded (1–1) modulator.

range, since the output of the second modulator is not amplified prior to being recombined with the output of the first modulator (see Section 6.3.1). The SNDR performance of the modulator in Figure 11.26 will be basically the same as for a second-order single-stage modulator, so an oversampling ratio of 256 will be required, as was the case for the example of Section 11.8.1. One of the principal circuit design considerations is to minimize noise and distortion in the first integrator, as is the case for the second-order single-stage modulator. Because of noise shaping, noise and distortion are most important in the first integrator in the first modulator. Noise and distortion from the integrator in the second modulator will be multiplied by $1 - z^{-1}$ when referred to the input of the overall converter. In the discussion that follows, it will be assumed that a fully differential version of the modulator shown in Figure 6.11 is used for the first modulator in the cascade and that a fully differential version of the modulator shown in Figure 6.12 is used for the second modulator in the cascade. The complete schematic for the 1–1 cascaded modulator is depicted in Figure 11.27. It is shown as a single-ended circuit for simplicity, but should actually be constructed as a fully differential circuit. As mentioned previously, fully differential circuits provide better interference rejection, double the allowed signal swing for a given supply voltage, and cause all even-order distortion products to be canceled.

Noise in the first integrator consists mostly of the op-amp's $1/f$ noise and also of thermal noise due to the switches and the op-amp. The $1/f$ noise in the op-amp can be reduced by increasing the size of the noise-contributing devices, since the $1/f$ noise power spectral density is inversely proportional to the gate area of the device. More efficient

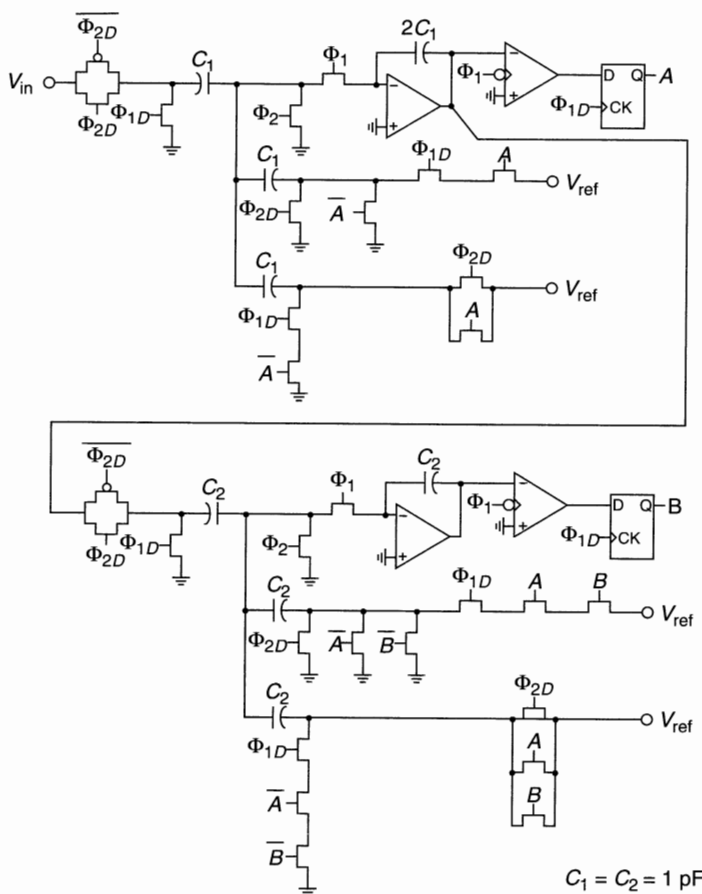


Figure 11.27 Schematic of the analog portion of a second-order cascade (1–1) modulator.

methods for reducing the $1/f$ noise of the op-amp are chopper stabilization and correlated double-sampling. The thermal noise in the integrator is due to the channel resistance of the MOSFETs that make up the switches and the amplifier, but because of the sampling that occurs, the thermal noise power spectral density depends on the size of the capacitors in the circuit (cf. Section 11.4.3). If switch noise is the dominant thermal noise source in the integrator, then the input-referred thermal noise of the modulator is determined by the size of the switched input capacitors. There will be the equivalent of two switched input capacitors for the example shown in Figure 11.27, one for the signal and one for the reference. Since the circuit is actually fully differential, the number of switched input capacitors will double to 4. For an oversampling ratio of 256, and an input capacitor equal to 1 pF, the input referred thermal noise is given by

$$e_T^2 = \frac{8kT}{1 \text{ pF}} + (-24.1 \text{ dB}) = -98.9 \text{ dBV} \quad (11.34)$$

If the full scale input signal is $4 V_{p-p}$, as in Section 11.8.1, the full-scale signal power is 3.0 dBV, so 1-pF input capacitors should be large enough for a 16-bit dynamic range. Note that the thermal noise power is 3 dB higher than that of the modulator in Section 11.8.1, because the lack of a differential reference doubles the number of switched input capacitors. Distortion is also mainly a problem in the first integrator in the first modulator. The summing junction switches are turned off prior to turning off the switches that are connected to the signal input, or the output of an op-amp, to eliminate signal-dependent charge injection. Incomplete linear settling will cause a gain error in the integrator, but incomplete signal-dependent settling will also give rise to distortion. The solution chosen here is to make the circuitry fast enough such that nearly complete settling is achieved. Then even if the settling is signal dependent, the resultant distortion will be small.

All the design considerations mentioned up until this point, regarding noise and distortion, apply equally well to single-stage and cascaded modulators. The main differences between the second-order single-stage modulator and the second-order cascaded (1–1) modulator are as follows:

1. The SNDR of the cascaded modulator will increase with increasing input signal all the way to full scale.
2. The signal ranges at the integrator outputs in the cascaded modulator are well defined.
3. Uncanceled noise from the first modulator in the cascaded (1–1) modulator that leaks to the output has the characteristics of first-order modulator noise. Hence, the cascaded (1–1) modulator has more need of a dither signal than the single-stage modulator.
4. The second-order single-stage modulator is very tolerant to gain and pole errors in the integrators, whereas these same errors in the cascaded (1–1) modulator will cause leakage of first-order shaped and unshaped noise, respectively, to the output of the overall modulator.

Assuming $V_{ref} = 1 \text{ V}$ and an oversampling ratio of 256, the leakage of noise to the modulator output due to the pole error β is given by

$$e_{pe}^2 = -28.9 + 20 \log \beta \quad \text{dBV} \quad (11.35)$$

The leakage of noise to the modulator output due to the gain error α is given by

$$e_{ge}^2 = -71.9 + 20 \log \alpha \quad \text{dBV} \quad (11.36)$$

The quantities α and β were defined in Table 11.1. Pole error is due to the finite gain of the amplifier, and gain error is due to incomplete settling, capacitor mismatch, and finite op-amp gain.

The main requirement on the operational amplifier is speed. An oversampling ratio of 256 with a bandwidth of 20 kHz requires a clock frequency of at least 10.24 MHz. The actual clock frequency will need to be greater than this to allow for a transition band in the low-pass digital filter that follows the modulator. If the clock frequency is 12.8 MHz, then the time slots are 39 ns long minus the nonoverlap time. The requirements of high speed and a reasonable amount of gain dictate the use of a cascode structure. The amplifier shown in either Figure 11.23 or Figure 11.9 could be used for this application, the main

difference being that the amplifier in Figure 11.23 will be more power efficient, since it is able to deliver a greater output current during slewing than its output bias current. If the dc gain of the op-amp is limiting the performance, as may well be the case for the second-order cascaded modulator, then the gain-insensitive integrator shown in Figure 6.9 may be used for the first integrator in the first modulator.

The only requirement of the comparator is that it be fast, since offset, noise, and hysteresis are not particularly important due to noise shaping. A strobed regenerative latch, as might be used for a RAM sense amplifier, will suffice.

Scaling the capacitor values for dynamic range is straightforward for cascaded modulators. If $V_{\text{ref}} = 1$ V, then the single-ended feedback is ± 1 V. If the switched input capacitor is equal to the switched feedback capacitor, then the single-ended input signal is limited to ± 1 V. If it is assumed that the allowed voltage range of each op-amp output is ± 1 V, then the integrating capacitor of the first integrator must be twice as large as the switched input capacitors. This gain of $\frac{1}{2}$ in the first integrator is compensated for by doubling the size of the switched input capacitor to the second modulator. The switched feedback capacitors in the second modulator are also doubled, because they are performing the dual functions of subtracting the comparator output of the first modulator and the comparator output of the second modulator. The feedback in the second modulator can be thought of as a tri-state feedback with values -2 , 0 , $+2$. Note that the capacitors and op-amp in the second modulator do not have to be as large as the corresponding elements in the first modulator. Because of noise shaping, thermal noise in the second modulator is reduced considerably when referred to the input.

11.9 CONCLUSION

This chapter has discussed the analog circuit design techniques appropriate for $\Delta\Sigma$ ADCs. The practical aspects of design, including the minimization of the unavoidable nonideal effects, were emphasized in the discussions. Two different design examples, achieving the same goals, were used to illustrate the circuit techniques described in the chapter.

REFERENCES

- [1] R. Adams, "Design and implementation of an audio 18-bit analog-to-digital converter using oversampling techniques," *J. Audio Eng. Soc.*, vol. 34, pp. 153–166, March 1986.
- [2] U. Roettcher, H. Fiedler, and G. Zimmer, "A compatible CMOS-JFET pulse density modulator for interpolative high-resolution A/D conversion," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 446–452, June 1986.
- [3] R. Koch, B. Heise, F. Eckbauer, E. Englehardt, J. Fisher, and F. Parzefall, "A 12-bit sigma-delta analog-to-digital converter with a 15-MHz clock rate," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 1003–1010, Dec. 1986.
- [4] K. Matsumoto, E. Ishii, K. Yoshitate, K. Amano, and R. Adams, "An 18b oversampling A/D converter for digital audio," in *ISSCC Dig. Tech. Papers, IEEE Int. Solid-State Circuits Conference*, San Francisco, CA, vol. 31, pp. 202–203, Feb. 1988.

- [5] B. Boser and B. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1298–1308, Dec. 1988.
- [6] J. Candy and G. Temes, "Oversampling methods for A/D and D/A conversion," in J. Candy and G. Temes, eds., *Oversampling Delta-Sigma Data Converters*, IEEE Press, New York, 1992.
- [7] T. Choi, R. Kaneshiro, P. Gray, W. Jett, and M. Wilcox, "High-frequency CMOS switched-capacitor filters for communications application," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 652–664, Dec. 1983.
- [8] K. Matsui, T. Matsuura, S. Fukasawa, Y. Izawa, Y. Toba, N. Miyake, and K. Nagasawa, "CMOS video filters using switched capacitor 14-MHz circuits," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1096–1102, Dec. 1985.
- [9] D. Senderowicz and S. Dreyer, "A family of differential NMOS analog circuits for a PCM codec filter chip," *IEEE J. Solid-State Circuits*, vol. SC-17, pp. 1014–1023, Dec. 1982.
- [10] R. Castello and P. Gray, "A high-performance micropower switched-capacitor filter," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1122–1132, Dec. 1985.
- [11] G. Temes, "Finite amplifier gain and bandwidth effects in switched-capacitor filters," *IEEE J. Solid-State Circuits*, vol. SC-15, pp. 358–361, June 1980.
- [12] G. Fischer and G. Moschytz, "On the frequency limitations of SC filters," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 510–518, Aug. 1984.
- [13] R. Bishop, J. Paulos, M. Steer, and S. Ardan, "Table-based simulation of delta-sigma modulators," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 447–451, March 1990.
- [14] Engineering Staff of Analog Devices, Inc., *Analog-Digital Conversion Handbook*, Prentice-Hall, Englewood Cliffs, NJ, 1986, pp. 212–214.
- [15] P. Ferguson, Jr., A. Ganesan, and R. Adams, "An 18 b 20 kHz dual DS A/D converter," in *ISSCC Dig. Tech. Papers*, pp. 68–69, Feb. 1991.
- [16] F. Op't Eynde, G. Yin, and W. Sansen, "A CMOS fourth-order 14 b 500 k-sample/s sigma-delta ADC converter," in *ISSCC Dig. Tech. Papers*, pp. 62–63, Feb. 1991.
- [17] K. Lee and R. Meyer, "Low-distortion switched-capacitor filter design techniques," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1103–1113, Dec. 1985.
- [18] J. Candy, "A use of double integration in sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 249–258, March 1985.
- [19] L. Williams, B. Boser, E. Liu, and B. Wooley, "MIDAS user manual, version 2.0," Integrated Circuits Laboratory, Stanford University, Aug. 1989.
- [20] S. Nadeem, C. Sodini, and H.-S. Lee, "A 1 mW delta-sigma modulator for multichannel applications," in *1993 Symp. VLSI Circuits, Dig. Tech. Papers*, pp. 119–120, May 1993.
- [21] A. Yukawa, "A CMOS 8-bit high speed A/D converter IC," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 775–779, June 1985.
- [22] J. Wu, "High-speed analog-to-digital conversion in CMOS VLSI," Ph.D. Dissertation, Stanford University, March 1988.
- [23] C. Kaya, H. Tigelaar, J. Paterson, M. de Wit, J. Fattaruso, D. Hester, S. Kiriaki, K. Tan, and F. Tsay, "Polycide/metal capacitors for high precision A/D converters," in *IEDM Tech. Digest*, pp. 782–783, Dec. 1988.

Analog Circuit Design for $\Delta\Sigma$ DACs

and discusses the influence the final layout has on the choice of architecture. The chapter concludes in Section 12.4 with two design examples of $\Delta\Sigma$ DACs, illustrating the analog circuit design trade-offs for each converter.

In an effort to limit the scope of the chapter, an emphasis has been placed on single-bit and MASH low-pass switched-capacitor implementations in CMOS VLSI technologies because of the relatively low cost of these technologies and their widespread use in digital signal processing applications. While most of the $\Delta\Sigma$ D/A implementations reported to date have employed these techniques, there has been a noticeable trend toward multibit implementations employing clever techniques to reduce the component matching requirements [1].

12.2 BUILDING BLOCKS

At the system level, $\Delta\Sigma$ ADCs and $\Delta\Sigma$ DACs have a similar structure, the difference lying mainly in which functions are accomplished in the analog domain and which functions are accomplished in the digital domain. The line-up for a $\Delta\Sigma$ ADC consists of a simple antialiasing filter (analog), the $\Delta\Sigma$ modulator (analog), and the decimation/antialiasing filter (digital). For a $\Delta\Sigma$ DAC, the line-up consists of an interpolation filter (digital), the $\Delta\Sigma$ modulator (digital), and the reconstruction filter (analog). In a switched-capacitor implementation, the reconstruction filter will generally consist of several stages of switched-capacitor filters followed by a continuous-time filter to attenuate spectral energy around multiples of the clock frequency. Since the overall converter performance will be limited by the analog circuitry, the modulator represents the greatest design challenge of the ADC, and the reconstruction filter represents the greatest design challenge of the DAC. In general, the design of $\Delta\Sigma$ DACs is more difficult than the design of $\Delta\Sigma$ ADCs, because a low-noise reconstruction filter is more difficult to design than a low-noise analog modulator. The analog reconstruction filter can also take advantage of noise shaping, but the analog modulator has the advantage of being followed by a low-pass digital filter with a sharp cutoff. Clock jitter also has a more deleterious effect on the $\Delta\Sigma$ DAC, as will be discussed later.

Like $\Delta\Sigma$ ADCs, $\Delta\Sigma$ DACs may employ either a single-stage modulator [2, 3] or a cascade of single-stage modulators [4, 5]. A generalized block diagram of a digital $\Delta\Sigma$ modulator is shown in Figure 12.1. This diagram shows a cascade of two loops, although it could be extended to more loops. The loop filter transfer functions $H_1(z)$ and $H_2(z)$ are assumed to have low-pass characteristics, although they could be bandpass transfer functions if a bandpass $\Delta\Sigma$ DAC was being implemented. The L -bit DAC and the M -bit DAC are low-resolution DACs that are used to convert the high-speed digital bit stream into an analog signal. A high-resolution data converter can be realized with low-resolution DACs by making use of noise shaping and oversampling. Mathematically, the signal processing properties of the analog and digital $\Delta\Sigma$ modulators are identical and the principles that apply to $\Delta\Sigma$ ADCs regarding stability and increases in dynamic range with increased modulator order, oversampling ratio, and number of quantizer bits also apply to $\Delta\Sigma$ DACs.

When $H_1(z)$ has a low-pass characteristic, then the analog signal processing performed on the output of the M -bit DAC will have a high-pass characteristic. This high-

12.1 INTRODUCTION

Oversampled DACs have a fundamental advantage over their Nyquist-rate counterparts: The requirements placed on the output anti-imaging filter are greatly reduced, due mainly to the spectral separation of the desired output and unwanted frequency “images” of the output. This benefit has long been recognized and utilized in digital audio systems, where digital interpolation filters were placed in front of conventional DACs to increase the output sampling rate.

Oversampling has additional benefits beyond output filter requirement relaxation, and these benefits have been presented in numerous chapters contained in this book. In the case of oversampled and noise-shaped DACs (in particular $\Delta\Sigma$ DACs), these benefits include tolerance for component mismatch and circuit nonidealities. These benefits allow for implementation of the highly linear and low-noise analog output structures alongside complex digital circuitry.

This chapter will explore the unique aspects of analog circuit design for $\Delta\Sigma$ DACs. As will be seen, the oversampled modulator in these converter architectures relaxes the image rejection and component tolerance requirements on the output antialiasing filter at the price of increased filter speed and step size, primarily due to increased high-frequency quantization noise. Thus, while the anti-imaging requirements of the output filter are greatly reduced, new requirements to remove out-of-band quantization noise are introduced.

Section 12.2 distinguishes the critical and noncritical nodes and components of $\Delta\Sigma$ DACs and focuses on the design of key subcircuits commonly used to implement these converters. Section 12.3 addresses important considerations for the layout of $\Delta\Sigma$ DACs

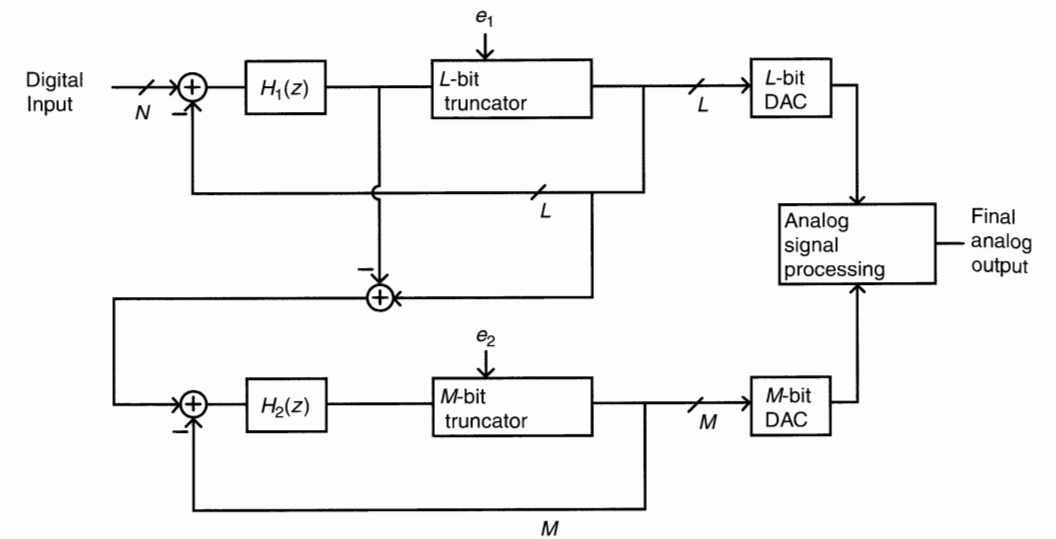


Figure 12.1 Generalized block diagram of $\Delta\Sigma$ DAC.

pass filtering will attenuate in-band spectral components, making the noise and distortion requirements for the M -bit DAC much easier to meet than the corresponding requirements for the L -bit DAC. Care still needs to be taken in the design of the M -bit DAC to make sure that it does not represent a signal-dependent load on the voltage reference. Such a load could cause unwanted signals on the reference. These signals would then be modulated by the data input to the L -bit DAC and fed through to the final analog output without the benefit of high-pass filtering. For a given order of noise shaping and a given oversampling ratio, the dynamic range of the converter can be increased by increasing the number of bits in the low-resolution DAC(s). However, the linearity of the DAC in the first modulator loop must be as good as the linearity of the overall DAC. For this reason, the DAC in the first loop (the L -bit DAC in Figure 12.1) is generally chosen to be a 1-bit one. A 1-bit DAC is inherently linear, since the output has only two levels. Because of the relaxed distortion requirements, the DAC in the second loop can be made with more than 1-bit resolution. Making the DAC in the second loop multibit will improve the performance of the overall converter and lessen the requirements of the analog reconstruction filter (see Chapter 10).

12.2.1 The Low-Resolution Input DAC

One of the differences between $\Delta\Sigma$ ADCs and $\Delta\Sigma$ DACs is in the placement of the low-resolution DAC, as is shown in Figure 12.2. In the analog modulator of a $\Delta\Sigma$ ADC, the low-resolution DAC is in the feedback path and converts the digital output of the modulator to an analog signal. This analog feedback signal is then compared to the analog input signal and the difference fed into the input of the first integrator. In the analog reconstruction filter of a $\Delta\Sigma$ DAC, the low-resolution DAC provides the input to the filter by converting the output of the digital modulator to an analog signal. The analog output of the

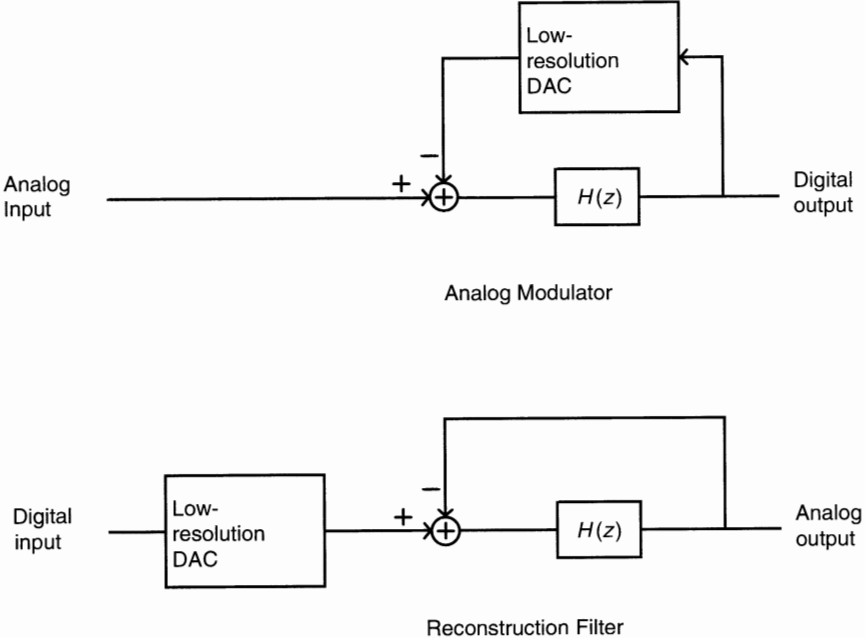


Figure 12.2 Placement of the low-resolution DAC in ADCs and DACs.

reconstruction filter is, in some implementations, fed back and compared to the output of the low-resolution DAC. The difference between these signals forms the input to the first integrator in the filter. By necessity, Figure 12.2 oversimplifies the situation, since in general both the analog modulator and the reconstruction filter may consist of multiple stages and contain multiple feedback and feedforward paths, but it serves to illustrate the basic principles.

An extremely important requirement on the low-resolution DAC is that it provides a constant load on the voltage reference, independent of the signal being converted. Since the output of a DAC is the ratio of the input to the reference, any signal present on the reference will mix with the input signal. Making the load on the reference signal independent ensures that the nonzero output impedance of the voltage reference will not cause signal-related disturbances to appear on the reference. The requirements and problems associated with the low-resolution DAC in the $\Delta\Sigma$ DAC are pretty much the same as for the $\Delta\Sigma$ ADC. A suitable circuit for implementing this function is the one discussed in Chapter 11 and depicted in Figure 11.16. The circuit shown is for a 1-bit DAC, but it can be easily extended to the N -bit case while retaining the signal-independent load property. In addition to disturbances on the reference caused by signal-dependent load variations, unwanted signals may also capacitively or inductively couple onto the reference. A signal that is commonly present on chip that is particularly troublesome if coupled into the reference is a clock at frequency $f_s/2$, where f_s is the high-frequency sampling rate. The presence of a signal at frequency $f_s/2$ on the reference will cause spectral components in the signal that are near $f_s/2$ to mix down into the baseband. The signal will contain large

amounts of quantization noise at frequencies around $f_s/2$. There will also be discrete spectral lines or idle pattern tones centered around $f_s/2$. These idle tones are inherent to the $\Delta\Sigma$ modulator system and will be present even in a software simulation. At frequencies near $f_s/2$, they do not affect the system performance, but nonlinearities in the reconstruction filter or unwanted signals on the references can cause them to mix down into the baseband. The magnitude of the idle tones is dependent on the input signal level, with the tones becoming less prominent at higher input signal levels. The frequency of the tones depends on both the dc level and the frequency of the input signal. Idle tones are more of a problem with $\Delta\Sigma$ DACs than $\Delta\Sigma$ ADCs [6], since the input to the digital modulator can always be expressed as the ratio of two integers, and unlike the analog modulator, the digital system does not benefit from the dithering effect of intrinsic device noise. Hence, it is desirable to use a dither signal with $\Delta\Sigma$ DACs, particularly in audio applications. The dither signal will break up the idle patterns, thereby whitening the noise. The spectral nature of the noise is important: A discrete tone may be audible even if the power of the tone is below the noise floor. Clock jitter in a $\Delta\Sigma$ DAC can also cause unwanted spectra to mix down into the baseband, depending on how much filtering has occurred before the discrete-time to continuous-time interface.

12.2.2 Voltage References

Having discussed the low-resolution DAC that forms the input signal to the analog reconstruction filter, the generation of the reference voltage for the DAC is now discussed. The circuit depicted in Figure 11.16 requires only a single-ended reference, with the two-level feedback being accomplished through the switching action of the capacitors. It is best to generate this reference with respect to the ground plane, which would be connected to V_{SS} in a single-supply circuit. The important parameters for a voltage reference are the absolute accuracy of the generated voltage, the drift of the reference voltage, the noise in the reference, power supply rejection, and the ability of the voltage reference to drive switched-capacitor as well as continuous-time loads. Use of an external reference will give the highest absolute accuracy and lowest drift. However, for applications that do not require extreme absolute accuracy, it is more economical to integrate the reference circuitry on-chip. For many applications of $\Delta\Sigma$ modulation (for example, telecommunications and audio), high absolute accuracy and extremely low drift are not required.

In switched-capacitor circuits, the values of the node voltages are only important at the end of each clock phase. There are two basic approaches to designing the voltage reference. One approach is to make the reference fast enough to completely settle within one switched-capacitor time slot. The reference output will be disturbed when the capacitive load is switched in but will have settled before sampling occurs at the end of the clock phase. The other approach is to provide a constant load to the reference in the form of a large external capacitor connected between the reference output and ground. This results in an incomplete settling of the reference voltage for every switched-capacitor disturbance but also results in a large reduction in the ΔV caused by each disturbance. The very long time constant of this circuit will cause the charge pulses taken from the reference to be averaged, and the switched-capacitor load can be modeled as a resistor of value $1/f_s C$. The second approach is preferable, since it provides good power supply rejection and low

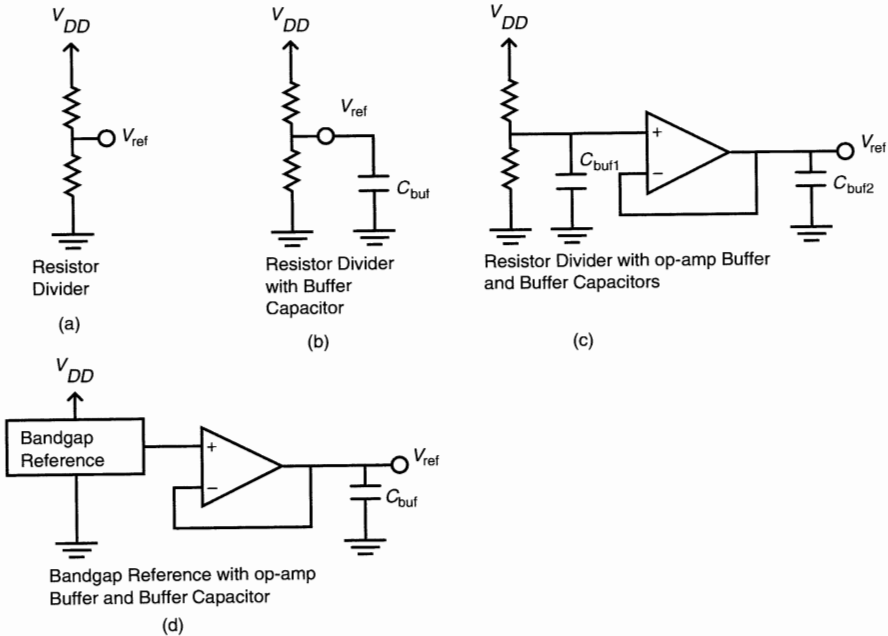


Figure 12.3 Alternative methods for creating a reference voltage.

noise by killing the bandwidth of the circuit. It does, however, require an external capacitor.

Various approaches to creating a reference voltage or an intermediate supply voltage for switched-capacitor circuits are shown in Figure 12.3, in order of increasing complexity. The first approach is simply a resistor divider between V_{DD} and ground. The switched-capacitor load is connected directly to the tap on the divider. When the load is switched in, the reference voltage is disturbed, but if the RC time constant formed by the resistors in the divider and the switched-capacitor load is small enough, the reference voltage will return to its proper value by the end of the time slot. One obvious drawback of this circuit is that it has very poor power supply rejection. An improvement in power supply rejection at higher frequencies can be achieved by placing a large capacitor between the resistor tap and ground. If the capacitor is large enough, the switched capacitor load will appear as a resistor. The reference voltage will remain almost constant, at a value determined by the output impedance of the resistor divider and the equivalent resistance of the switched-capacitor load. A further improvement can be achieved by using an op-amp configured as a voltage follower to provide a lower output impedance. A large capacitor is placed between the follower output and ground to compensate the op-amp and provide load regulation. Depending on the speed of the follower, it may also be desirable to place a capacitor between the resistor divider tap and ground. Since this is a higher impedance point than the output of the follower, more filtering can be provided with a smaller size capacitor. In the final circuit, the resistor divider has been replaced by a bandgap reference. It is not necessary to place an additional capacitor from the follower input

to ground in this circuit, since the bandgap reference should have good power supply rejection from V_{DD} .

The decision as to whether to use a bandgap reference depends on the nature of V_{DD} . The main advantage of a bandgap reference is that it provides an output voltage that is independent of V_{DD} . If the converter is to be sold as a standard part, then it will have to be able to operate with a wide variety of power supplies and a bandgap reference should probably be included. On the other hand, if the part is being designed for a particular application and it is known that V_{DD} is a well-controlled regulated supply, a resistor divider may be sufficient, or even preferable, to a bandgap reference. It should be kept in mind that a well-designed untrimmed bandgap reference in a CMOS process will have a tolerance of $\pm 5\%$ and a drift of 100 ppm/C. Trimming can eliminate the tolerance and reduce the drift, but not without making the part more expensive. Finally, the frequency band of the signal being converted should be considered. A resistor divider with a filter capacitor only provides adequate power supply rejection at frequencies well past its pole frequency.

12.2.3 Reconstruction Filter

12.2.3.1 Specifications. The most important accuracy specifications for a $\Delta\Sigma$ DAC (or $\Delta\Sigma$ ADC) are its dynamic range and $S/(N+D)$ at full scale. Dynamic range is the ratio of the full-scale input signal to the noise floor, usually expressed in decibels. The noise floor is measured with an input signal sufficiently down from full scale such that distortion components are negligible. Thus, the dynamic range is a measure of the noise of the converter. Signal-to-noise+distortion measured at full scale would be equal to the dynamic range if the converter were perfectly linear. In practice this is not the case. The $S/(N+D)$ at full scale will be somewhat less than the dynamic range and will be dominated by distortion. Therefore, the $S/(N+D)$ at full scale is also a measure of the linearity of the converter. Measuring distortion for a $\Delta\Sigma$ DAC is straightforward, but the noise measurement is subject to some “specmanship.” In a $\Delta\Sigma$ ADC what constitutes noise is clearly defined. The output of the ADC is a stream of digital words at a given sampling rate. A number of these words are captured and an FFT performed on the data. The resultant spectrum is periodic in frequency, with a period equal to the sampling rate. In addition, since the signal is real valued, the magnitude spectrum will be symmetrical about zero, so that all the spectral information is contained in a band from zero to half the sampling rate. Therefore, the total noise is obtained by integrating the noise power in the frequency band from zero to half the sampling rate. In a $\Delta\Sigma$ DAC, the bandwidth over which noise should be measured is not as clear. Since the output of a DAC is a continuous-time signal, the spectrum extends from zero to infinity. Theoretically, the total noise power in the output should be measured over an infinite bandwidth. In practice, the bandwidth over which the noise is measured is restricted, with the choice of measurement bandwidth being application dependent.

In the discussions that follow regarding reconstruction filter design, the application will be assumed to be digital audio since this is one of the most common applications for $\Delta\Sigma$ DACs. For digital audio, the measurement bandwidth is usually specified as 0–20 kHz. Additionally, the noise measurement may be made with an A-weighted filter. The A weighting attempts to mimic the response of the human ear and provides substantial

attenuation at frequencies below 100 Hz. These ideal “measurement filters” have an effect similar to the brick-wall digital decimation filter used in a $\Delta\Sigma$ ADC. Restricting the measurement bandwidth allows the use of noise-shaping techniques in the reconstruction filter that reduce the in-band noise at the expense of increasing the out-of-band noise. Since the noise is only being measured in a specific band, it might seem that the highest performance would be obtained by eliminating all analog circuitry and just putting out the high-speed 1-bit digital stream. This is not the case. The digital output would be essentially using V_{DD} as the reference voltage. Therefore, any noise on V_{DD} would modulate the bit stream, causing high-frequency quantization noise to mix back into the band. Also, any mismatch in the rise and fall times of the output driver will give rise to distortion and intermodulation products. At a minimum, the 1-bit digital stream needs to be converted into an analog signal by a high-performance 1-bit DAC using a clean reference. This arrangement would yield the highest performance if the specifications were followed to the letter. However, the specifications usually do not tell the entire story. The output of the DAC will probably drive an audio amplifier. If the D/A output is not sufficiently filtered, then the high-frequency noise components may cause the connected amplifiers to slew, giving rise to intermodulation and harmonic distortion. Highest performance is obtained by placing this additional filtering off-chip, but it is economically advantageous to integrate it on-chip. In addition to the in-band noise specification, it is also appropriate to have a specification for the noise contained in a bandwidth much wider than the signal bandwidth. This out-of-band power specification is rarely given on $\Delta\Sigma$ DAC data sheets.

12.2.3.2 Switched-Capacitor Reconstruction Filter Using Biquads. As was discussed in the previous section, the filtering requirements on the low-resolution DAC output amount to more than just what is needed to meet the noise specification over the measurement bandwidth. The DAC output needs to be sufficiently filtered before being converted to continuous time such that the active circuitry following the DAC does not go into the slewing mode and introduce nonlinearities. In this chapter it is assumed that the reconstruction filter consists of a switched-capacitor filter followed by a continuous-time filter that removes out-of-band energy around multiples of the switched-capacitor clock rate. The more discrete-time filtering is performed, the less deleterious the effects of impairments at the discrete-time/continuous-time interface will be.

As a first example, the switched-capacitor reconstruction filter is chosen as a fourth-order Bessel filter constructed in a straightforward manner using biquads. The clock rate is chosen to be 3.2 MHz and the 3-dB frequency is at 25 kHz. The gain response of this filter will be down 1.851 dB at 20 kHz. In order to have a flat passband, this droop will have to be compensated for in the digital filter that precedes the modulator. A switch-level schematic of the circuit is shown in Figure 12.4. The circuit can be implemented as two biquads [8]. The circuit is drawn single-ended for simplicity but would actually be implemented as a fully differential circuit. The 1-bit DAC shown in Figure 11.16 forms the input to the filter.

It is instructive to examine the effect that noise sources at different points in the circuit have when referred to the output. Considering just a single biquad, Figure 12.5 shows the magnitude of the transfer functions for a noise source at the input of the first biquad and for a noise source at the output of op-amp 1, when referred to the output of the first biquad. A noise source at the input of the biquad, such as the thermal noise from switched capacitors G_1 and C_1 , will be low-pass filtered when referred to the biquad output. Noise

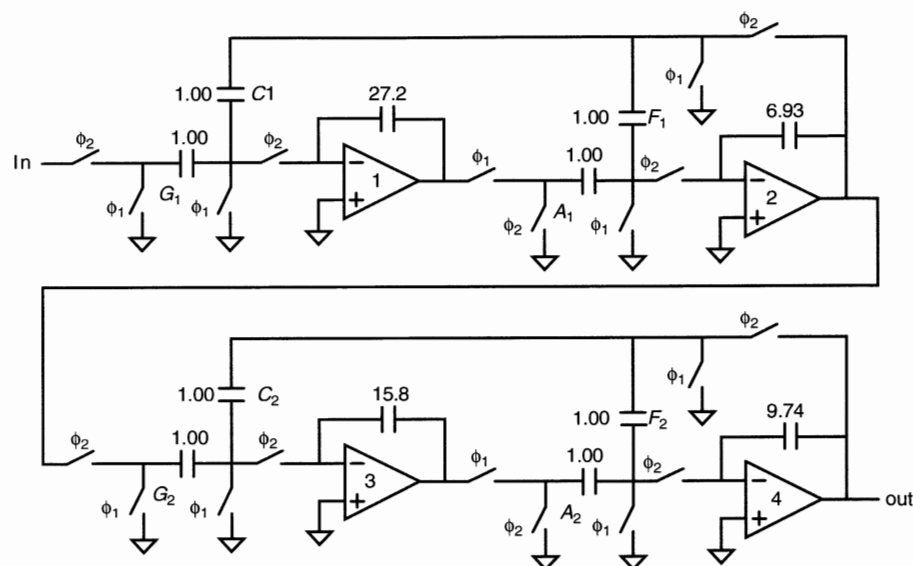


Figure 12.4 Fourth-order Bessel low-pass filter implemented with biquads.

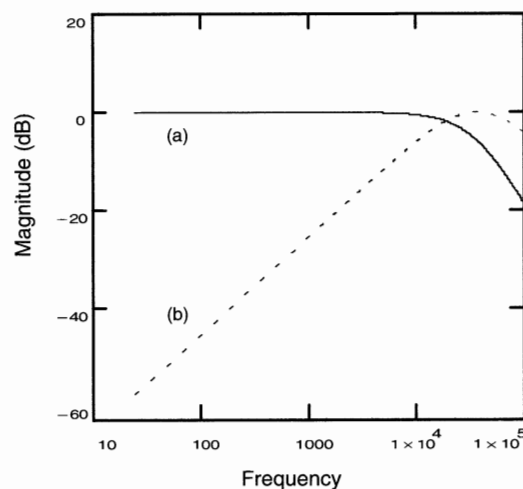


Figure 12.5 Transfer functions of SC biquad from (a) filter input to output and (b) second amplifier input to output.

that occurs after the first stage, such as the thermal noise due to switched-capacitors A_1 and F_1 , will be both high-pass and low-pass filtered. The high-pass filtering occurs because the noise sources are introduced after an integrator. Referring them back to the input involves dividing by the integrator transfer function, or differentiation. From Figure 12.5, it can be seen that a noise source introduced after the first integrator in the biquad will have much lower power spectral density at low frequencies than an equivalent noise source introduced at the input of the biquad. However, the power spectral density of the noise source introduced after the first integrator will rise at 6 dB per octave and eventually become greater than the power spectral density of the input noise source. If capacitors G_1 and A_1 have the same values, then in a 0–100 kHz bandwidth the total noise power at the output of biquad 1 due to A_1 will be 3.44 dB greater than the total noise power due to G_1 . But, in a 0–20 kHz bandwidth, the total noise power due to G_1 will be 4.76 dB greater than the total noise power due to A_1 , while in a 0–10 kHz bandwidth, the total noise power due to G_1 will be 10.34 dB greater than the total noise power due to A_1 . This noise-shaping effect is similar to that observed in the analog modulator of a $\Delta\Sigma$ ADC. The greater the ratio of the filter cutoff frequency to the measurement bandwidth, the less the sources of noise after the first integrator contribute to the total in-band noise. This reduction applies to both switch noise and op-amp noise. Thus, the offset and $1/f$ noise performance of the biquad is determined primarily by the offset and $1/f$ noise of op-amp 1.

The same arguments apply to the second biquad of the filter. Therefore, in this example, the in-band kT/C noise is due mostly to switched capacitors G_1 , C_1 , G_2 , and C_2 . Thermal and $1/f$ op-amp noise is contributed mainly by op-amps 1 and 3. This noise-shaping property will be discussed in greater detail in the next section.

12.2.3.3 Noise-Shaping Filter. A generalized system diagram of a noise-shaping unity-gain low-pass filter is shown in Figure 12.6. The principle behind the filter is that at frequencies in the passband the magnitude of the product $A(z)B(z)$ will be much larger than 1. This high magnitude will cause the overall transfer function in the passband to be determined by just two passive elements: the input switched capacitor and the switched capacitor between the output and the input summing junction. The noise source e_n represents noise that is added in at some intermediate point in the filter. The input-referred e_n is equivalent to dividing it by the magnitude of $A(z)$. At a given frequency, for the contribution of e_n to the total noise to be less than the contribution of an equivalent noise source at the input, the magnitude of $A(z)$ must be greater than 1. But the magnitude

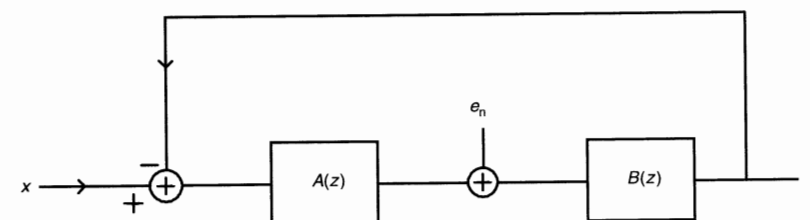


Figure 12.6 General noise-shaping unity-gain low-pass filter, with noise injected at an intermediate point.

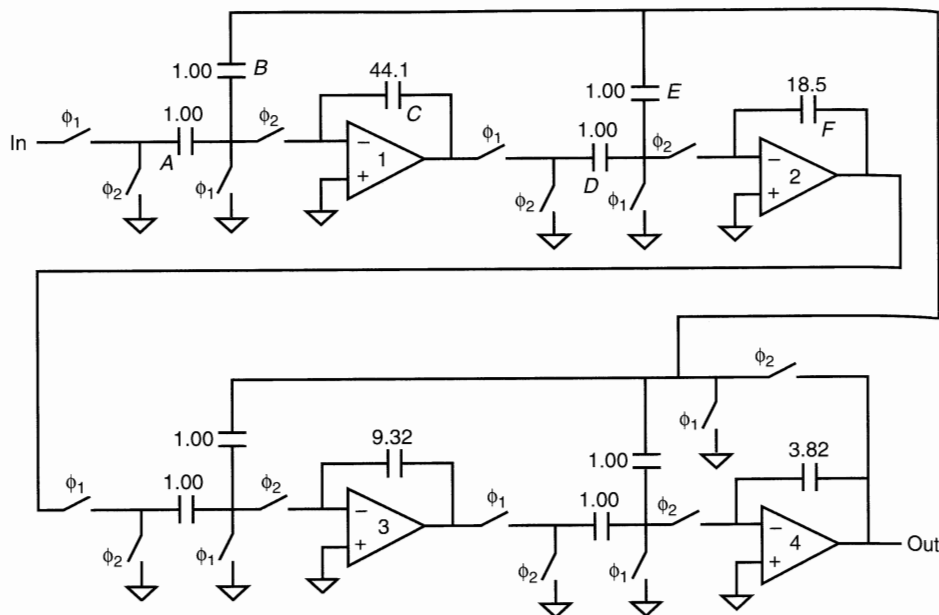


Figure 12.7 Fourth-order Bessel low-pass filter implemented with the IFLF topology.

of $A(z)$ is limited to the inverse of $|x - y|$ because dynamic range considerations dictate that the magnitude of the output of $A(z)$ must be less than 1. For practical low-pass filters, $|x - y|$ will become greater than 1 near the filter cutoff frequency due to the phase shift through the filter. Since the magnitude of $A(z)$ must be less than 1 at these frequencies, e_n will contribute more noise than an equivalent noise source at the filter input. Thus, noise shaping can only be used effectively in a measurement bandwidth that is less than the cutoff frequency of the filter.

A suitable noise-shaping topology for the switched-capacitor reconstruction filter in a $\Delta\Sigma$ DAC is the inverted follow-the-leader feedback (IFLF) configuration [8]. As will be shown below, the output-referred noise from the op-amps and switched capacitors in this topology is dominated by the first-stage components. This property allows for a reduction in the size of the other op-amps and capacitors compared to other topologies. For example, the fourth-order Bessel filter of Figure 12.4 is redesigned using the IFLF topology. A switch-level schematic for the filter is shown in Figure 12.7. Dynamic range scaling for this filter is particularly easy, since the transfer function from any intermediate op-amp output to the final output is also a unity-gain low-pass filter. The in-band kT/C noise is dominated by switched capacitors A and B. The in-band op-amp noise is dominated by the first op-amp. A plot of the output-referred noise transfer functions for noise sources at the inputs of op-amps 1, 2, 3, and 4 is shown in Figure 12.8. This figure shows the spectral shaping of the input-referred op-amp noise away from the passband of the filter, particularly from op-amps 2–4.

12.2.3.4 Analog Decimation Filters for $\Delta\Sigma$ DACs. In switched-capacitor circuits, only the node voltage values at the end of each time slot are significant. There-

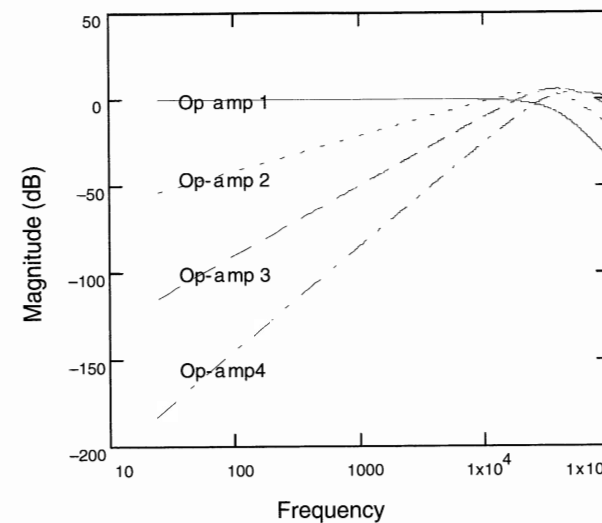


Figure 12.8 Noise gains from various points in the IFLF filter to the output.

fore, if the settling is complete enough, the exact nature of the settling is unimportant. However, when the signal is utilized as a continuous-time signal, the value of the signal is important at all points in time. Any nonlinearities due to slewing or signal-dependent settling will result in a degradation of the overall DAC performance. It is clear that the final switched-capacitor stage should be designed to have as simple and as linear a response as possible. It also is clear that, regardless of the nature of the switched-capacitor transient response, less distortion will be generated if the settling time is small compared to the clock period. This makes it tempting to try and decimate to a lower clock frequency in the switched-capacitor circuitry. Using a lower clock rate may also provide for better capacitor ratios if the ratio of the original clock rate to the poles and zeros being realized is excessively large. However, decimation is not without its dangers. The quantization noise and idle tones present near multiples of the decimated clock rate must be attenuated sufficiently by the decimation filter so that they are not aliased back into the baseband.

A natural choice for the decimation filter is a comb filter, since it provides zeros at integral multiples of the decimated clock frequency. In this case the comb filter will be implemented as an analog finite impulse response (FIR) switched-capacitor filter, as opposed to the digital comb filter used in a $\Delta\Sigma$ ADC. As in an ADC, the order of the comb filter in the DAC should be chosen to be 1 higher than the order of the modulator, so that performance will not be degraded by aliasing of the quantization noise.

A block diagram of a third-order decimate-by-2 switched-capacitor comb filter is shown in Figure 12.9 [9]. The digital $\Delta\Sigma$ modulator outputs 1-bit data at the high-speed clock rate f_s . The data are loaded serially into a 4-bit shift register. At the decimated clock rate $f_s/2$ the data are parallel loaded into another 4-bit register. The 4-bit output of the parallel-load register is used to control the switching of the input capacitors to the switched-capacitor summing amplifier. Prior to the creation of an output sample, the summing amplifier output is reset to zero by a shorting switch. During this reset phase the

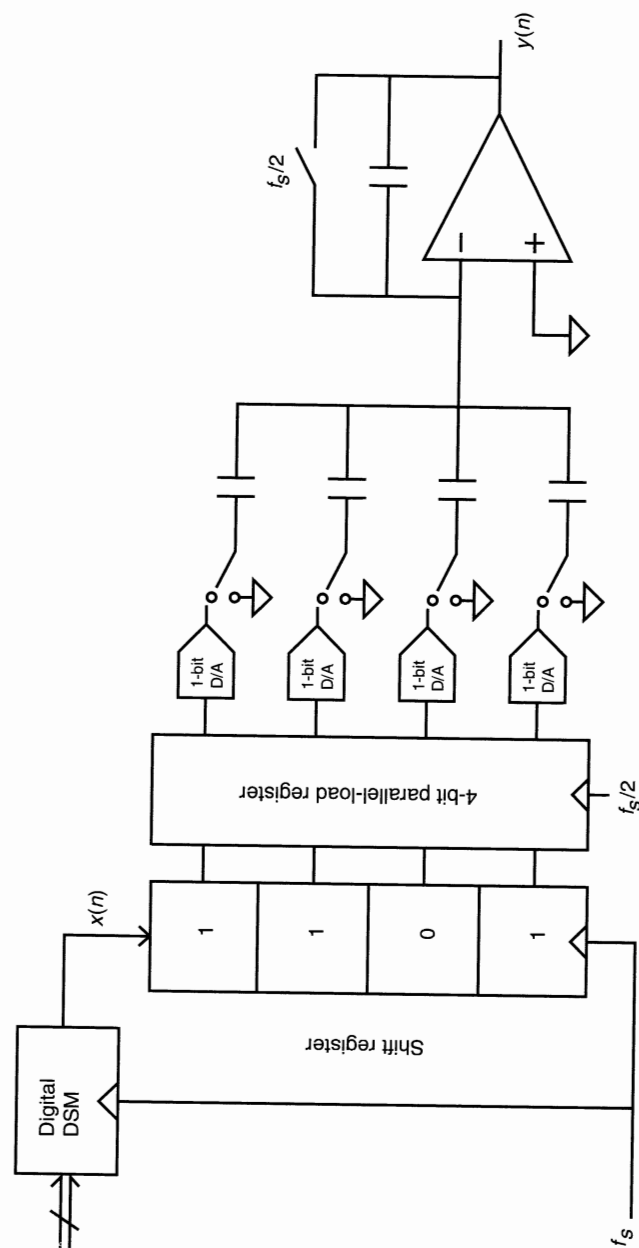


Figure 12.9 Block diagram of a third-order decimate-by-2 switched-capacitor comb filter.

input side of each input capacitor will be switched either to the reference voltage or to ground, the decision being based on whether the corresponding bit in the 4-bit parallel-load register is a 1 or a 0. After the reset switch has been opened, each input that was connected to the reference voltage is switched to ground and vice versa. The net result at the output is a voltage given by

$$V_{\text{out}} = \sum_{i=0}^3 \frac{C_i}{C_f} (V_{\text{ref}}) x_i \quad (12.1)$$

In the above expression, V_{ref} is the reference voltage and $x_i = 1$ if the corresponding data bit $b_i = 1$, whereas $x_i = -1$ if the corresponding data bit $b_i = 0$. Thus, the circuit of Figure 12.9 implements a FIR switched capacitor with the output being updated at a rate $f_s/2$. The coefficients of the FIR switched capacitor are determined by the values of the input capacitors. It can be shown that the operation of the circuit in Figure 12.9 is equivalent to performing a third-order comb filter operation on the input data prior to decimating by 2.

There are a couple of drawbacks to using a FIR switched capacitor as a decimation filter in a $\Delta\Sigma$ D/A reconstruction filter. The first drawback is that the number of input capacitors increases rapidly with increases in the filter order and/or decimation ratio. The second drawback is that proper operation of the decimation filter depends on achieving steep notches in the filter transfer function at multiples of the decimated clock rate by placing zeros at these frequencies. Errors in the sizes of the input capacitors due to tolerance will alter the placement of the zeros, which will increase the amount of noise aliased down into the baseband during the decimation process. The requirements on capacitor matching become more severe for higher performance DACs.

The number of input capacitors for a given comb filter can be reduced by approximately a factor of 2 by taking advantage of the symmetry of the FIR coefficients. Filter coefficients spaced equal distances from the center of the impulse response will be of equal value. If the number of coefficients is odd, there will also be a coefficient directly at the center of the impulse response that has no corresponding equal-valued coefficient. Any pair of equal-valued coefficients may be represented by a single capacitor. This is accomplished by combining the 1-bit data corresponding to the tap positions of that pair of coefficients and using the result to control the switching of the capacitor. Since each capacitor now represents 2 bits of data, the switching sequence must be capable of implementing the values +2, 0, and -2 instead of just +1 and -1. The +2 and -2 are realized in the same manner as the +1 and -1, but with the size of the input capacitor doubled with respect to the feedback capacitor. The 0 is realized by including an additional switching sequence where the input side of the input capacitor is connected to either V_{ref} or ground during both the reset and output phases of the summing amplifier.

There is another modification that can be made to the circuit of Figure 12.9 that reduces the number of input capacitors required and also reduces the sensitivity of the notch depth at multiples of the decimated clock rate to capacitor matching. This modification is accomplished by realizing one of the orders of the comb filter as a first-order prefilter by summing together n consecutive outputs of the FIR filter, where n is the decimation ratio. An additional amplifier stage is not required. The circuit is basically the same as

before, except that the summing junction side of the input capacitors are switched between ground and the summing junction instead of being permanently connected to the summing junction. The summing amplifier is reset at the decimated rate, but the switched-capacitor FIR outputs are generated at the fast rate f_s . Between resets, a number of consecutive FIR output samples equal to the decimation ratio are added together. The final output at the decimated rate is valid during the last time slot before the summing amplifier is reset and must be sampled by the following stage during this time slot. The implementation of a switched-capacitor decimation filter that realizes a first-order comb filter response by summing together consecutive samples is described in [10]. This implementation gives zeros at multiples of the decimated clock rate independent of capacitor ratios. Implementing one stage of a high-order comb filter in this manner makes the amount of noise aliased back into the baseband from the decimation process less sensitive to capacitor matching.

12.2.3.5 Effect of Nonideal Integrator Transfer Function. The physical causes for the deviation of a switched-capacitor integrator transfer function from the ideal are discussed in detail in Section 11.4.1 of this text, so that discussion will not be repeated here. In this section some qualitative observations will be made regarding the effect of various integrator nonidealities on the overall DAC performance. A switched-capacitor reconstruction filter will be assumed, and it will also be assumed that only the values of the node voltages at the end of a time slot are important. Issues concerning the discrete-time/continuous-time interface are discussed in the next section of this chapter.

Since the modulator in a $\Delta\Sigma$ DAC is digital, the only place where nonideal integrators can affect the converter performance is in the reconstruction filter. The effect of any integrator impairment on the reconstruction filter performance depends on the structure of the reconstruction filter and the placement of the integrator within the filter. The noise-shaping topology of Figure 12.7 will be used as an example in this discussion, but the general principles are also applicable to other topologies.

As described in Section 11.4.1, finite op-amp gain will cause gain and pole errors in the integrator transfer function, and nonzero switch resistance and finite op-amp bandwidth will give rise to gain errors [cf. Eqs. (11.20)–(11.22)]. Settling errors may be linear (signal independent) or nonlinear (signal dependent). In addition to incomplete settling, nonlinearities in the integrator transfer function may also be caused by a nonlinear op-amp transfer function, signal-dependent switch charge injection, or voltage-dependent capacitors.

As is the case for the analog modulator in the $\Delta\Sigma$ ADC, linear errors in the D/A reconstruction filter are much less of a problem than nonlinear errors. Linear errors due to finite op-amp gain or incomplete settling will have very little effect on the performance of the circuit shown in Figure 12.7. If the digital modulator is of the cascaded type and the quantization noise cancellation is accomplished through analog differentiation and summation utilizing switched-capacitor circuits, then a gain error due to capacitor mismatch, finite op-amp gain, or incomplete settling will result in a leakage of quantization noise to the overall DAC output. This is the same effect that was described for cascaded $\Delta\Sigma$ ADCs in Chapter 11. There is an important difference, however, between the performance of cascaded $\Delta\Sigma$ ADCs and cascaded $\Delta\Sigma$ DACs with regard to phase error. Finite op-amp gain will cause the pole of the switched-capacitor integrator in the analog modulator of a $\Delta\Sigma$ ADC to be displaced from $z = 1$, resulting in a leakage of quantization noise to

the converter output. The situation in the analog differentiator of a cascaded $\Delta\Sigma$ DAC is much different. In that case, the zero of the differentiator will be exactly at $z = 1$, independent of the characteristics of the op-amp. Thus, the critical phase error is not a concern in cascaded $\Delta\Sigma$ DACs.

As mentioned previously, nonlinear errors are much more detrimental to the filter performance than linear errors. The effect of a particular nonlinearity on the reconstruction filter performance will depend on the point in the filter where the nonlinearity occurs. Again referring to Figure 12.7, assume that the input capacitor A is realizing the 1-bit DAC by having its input side switched between V_{ref} and ground, the phasing of the switching being determined by whether ± 1 is being fed to the filter. Then it can be seen that A may be a nonlinear voltage-dependent capacitor without degrading the linearity of the filter. This is because there are only two possible values of input, so the process is guaranteed to be linear. However, if capacitor B is nonlinear, then distortion will be generated at the filter output. At low frequencies, the open-loop gain of the first switched-capacitor integrator is very large. When the filter loop is closed around this integrator, the net low-frequency charge flowing into the integrator capacitor C is very close to zero. This implies that the charge fed back from the filter output through capacitor B must cancel the ± 1 charge pulses fed in through capacitor A . If capacitor B is voltage dependent, then the output of the filter will be distorted in a manner that causes the charge fed back through capacitor B to balance the charge fed in through capacitor A . Nonlinearities in the first op-amp and capacitors D , E , and F will also cause distortion at the filter output. The distortion caused by these components, however, will be reduced at lower in-band frequencies by the noise shaping that occurs. Distortion introduced after the first integrator will undergo a differentiation when referred to the filter input. Similarly, nonlinearities introduced after the second and third integrators will have even less of an effect on the filter performance.

12.2.4 Discrete-Time/Continuous-Time Interface

12.2.4.1 Final Discrete-Time Stage. Switched-capacitor circuits depend on nearly complete settling during each time slot for correct operation. Internal to the circuit, the values of the node voltages are only important at the end of each time slot. The manner in which a node voltage changes from the end of one time slot to the next is only important at the final output, where the output voltage is treated as a continuous-time signal, and its value at all points in time is important. In a switched-capacitor reconstruction filter, the nature of the settling is only important in the last switched-capacitor stage before the continuous-time interface. This stage should be switched so that it does not settle in series with any other stages, the settling response should be linear, and the settling time should be as short as possible.

In order to satisfy the above stated requirements, the last switched-capacitor stage should be a single op-amp stage, and the switching configuration should be such that no previous stages are directly connected to the last stage while its output is changing. A circuit that meets these constraints is the noninverting switched-capacitor low-pass stage shown in Figure 12.10. The circuit operation may be improved by placing linear resistors (polysilicon, for example) in series with the switched capacitors. The use of resistors will

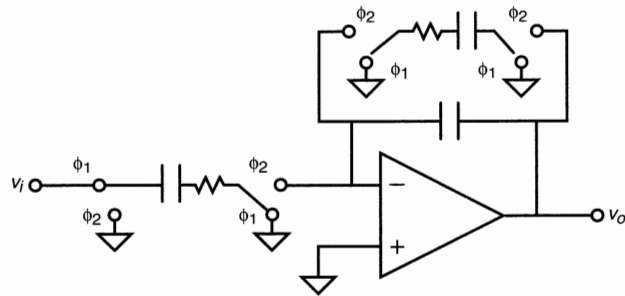


Figure 12.10 Switched-capacitor low-pass stage with antislew resistors.

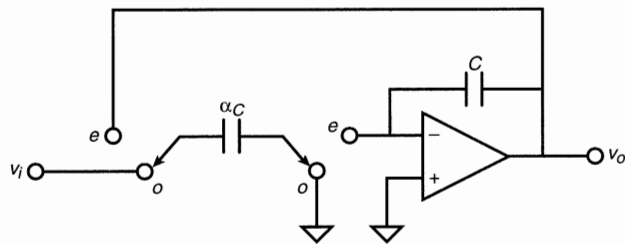


Figure 12.11 Direct charge transfer low-pass filter.

cause the settling speed to be limited by the RC time constant of the switched capacitors, so the op-amp will not go into a slewing mode, thereby preventing nonlinearities from being introduced. The same effect could be accomplished by using switches with smaller W_s , but then the settling behavior would be dominated by the resistance of the switches, which is voltage dependent and would also introduce nonlinearities. Using linear resistors in series with the capacitors allows the response of the circuit to be slowed down without introducing significant nonlinearities.

Another good choice for the final switched-capacitor stage is the direct charge transfer low-pass filter shown in Figure 12.11 [11]. In this configuration, the same capacitor is used as both the switched input capacitor and the switched feedback capacitor. The transfer function for this filter is given by

$$H(z) = \frac{\alpha z^{1/2}}{1 + \alpha - z^{-1}} \quad (12.2)$$

The response of this circuit is fast and linear. Since the input capacitor directly deposits the charge on the integrating capacitor, then, theoretically, the op-amp does not have to provide charging current for these capacitors.

12.2.4.2 Continuous-Time Reconstruction Filter. In a $\Delta\Sigma$ DAC, the switched-capacitor reconstruction filter is generally followed by a continuous-time low-pass reconstruction filter. The purpose of the continuous-time filter is to remove spectral energy around the high-frequency sampling rate f_s . The main requirement of the contin-

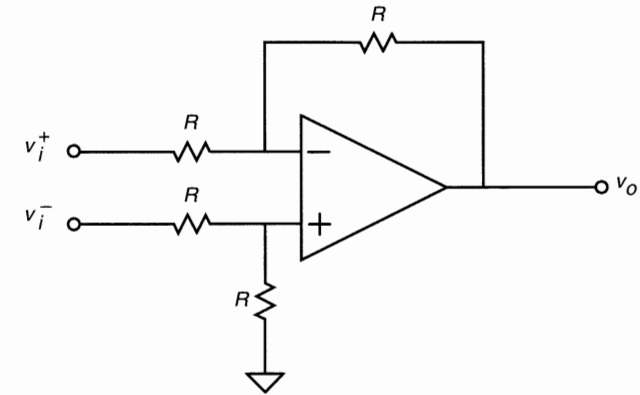


Figure 12.12 Differential to single-ended conversion circuit.

uous-time filter is that it provide sufficient attenuation near f_s but not affect the passband. Since the sampling rate is at a much higher frequency than the switched-capacitor filter cutoff frequency, the tolerance on the poles and zeros of the continuous-time filter does not have to be very tight. Therefore, this filter may be implemented on-chip with polysilicon resistors and double-poly capacitors. For very low distortion, the filter topology chosen should use one op-amp per pole, with the inputs of each op-amp both remaining at the analog ground potential. If the inputs of the op-amps move with the signal, parasitic capacitance connected to the inputs will charge and discharge, causing distortion at the filter output.

In order to obtain the highest $S/(N+D)$, fully differential circuits should be used wherever possible in both the switched-capacitor and continuous-time filters. It is preferable to have a fully differential output from the chip. In some cases, system constraints dictate a single-ended output. In these cases, a differential-to-single-ended conversion circuit constructed from a single-ended op-amp and resistors is commonly used (Figure 12.12). Another approach is to only use fully differential circuitry and just use one of the differential outputs. In this latter case, any common-mode signal that is not canceled by the common-mode feedback circuit will show up in the output. The common-mode feedback op-amp will therefore need to perform to the same stringent specifications as the overall DAC.

12.2.5 Other Nonideal Effects

12.2.5.1 Intrinsic Noise. As was explained in Chapter 11, intrinsic noise refers to noise that is generated in the device itself, as opposed to noise that couples in from an interfering source. The two most important noise mechanisms in MOS devices are thermal noise and $1/f$ noise. Due to the high gate impedance of an MOS device, the gate current noise is negligible, and the overall noise can be modeled as a voltage noise source in series with the gate. Thermal noise is important in both the op-amps and switches in a switched-capacitor circuit, and $1/f$ noise is important in the op-amps.

Intrinsic noise in switched-capacitor circuits was discussed in Section 11.4.3, so that discussion will not be repeated here. The most important point to be made is that the effect

of any noise source on the final reconstruction filter output is dependent on the point in the circuit where the noise source is introduced. The reconstruction filter examples given in Section 12.2.3 serve to illustrate this point.

12.2.5.2 Dynamic Range Considerations. As with any active filter, the reconstruction filter needs to be scaled properly for dynamic range in order to maximize the signal-to-noise performance. When the filter has been scaled to achieve the maximum dynamic range, the peak gain over all frequencies from the filter input to the output of any op-amp in the filter will be the same. By scaling the filter in this manner, the voltage excursion at the output of each op-amp is maximized, which in turn optimizes the signal-to-noise performance of the filter. In a switched-capacitor circuit, the signal level at the output of any op-amp can be scaled down by an arbitrary factor by scaling up all the feedback capacitors associated with that op-amp, as well as any nonfeedback capacitors that the op-amp drives, by the same factor. This scaling will alter the signal swing at that particular amplifier output without changing the overall filter transfer function (unless, of course, the output of that amplifier happens to be the filter output). When implementing dynamic range scaling, filter blocks cannot be scaled independently. When scaling the output of an op-amp, all previous filtering must be taken into account.

After adjusting all capacitor values for dynamic range, a further scaling may be necessary to minimize the spread in capacitor values. If all the input capacitors and feedback capacitors associated with an op-amp are scaled by the same factor, then both the overall transfer function and the voltage swing at the output of that op-amp remain unchanged. It is obviously desirable to reduce the capacitor spread, since large capacitors take up a lot of area, and the minimum capacitor size is limited by tolerance and noise considerations.

12.3 LAYOUT CONSIDERATIONS

12.3.1 Differences from the ADC

As mentioned previously, the analog portion of a $\Delta\Sigma$ DAC generally consists of a discrete-time filtering stage to reduce the quantization noise energy, a discrete-to-continuous-time interface, and a continuous-time filter stage to reduce images found at the discrete/continuous-time interface. The vast majority of fully integrated $\Delta\Sigma$ DACs use switched-capacitor circuitry for the discrete-time filtering, and the layout techniques for these types of filters were discussed in Section 11.7. It is important to note that since we are filtering a digital bit stream to produce an analog output, noise injected at frequencies in the stopband of the analog filters will be attenuated along with the quantization noise. However, low-pass sampled-data filters do not provide any attenuation at multiples of the sample rate, so special care must be taken with layout-coupled noise sources at these frequencies. This is one of the motivations for adding a continuous-time filter stage at the output of the signal path. The RF coupling effects mentioned in Section 11.7 can cause high-frequency interference. Supply current spikes from a digital interpolation filter or $\Delta\Sigma$ modulator can also cause high-frequency interference with signal-dependent sidebands that can produce in-band signal distortion.

Another way to view this is to note the fact that switched-capacitor circuits are sampling circuits that can alias signals from near multiples of the sampling frequency to near DC. This is particularly troublesome for the sampled input voltage source in a $\Delta\Sigma$ DAC: namely, the reference voltage (see Section 11.7 for discussions of this signal). Since the output of a DAC is equal to the digital input *multiplied* by the voltage reference, signals on the reference will be modulated with signals carried in the bit stream. This means that digital input signals coupled into the reference will cause second-harmonic distortion, and spectral noise at $f_s/2$ coupled into the reference will cause signals near $f_s/2$ in the bit-stream (primarily quantization noise and idle tones) to be folded down to dc.

12.3.2 Layout Influences on the Architecture

As mentioned earlier in this chapter, the architecture chosen for the switched-capacitor filter can be influenced by layout considerations. For example, Figure 12.13 shows a third-order switched-capacitor filter designed as a switched-capacitor biquad followed by a single low-pass filter stage. This architecture provides three points into which low-frequency noise can be injected and passed unattenuated to the output (nodes A, B, and C in Figure 12.13).

The coupled biquad filter shown in Figure 12.14 can be designed to reduce the noise coupled from one of these nodes (node B) to the output by providing in-band attenuation from that node to the output. This structure can also be designed with a smaller total capacitance for the same kT/C noise budget by using the proper noise-shaping filtering between intermediate stages and the output [12]. This property was also used in $\Delta\Sigma$ ADCs to shape quantization noise in the loop and to reduce op-amp sizes, capacitor sizes, and parasitic layout coupling effects in stages closer to the comparator. The property was also extensively discussed earlier in this chapter, where an inverse follow-the-leader structure was proposed. It is brought up again in this section to point out how layout constraints can influence the choice of filter architecture.

12.4 DESIGN EXAMPLES

A graphical description of a generalized $\Delta\Sigma$ DAC is shown in Figure 12.15. This figure shows plots of amplitude versus frequency and time of signals at various points in the architecture, including the digital input, the output of an interpolation-by-64 and filtering stage, the output of a 1-bit $\Delta\Sigma$ modulator, and the final output after analog filtering. Chapter 13 describes the design of the interpolator, Chapter 10 describes different digital modulator structures, and the next section of this chapter will describe the implementation of the analog filters for two designs: a 1-bit high-order $\Delta\Sigma$ DAC and a MASH DAC.

12.4.1 A 1-Bit High-Order $\Delta\Sigma$ DAC

The first design, most prevalent in $\Delta\Sigma$ codecs for computer audio (such as the AD1848), incorporates a digital 1-bit high-order $\Delta\Sigma$ modulator for its noise-shaping loop. The 1-bit output stream is passed to the analog portion for conversion to an analog voltage. The spectrum of the 1-bit stream contains shaped quantization noise and unfiltered or

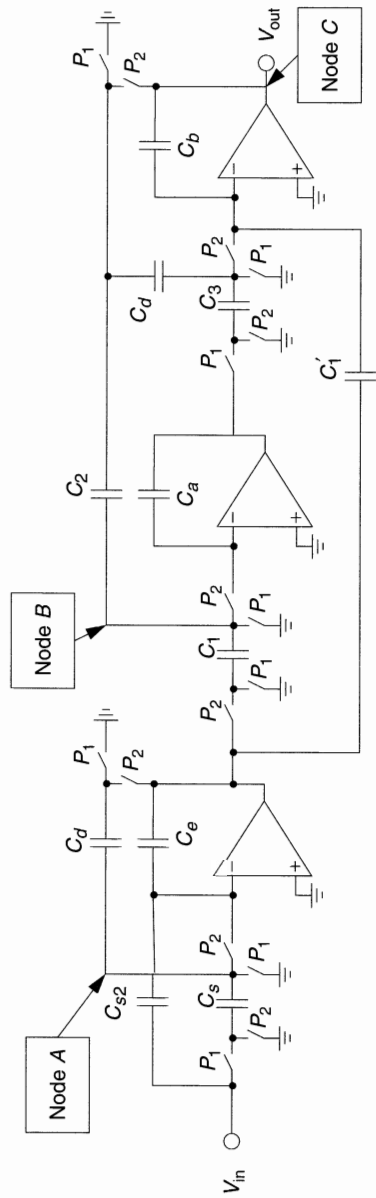


Figure 12.13 Third-order switched-capacitor filter: cascaded biquad sections approach.

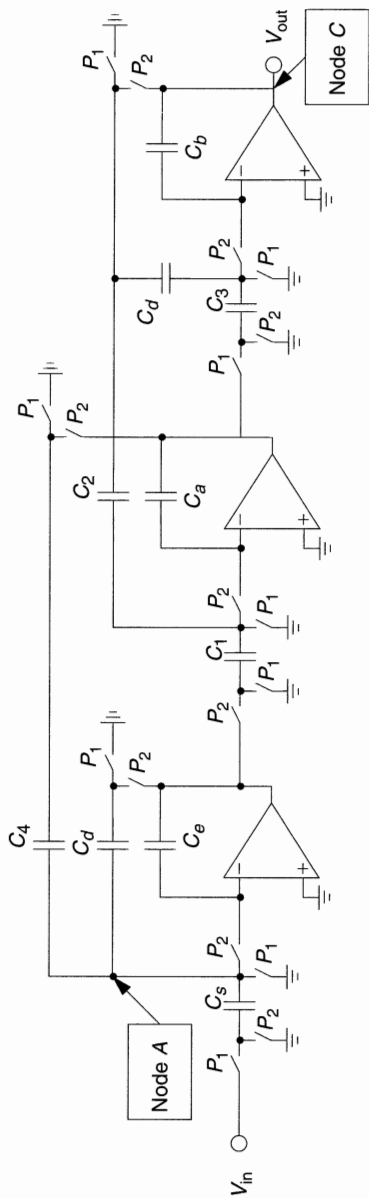


Figure 12.14 Third-order switched-capacitor filter: coupled biquad sections approach. (Incorrect clock phases are shown for simplicity; the circuit must be implemented differently for full functionality.)

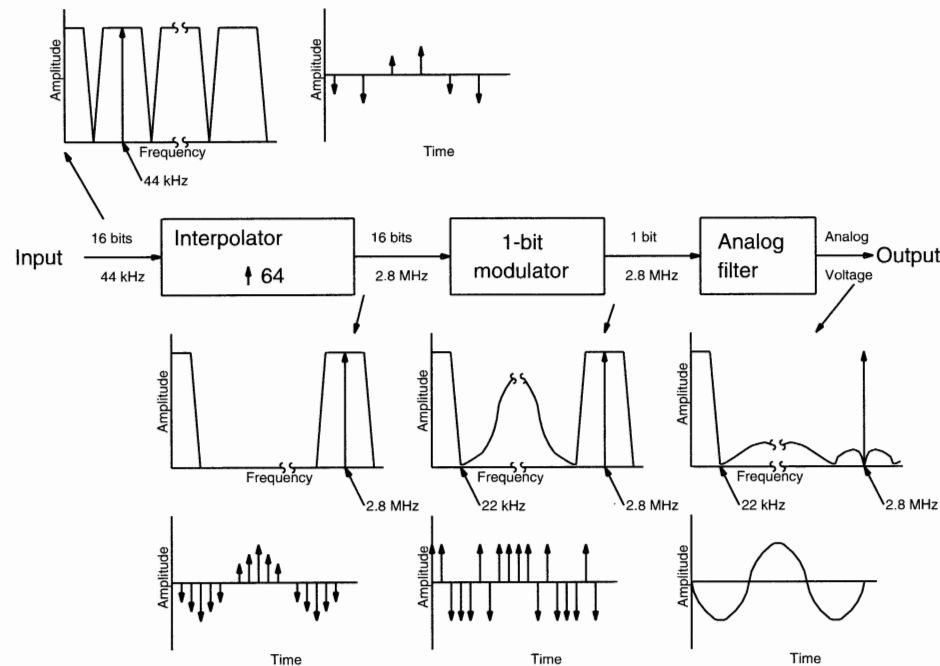


Figure 12.15 Generalized architecture of a $\Delta\Sigma$ DAC showing amplitude versus frequency and time at various points in the architecture.

inadequately filtered spectral images of the input signal, which need to be reduced or removed before the signal can be passed to the output pin.

The first analog design decision is, where should the discrete-to-continuous-time interface occur? The answer to this question lies in the understanding of the effects of clock jitter. Clock jitter at the discrete-to-continuous-time interface causes errors in the time placement of sample transitions at that interface. The samples are generated under the assumption that the reconstruction clock will be perfectly periodic. Jitter causes nonidealities because it violates this basic assumption.

Figure 12.16 demonstrates that the jitter-induced error in a reconstructed waveform is proportional to the time placement error of the clock edge (Δt) and the voltage step between samples (ΔV). The time placement error can be reduced by providing low-jitter clocks to this interface, but this can be very difficult if, for example, the clocks are generated using a phase-locked loop. The voltage step can be reduced by ensuring that the highest frequency content of the signal at the discrete/continuous-time interface is much lower than the clock frequency.

In the case of a 1-bit DAC, the magnitude of the voltage steps at the DAC output is the difference between the voltage at the high and low states of the output, or V_{ref} . Thus, the error generated at every output transition is $\Delta t V_{ref}$. If we remove all shaped quantization noise from the output signal before the interface, however, we can reduce the maximum Δv to be only that required by the slope dV/dt of the input signal. Since the system is oversampled, this improvement can result in a large reduction in the amount of the jitter-induced error.

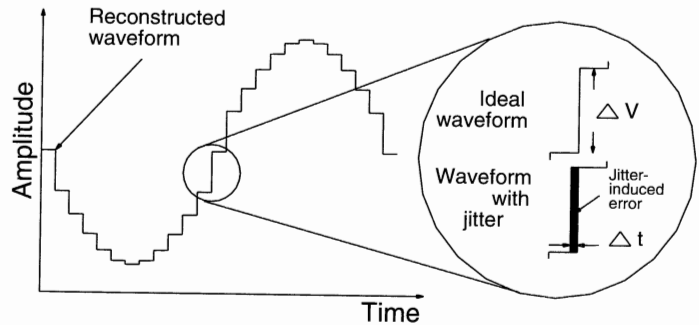


Figure 12.16 Effects of clock jitter on a reconstructed waveform from a DAC output; Δt is the time jitter and ΔV is the voltage step between two selected samples.

Therefore, we try to do as much filtering as possible in the discrete-time domain to remove the high-frequency noise and images before we convert to continuous time. This filtering can become somewhat expensive, as a fairly high order filter is usually required to do an adequate job on a 1-bit signal. This has led to the recent popularity of digital $\Delta\Sigma$ modulation schemes that either use multibit quantizers or some form of feedforward quantization noise cancellation. These structures typically give up the advantage of the inherent linearity of the 1-bit DAC but use other techniques to linearize the output.

Clock jitter can still be a problem, even in the presence of a brick-wall sampled-data filter, since the signal at the continuous-time output of this stage still contains filtered images at multiples of the sample rate (seen in the time domain as small steps from one output sample to the next). Therefore, some continuous-time filtering must still be applied before the signal can be passed to the output. Since the oversampling ratio is high (64 in our example), these images tend to be quite small due to the filtering action of the implicit sample-and-hold function at the discrete/continuous-time interface. Therefore, first-order continuous-time filtering is usually sufficient.

Now that the discrete-to-continuous-time interface has been architecturally placed, we can go about designing the rest of the circuit. To do this, we need to determine what order of filtering is required, based on the specifications for out-of-band noise (noise at frequencies outside the band of interest), the image attenuation of the interpolation filter, and the amount and shaping of the quantization noise. In our example, these factors combine to give us a filter order of 4: three orders of discrete-time filtering and one order of continuous-time filtering.

The structure chosen for the discrete-time filter was the coupled biquad structure for reasons stated earlier in this chapter. For this design, the coupled biquad structure had the smallest capacitors and lowest coefficient sensitivity of the structures evaluated, including the straight biquadratic design approach and the inverse follow-the-leader feedback structure presented in this chapter. The first two stages are standard switched-capacitor filter stages, and the 1-bit DAC at the filter input is the one shown in Figure 11.16.

The discrete/continuous-time interface occurs at the output of the third op-amp. Most of the high-frequency noise has been removed at this point in the circuit. However, the

output of this stage still traces voltage steps due to the frequency content of the input signal and the discrete-time nature of the sampled-data system. These steps must be accurately produced, meaning that the voltage transitions must be made in a linear fashion with minimal overshoot and glitch. This means that the voltage steps produced at the op-amp output should not result in the op-amp overloading into a slew-limited condition and the glitches caused by sampling the output of this stage by feedback paths in the switched-capacitor circuit must be minimized.

Several designs were presented earlier in this chapter, and some of the choices are reiterated here along with some different options. One approach to guarantee linear settling involves increasing the linear input voltage range of the op-amp by reducing the g_m of the MOS input pair. Reducing the g_m can compromise the op-amp gain and input-referred thermal noise but is usually the most economical approach. A second design technique [13] involves placing a large capacitor across the inputs of the op-amp to absorb the charge and minimize the steps seen across the op-amp inputs. This technique will also increase the high-frequency gain from the op-amp input-referred noise. A third design technique [14] involves using a switched-capacitor low-pass filtering buffer network that transfers the charge from the input switched capacitor, to the integrator capacitor *passively* (Figure 12.11). The amplifier does not need to charge the switched capacitor, and therefore it only needs to charge the output load capacitance, which can be minimal if the circuit is followed by a continuous-time filtering stage. This third alternative, discussed previously in this chapter, is the most complex but may be suitable for very high performance applications.

12.4.2 A MASH DAC

A second design example [14] uses the largest selling $\Delta\Sigma$ DAC architecture in the world. It is the MASH DAC that is found in most Japanese CD players and is very cost competitive since all of the circuits are integrated onto a single all-digital chip, with the analog signal processing all done off-chip.

The MASH architecture was covered in Chapter 6. The DAC counterpart is identical, except that the problems of matching analog integration with digital differentiation disappear if the designer combines the outputs of the first and second $\Delta\Sigma$ loops digitally, as discussed earlier in this chapter. This digital output combination will produce a multibit digital output at a high rate (typically 3.072 MHz) that needs to be converted into an analog waveform.

There are several ways to maintain the linearity of a multibit DAC, as discussed in Chapter 10. This design example converts the multibit data into pulse-width-modulated (PWM) single-bit data using a parallel-to-serial converter, a retiming circuit, and a 1-bit DAC made from CMOS inverters. The differential PWM output circuit configuration is shown in Figure 12.17 [14]. The basic principle is to convert the multibit signal into a pulse, with a width proportional to the code, using a clock faster than that of the multibit data rate. Time slots in the faster clock are used to force the width of the pulse to be proportional to the multibit code: If a 3-bit code is produced (codes 0–7) and code 4 is desired, for example, then the $1/(3.072 \text{ MHz})$ clock period is divided up into eight equal intervals and the logic output is high for intervals 0–4 and low for intervals 5–7. As the code changes, the width of the pulse changes accordingly.

Steven R. Norsworthy
Ronald E. Crochiere

Decimation and Interpolation for $\Delta\Sigma$ Conversion

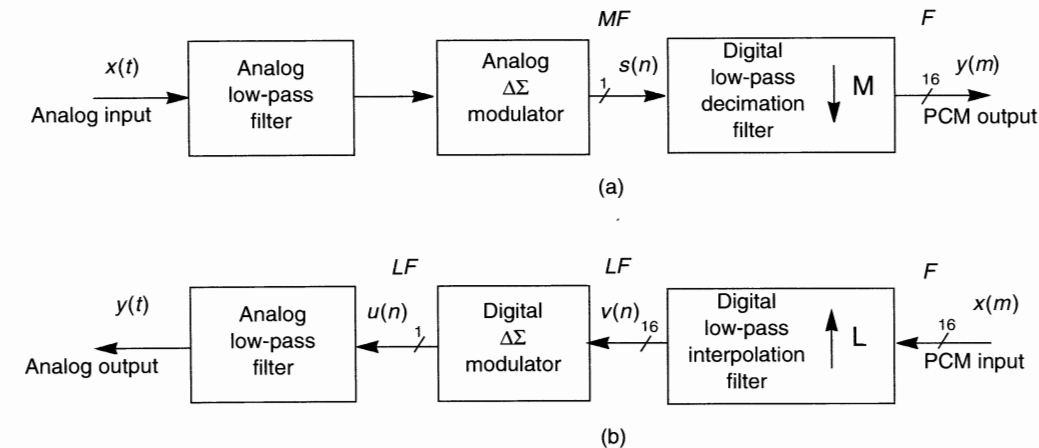


Figure 13.1 (a) A $\Delta\Sigma$ A/D converter system; (b) $\Delta\Sigma$ D/A converter system.

a signal $u(n)$ that can be represented by 1 bit (or a few number of bits) per sample. This signal is then converted to the desired analog output $y(t)$ by means of an analog low-pass filter. The same arguments for design efficiency that apply to the above A/D process also apply to this reverse approach to D/A conversion.

An important aspect of this design concept is the manner in which the quantization noise in the signal $s(n)$ is spectrally distributed by the $\Delta\Sigma$ modulator. Because the number of bits per sample is extremely low, the total quantization noise energy in $s(n)$ is very high. However, by careful design of the noise feedback mechanism in the $\Delta\Sigma$ modulator, as discussed in previous chapters and reference [1], this quantization noise is minimized in the baseband frequency region of the signal, that is, the spectrum from zero to half the final PCM sampling frequency. It is then left to the decimator to filter out the unwanted noise in the spectrum above the Nyquist band so that it is not aliased into the baseband by the decimation process. In a similar manner in the $\Delta\Sigma$ D/A converter, the intermediate signal $u(n)$ may be allowed to have considerable out-of-band noise energy that is removed by the analog filters.

In this chapter, we examine basic filter architectures and hardware implementation methods that can be effectively applied in the design of the above decimators and interpolators. We begin by discussing the overall scope of design trade-offs associated with the design of decimators and interpolators. This is followed by a discussion of basic principles of digital sampling rate conversion, multistage conversion, and the duality relationship between the two processes [2]. We then discuss issues of traditional designs such as comb, or sinc^K , filter designs and filters with reduced coefficient word lengths (0's and 1's). We also discuss filter types such as linear versus minimum-phase designs and FIR versus IIR design options as applied to A/D conversion. Next, we will discuss system implications of sampling rate conversion combined with quantization noise models for oversampled signals to achieve high-performance A/D conversion performance. We will then address signal processing architectures and architectural issues for implementing multirate and multistage sampling rate conversion. Examples will be given based on two important applications: digital audio (sampling rate of $F = 44.1$ kHz) and voiceband telephony

As seen in previous chapters of this book, the techniques for oversampled $\Delta\Sigma$ data conversion are heavily based on principles of sampling rate conversion and the efficient implementation of architectures for changing the sampling rate. In this chapter, we will review classical principles of sampling rate conversion and hardware design and discuss how they apply to modern $\Delta\Sigma$ conversion techniques.

The basic concept of a $\Delta\Sigma$ A/D converter for converting an analog input signal $x(t)$ into a Nyquist-sampled digital PCM signal $y(m)$ is illustrated in Figure 13.1(a). A $\Delta\Sigma$ modulator converts $x(t)$ into a highly oversampled digital signal $s(n)$ with 1 bit (or a small number of bits) per sample. Because the number of bits in this signal is small, the accuracy requirements on the A/D front end are considerably reduced and traded for circuit speed. Furthermore, the high degree of oversampling of the input signal eliminates the need for expensive analog antialiasing filters. The signal $s(n)$ is then digitally converted to the desired PCM sampling rate through a digital smoothing and resampling process called decimation, where F refers to the desired PCM sampling rate and M is the oversampling ratio. These three factors—elimination of analog antialiasing filters, low-accuracy circuit requirements, and digital signal processing for sample rate conversion—contribute to making this design approach particularly attractive for implementation in high-performance CMOS VLSI.

A similar argument can be made for the reverse process of converting a Nyquist-sampled PCM signal $x(m)$ into an analog signal $y(t)$, as illustrated in Figure 13.1(b). In this case, the PCM input $x(m)$ is first converted, through an interpolation process, to an oversampled signal $v(n)$ (oversampled by a factor of L), then $\Delta\Sigma$ modulated to produce

(sampling rate of $F = 8$ kHz). Finally, we discuss hardware issues and implementation techniques that combine the above theoretical concepts with practical considerations of high-performance mixed analog and digital VLSI technology.

13.2 SCOPE OF DESIGN TRADE-OFFS AND ALTERNATIVES

Our focus in this chapter is on issues of how to develop efficient VLSI designs of $\Delta\Sigma$ converters described by Figure 13.1. One very important characteristic of all good VLSI architectures is that they make efficient use of the available technology within the context of the application requirements. Unfortunately, there is no cohesive theory that relates all the different architectural possibilities. Several basic questions arise: What is the space of possible designs? How do we characterize a given task? How do we measure performances and cost for a given design and a given task? How do we relate this to silicon area, power dissipation, MIPS, memory size, signal-to-noise ratio, and so on? The availability of application-specific computer-aided design (CAD) tools usually has a big effect on the architecture and design style chosen. Commercial pressures and schedules often force designers to make many compromises, often resulting in an architecture that is not ideal. Therefore, how do we trade off an *optimum* architecture against a *flexible* architecture, that is, one that is more easily programmable and/or reconfigurable? We have many design alternatives, but the interrelationships between these parameters are so complex that we are left unable to quantify these important questions in the most general sense. Only by very careful examination of a specific set of design objectives can we then proceed with an intelligent choice of architecture.

Table 13.1 summarizes many of the important design trade-offs one must consider when determining how to best map a digital filtering algorithm onto a VLSI chip. It shows that chip performance is a function of algorithm, architecture, layout, and technology. In this chapter we focus mainly on the first two columns, namely, issues associated with *algorithm efficiency* and *architectural efficiency*. It should be noted that the choice of architecture is also strongly driven by the improvements and strides made in the other two columns, namely, better *technology* and *layout efficiency*. Advancements in technology and layout are more dynamic and rapidly changing over time, such that there is a latency associated with exploiting these advancements in next-generation designs. However, advancements in algorithms and architectures are much more slowly changing over time. As a result, a new design is likely to incorporate algorithms or architectures that are relatively efficient with respect to the most current art. Sections 13.3–13.5 cover topics primarily associated with algorithm efficiency. Sections 13.6 and 13.7 focus primarily on architecture efficiency with some minimal discussion of issues related to technology and layout.

13.3 BASIC PRINCIPLES OF SAMPLING RATE CONVERSION—
ALGORITHM ISSUES

The basic principles of sampling rate conversion are based on the concepts of the Nyquist sampling theorem and digital filtering theory. In principle, this theory states that the minimum sampling frequency of a signal must be greater than twice that of the highest

TABLE 13.1 DESIGN TRADE-OFFS: DIMENSIONS AND OPTIONS FOR VLSI DECIMATORS AND INTERPOLATORS

Chip Performance (Speed, Area, Power, . . . , etc.) Function of: 	More Rapidly Changing over Time 			
	Algorithm Efficiency (Behavioral)	Architectural Efficiency (Structural)	Layout Efficiency (Physical)	Technology (Physical)
Design choices and options	• FIR vs. IIR • Single stage vs. multistage • Direct form vs. polyphase • vs. transpose vs. parallel • Coefficient word lengths • Hardware vs. software	• Distributed vs. centralized arithmetic • Bit parallel vs. bit serial • vs. digit serial • Hard wired vs. reconfigurable vs. programmable • Random vs. structured (e.g., PLA vs. ROM)	• Programmable logic • Gate array • Standard cell • Full custom	• CMOS • BICMOS • Bipolar • GaAs
Associated trade-offs	• Accuracy vs. speed and performance • Area vs. performance • Testability • Design time	• Performance vs. area vs. power dissipation • Testability • Flexibility • Design time	• Design time and cost vs. performance and area	• Speed vs. power vs. complexity vs. cost

PLA, programmable logic array; ROM, read-only memory.

frequency of interest that we wish to represent in the signal. All signal energy above this frequency must be filtered out of the signal to avoid undesired aliasing in the sampling process. Similarly, in the reconstruction process, all signal energy at frequencies above the Nyquist baseband (i.e., above half the sampling rate) must be removed (filtered) to eliminate the effects of imaging or signal replication in the output analog signal. These principles must also be applied when converting a signal between two digital sampling rates in order to avoid undesirable effects of aliasing or imaging.

13.3.1 Decimation by M

Figure 13.2 illustrates the basic principle of reducing the sampling rate F of a signal $x(n)$ by an integer factor M . The signal $x(n)$ is first digitally filtered by a low-pass filter $h(n)$ with a digital cutoff frequency of π/M , where π is the normalized (radian) frequency

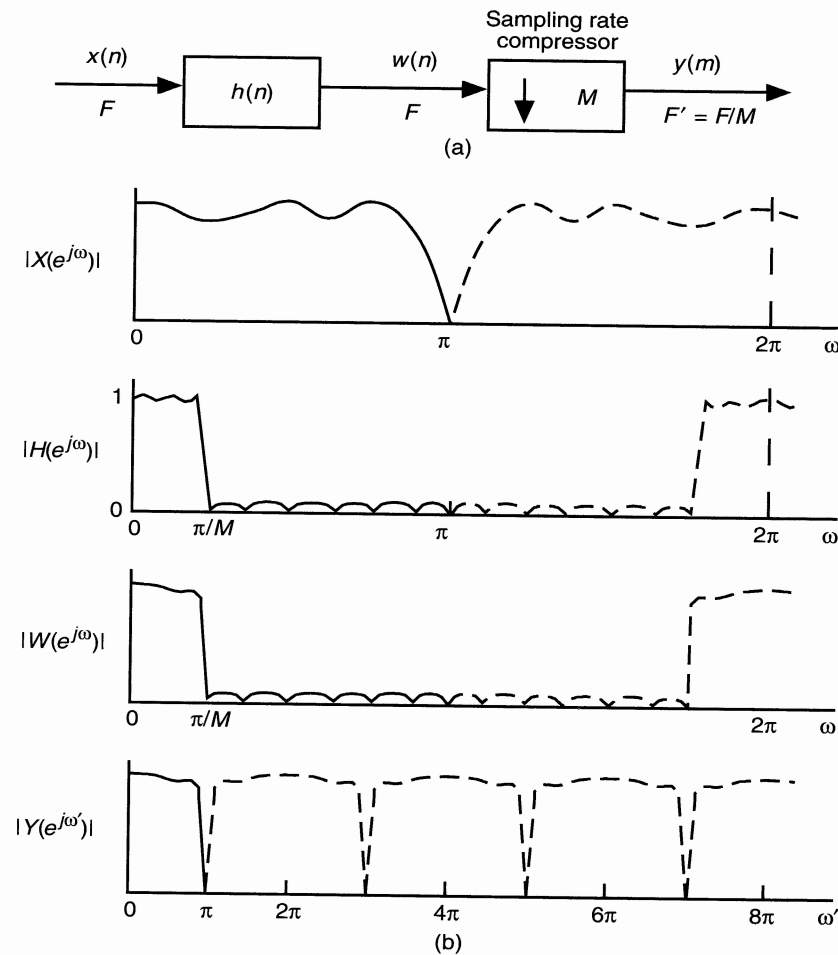


Figure 13.2 (a) Block diagram and (b) typical spectra for decimation by an integer factor M .

corresponding to half the sampling frequency of $x(n)$. The purpose of this filter is to remove all signal energy in $x(n)$ above π/M to avoid aliasing in the decimation process. It is assumed in this process that only the low-frequency information in $x(n)$ is important. The band-limited signal $w(n)$ can then be resampled by discarding $M - 1$ out of every M samples (sampling rate compression) to produce the output $y(m)$ at a rate F/M that is free of aliasing. In practice, this process is typically performed by computing only one out of every M outputs of the digital filter. The digital filter, while specified at the high sampling rate F , is actually implemented at the low rate F/M . This is shown by the relationship

$$y(m) = \sum_{k=-\infty}^{\infty} h(k)x(Mm - k) \quad (13.1)$$

This equation has the form of a convolution in which the input signal $x(n)$ is shifted by M samples for each new computed output. Because of this shift, it is seen that this system is no longer a time-invariant system but instead a periodically time-varying system.

13.3.2 Interpolation by L

The reverse process of interpolating a signal by an integer factor L is illustrated, in principle, by the block diagram in Figure 13.3(a). The input signal $x(n)$, sampled at a rate F , is first increased in sampling rate by inserting $L - 1$ zero-valued samples between each sample of $x(n)$. The intermediate signal $w(m)$ at a rate LF contains the desired baseband information of $x(n)$, but it also has $L - 1$ imaged replications of this spectrum at frequencies above π/L that are undesired. Figure 13.3(b)–(d) illustrate time- and spectral-domain interpretations for these signals for the case $L = 3$. These images are removed by a low-pass filter. In the time domain, this filter effectively interpolates the zero-valued samples to their desired values. The input-to-output relation for this process can be shown to be [2]

$$y(m) = \sum_{k=-\infty}^{\infty} h(m - kL)x(k) \quad (13.2)$$

This equation also has a form similar to a convolution in which only one out of every L samples of the filter $h(n)$ is used in the computation of a given output; that is, multiplication by the zero-valued samples of $w(n)$ are not included in this equation. As in the decimator, this is a periodically time-varying system in which different values of $h(n)$ are periodically used for the computation of each output. Although the filter is specified at the high sampling rate $F' = LF$, in practice it is implemented in a decimated fashion at the low sampling rate F for efficiency.

13.3.3 Duality

The processes of decimation and interpolation are in effect duals. The modulation or mapping of signal energy called aliasing in the decimation process is a many-to-one mapping that can be eliminated by filtering. In the interpolation process, when $M = L$ as defined above, the reverse modulation or mapping of signal energy called imaging is a one-to-many mapping that is the *transpose*, or *dual process*, to that of decimation and can similarly be eliminated by filtering. A filter defined for one process can often be used for

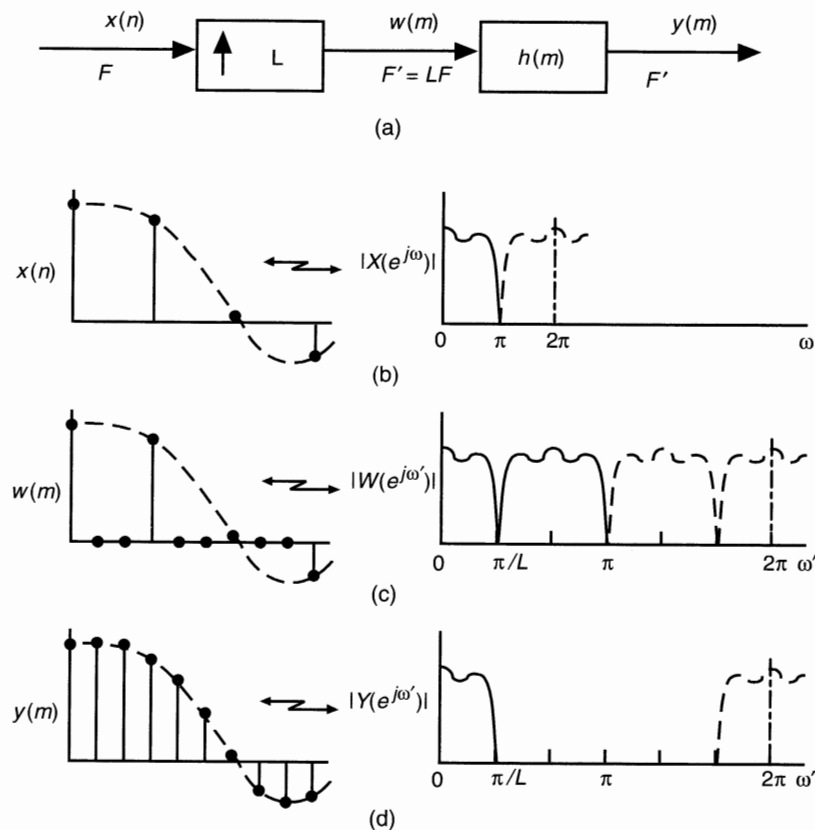


Figure 13.3 (a) Block diagram and (b) typical waveforms and spectra for interpolation by an integer factor L .

the other if the same parameters (cutoff frequencies and attenuation levels) are used. Furthermore, an architecture that is efficiently defined for one process can often be transposed for use as an efficient architecture in the dual process [2].

The duality between the decimation process in a $\Delta\Sigma$ A/D converter versus the interpolation process in a $\Delta\Sigma$ D/A converter is a somewhat more complicated situation. In the $\Delta\Sigma$ A/D converter, the decimation filter serves two purposes: It is a digital antialiasing filter that removes out-of-band noise seen at the analog input to the $\Delta\Sigma$ modulator, and it removes out-of-band quantization noise produced by the $\Delta\Sigma$ modulator. However, in the $\Delta\Sigma$ D/A converter, the interpolation filter only removes images of the baseband PCM input. Since this filter comes before the $\Delta\Sigma$ modulator, the analog filter must remove the out-of-band quantization noise produced by the modulator. The analog low-pass filter can also aid in the removal of high-frequency images left over in the digital interpolation filter. Hence, a $\Delta\Sigma$ D/A converter signal path includes two low-pass filters: one digital and one analog. Therefore, the specifications of the digital interpolation filter of the D/A converter can be somewhat relaxed when compared with those for the decimation filter of the A/D converter, that is, if the A/D and D/A signal paths have the same end-to-end filtering specifications.

13.3.4 Fractional Rate Changing

By combining the processes of integer interpolation and decimation, it is also possible to perform sampling rate conversions by rational ratios of M/L . These techniques, while important in other applications, are typically not required for oversampled A/D conversion (where large integer ratios are used) and therefore will not be discussed here. It should be noted, however, that once a signal is defined as a digital PCM signal, it is a relatively straightforward process to convert its sampling rate to a different rate through a process of digital interpolation. If the two sampling rates are related by a rational ratio, then the process can be defined as a periodic linear time-varying system. If the two sampling rates are not rationally related or are drifting independently, then direct digital-to-digital conversion is still possible by means of oversampling and/or by interpolation with digital filters that have (not necessarily periodic) time-varying coefficients [2, 3].

13.4 MULTISTAGE CONVERSION

An important criterion in the design of an A/D or D/A converter is the efficiency in which the decimator or interpolator operation can be implemented. This efficiency is directly related to the type, the order, and the architecture of the digital filter used in the implementation. Typically the order of an FIR low-pass filter is directly related to a function of the required ripples δ_p and δ_s in the passband and stopband, respectively, and inversely related to the normalized width of the transition band; that is, it has the approximate form [2]

$$N \approx \frac{D_\infty(\delta_p, \delta_s)}{\Delta F/F} \quad (13.3)$$

where

$$D_\infty(\delta_p, \delta_s) = \log_{10} \delta_s [a_1 (\log_{10} \delta_p)^2 + a_2 \log_{10} \delta_p + a_3] + [a_4 (\log_{10} \delta_p)^2 + a_5 \log_{10} \delta_p + a_6] \quad (13.4)$$

where $a_1 = 0.005309$, $a_2 = 0.07114$, $a_3 = -0.4761$, $a_4 = -0.00266$, $a_5 = -0.5941$, $a_6 = -0.4278$, and ΔF is the transition bandwidth (i.e., stopband cutoff minus passband cutoff), and F is the sampling frequency at which the filter design is referred. When large oversampling ratios are required, as in the case of $\Delta\Sigma$ converters, it can be seen that the cutoff frequency requirements on the digital filters defined above can become extremely severe; that is, ΔF becomes small relative to F , leading to excessively large filter orders and high-word-length requirements on these filters. Fortunately these constraints can be overcome by considering multistage designs in which the decimator or interpolator is defined as a cascade of two or more stages such that the overall conversion ratio is the product of the ratios of the stages.

To illustrate this point, consider the example of a 64-to-1 decimator ($M = 64$) implemented as a one-stage and a two-stage design, as shown in Figure 13.4. In this example, it is desired to convert a signal from a sampling rate of 2,822.4 kHz to one of 44.1 kHz with a baseband extending from zero to 20 kHz and passband and stopband ripples of 0.001 and

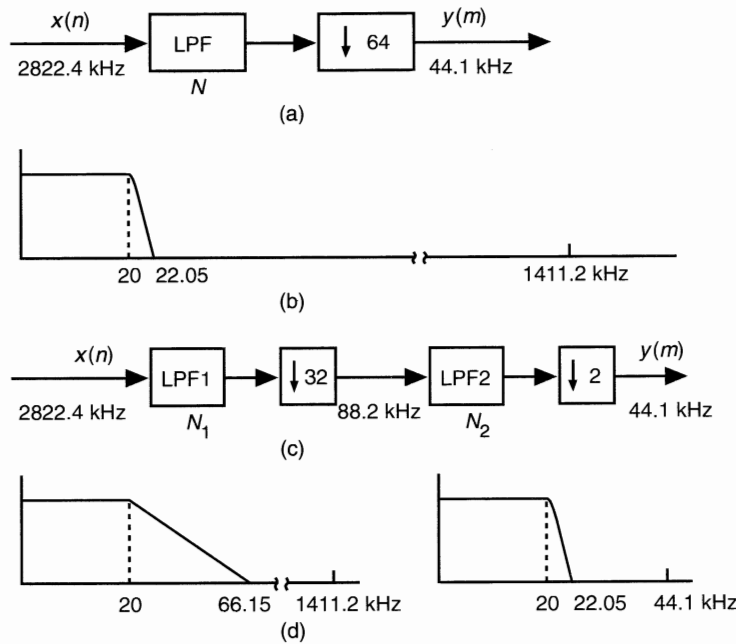


Figure 13.4 Simple example of a one-stage and a two-stage network for decimation by a factor of 64 to 1.

0.00001, respectively ($D_\infty = 4.54$). In the case of the single-stage design illustrated in Figure 13.4(a) and (b), it can be shown that these filter constraints lead to a filter size requirement of 6250 taps and a resulting computation rate of 138×10^6 multiplies/sec. The reason for this large filter size is due to the fact that a transition band of 2.05 kHz (i.e., 22.05 – 20 kHz) must be realized in a filter that is referenced to the extremely high sampling rate of 2822.4 kHz.

Now consider the two-stage implementation in Figure 13.4(c). The decimator is realized as a cascade of two decimators, a 32:1 converter that reduces the sampling rate from 2822.4 to 88.2 kHz and a 2:1 converter that then reduces the rate from 88.2 to 44.1 kHz. For the first stage, the passband is from 0 to 20 kHz but the transition region can be relaxed in the range 20–66.15 kHz; that is, signal aliasing is deliberately allowed in the 22.05–44.1-kHz region of the intermediate baseband signal. This aliasing is then removed by the second stage, which has the final desired transition band from 20 to 22.05 kHz but is now referenced to a much smaller intermediate sampling rate of 88.2 kHz. These values can now be applied to Eq. (13.3), recognizing that the passband ripples must be reduced by 2 so that the cascaded result gives the desired passband ripple; that is, $D_\infty(\delta_p/2, \delta_s) = 4.77$. This gives the respective filter orders of $N_1 = 291$ and $N_2 = 205$ for the two stages. The combined filter order is about 12 times smaller than that of the one-stage design. A total computation requirement of only 17.3×10^6 multiplies/sec for the two-stage design leads to a reduction in computation of a factor of 8 over that of the one-stage design. This same conversion efficiency could be obtained in a 1:64 interpolator design because of the duality properties discussed above; that is, a two-stage interpolator design is considerably more efficient than a one-stage design.

Further reductions can be achieved in principle by considering more than two stages. Using the same example specifications, the decimator could be implemented in three stages: an 8:1 first stage, a 4:1 second stage, and a final stage of 2:1. The respective filter orders for this design are $N_1 = 45$, $N_2 = 38$, and $N_3 = 211$ and lead to a total computation requirement of only 14.3×10^6 multiplies/sec for the overall design. Examples of audio band $\Delta\Sigma$ converters with more than two stages of FIR filtering are given in [4, 5]. Practical considerations of implementing more than two stages, however, sometimes lead to the conclusion that a two-stage design is best. Also, it is possible to consider other choices of conversion rates for the two stages. These trade-offs have been considered extensively, and the reader is referred to reference [2] for more details. For most cases, however, the choice of 2:1 for the last stage is both the theoretically best option as well as the most practical one.

In the above example, we assumed that the filter design for the first stage was a conventional FIR filter whose order was approximated by Eq. (13.3). Further reductions in this design are often possible by a more careful consideration of the filter requirements in this stage (or for that matter any intermediate stage of a multistage design). These requirements are often more easily visualized in the interpolator than in the decimator and can then be translated to the case of a decimator because of the duality properties discussed above. Figure 13.5 illustrates these general requirements for the intermediate stage of an

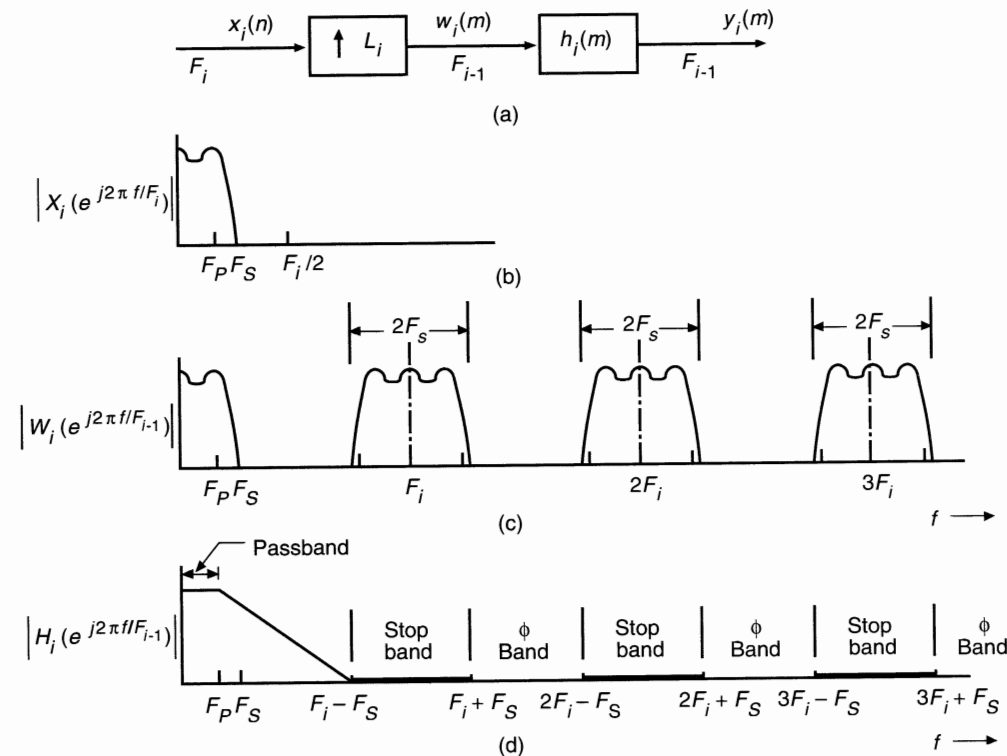


Figure 13.5 General multiple stopband specifications for stage i of an I -stage interpolator.

interpolator. The sampling rate expander creates unwanted spectral images of the baseband signal at periodic multiples of the input sampling rate. If the width of these images is less than that of the initial baseband signal (e.g., due to filtering in previous interpolator stages), then the filter in this stage only has to remove these spectral images, and the spectral regions between these images become “don’t care” bands. As a result, the filter requirements can be considerably relaxed to that of a multiple stopband design, illustrated in Figure 13.5(d), and lead to the option of using simplified comb or sinc^K filters that are able to take advantage of this property for high-sampling-rate stages of a multistage design. By duality these same requirements apply to the transposed design of a multirate decimator, where the stopband regions of the specification are regions of the spectrum that get aliased into the final baseband and the don’t care regions represent spectral regions where aliasing occurs but gets removed by later stages of the decimator design. These design considerations are considered in the next section on filter design choices.

13.5 FILTER DESIGN CONSIDERATIONS

As seen from the previous discussion, the design of a decimator or interpolator basically revolves around the design of a digital low-pass filter with single or multiple stopbands. In principle, any number of classical filter design techniques can be applied to meet these requirements. However, because of the multirate and/or multistage considerations, many of these classical techniques are often ruled out in favor of designs that can take better advantage of the above multirate criteria and achieve a more effective design. Such choices are often driven by practical hardware considerations, especially when extremely high sampling rates are involved. These considerations often lead to the use of architectures that minimize the coefficient word lengths, eliminate the need for a dedicated high-speed parallel multiplier, reduce the memory storage requirements, or make the program control simpler. We will discuss the hardware implications in greater detail in Section 13.7. The choice of architecture is also heavily driven by the specifications of the application. For example, a strict linear phase is required of most digital audio data converters. This reason alone rules out the use of an IIR filter as an intermediate-stage decimator.

13.5.1 sinc^K Filters

As seen in Chapter 1, one of the most effective illustrations of matching design simplicity with the above multirate criteria is given by the use of a sinc^K filter for the high-rate stage of a decimator or interpolator [6–9]. They are very attractive for hardware implementation because they do not require the use of a digital multiplier. They are most efficiently implemented by cascading K stages of accumulators operating at the high sample rate, followed by K stages of cascaded differentiators operating at the low sample rate [8]. Such architectures utilize wrap-around arithmetic and are inherently stable. The transfer function for a sinc^K decimation filter has the general form

$$H(z) = \left(\frac{1}{M} \frac{1 - z^{-M}}{1 - z^{-1}} \right)^K \quad (13.5)$$

and its frequency response is therefore

$$|H(e^{j\omega})| = \left(\frac{1}{M} \frac{\sin(\omega M/2)}{\sin(\omega/2)} \right)^K \quad (13.6)$$

where

$$\omega = 2\pi f/f_s \quad (13.7)$$

It has $M/2$ spectral zeros if M is even, or $\lceil M/2 \rceil - 1$ if M is odd, at frequencies that are multiples of the decimated sampling frequency ω_d , that is,

$$H(e^{j\omega}) = 0 \quad \omega = n\omega_d \quad n = \{1, 2, 3, \dots, \lfloor M/2 \rfloor\} \quad (13.8)$$

Figure 13.6 illustrates an example of the frequency response of a sinc^3 filter having a decimation factor of $M = 16$. By matching this frequency response to the multiple stopband filter requirements shown in Figure 13.5 and considering the properties of duality, it is seen that this class of filters can be applied to the first stage of a multistage decimator. An important consideration in this design procedure is that the width of the notches or stopbands in the filter be wide enough to cover the required stopband widths of the decimator specification. The frequency bands between these notches become the don’t care bands. The critical bands for aliasing considerations are found at

$$\omega = n\omega_d \pm \omega_p \quad (13.9)$$

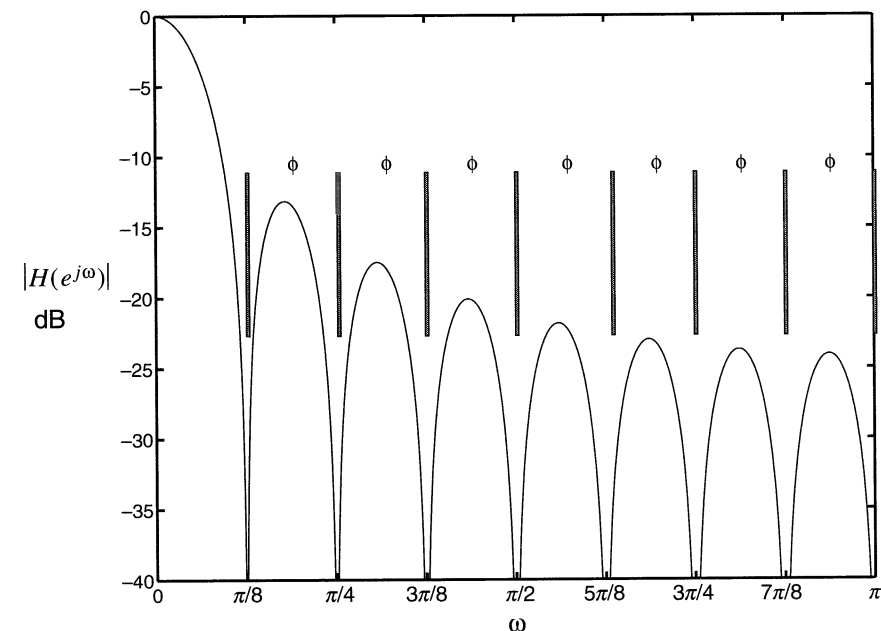


Figure 13.6 Frequency response of sinc^3 filter with $M = 16$, showing “don’t care” bands.

where ω_p is the passband edge. The least amount of aliasing attenuation (worst case) will be found just before the first null precisely at

$$\omega = \omega_d - \omega_p \quad (13.10)$$

The graph shown in Figure 1.38 illustrates this worst case aliasing attenuation as a function of the oversampling ratio of the decimation frequency and the order K of the sinc^K filter.

The passband response of this class of filters is seen to droop across the band and may be too low to meet the ripple requirements at the passband edge. A graph showing the passband droop as a function of the oversampling ratio of the decimation frequency and the order K is illustrated in Figure 1.37. This is often compensated for in the final stage of the decimator by a filter design that tilts upward at the upper end of the passband and results in an overall passband response that is flat. Such filter design characteristics can readily be accommodated with the newer filter design packages [10].

Candy [6] has shown that the order K of the sinc^K filter should be at least 1 plus the order of the $\Delta\Sigma$ modulator in order to prevent excessive aliasing of out-of-band noise from the modulator from entering the baseband. Figure 1.36 illustrates this effect of aliasing from the decimator. This rule applies to the decimator design, but not to the interpolator. It was also shown in Chapter 1 that this rule does not necessarily apply to high-order single-stage modulators whose noise transfer functions include both poles and zeros, because of the sharply rising noise at the band edge. However, sinc^K decimators can be successfully used with these type of modulators, but the minimum order K must be carefully chosen based on noise aliasing analysis.

The above discussion on sinc^K filters illustrates just how dramatically the choice of decimator architecture is affected by the choice of modulator architecture.

13.5.2 Half-Band Filters

A second class of special-purpose filters that have frequently been applied in multi-stage decimators and interpolators are half-band filters [2, 9, 11, 12]. Half-band filters are characterized by the constraints that their passband and stopband ripples are the same (i.e., $\delta_p = \delta_s$) and that the cutoff frequencies are symmetrical around $\pi/2$ such that

$$\omega_p + \omega_s = \pi \quad (13.11)$$

These properties, illustrated in Figure 13.7, lead to a family of filters that exhibit odd symmetry around $\pi/2$ and whose impulse responses $h(n)$ have zero values for all even values of n except $n = 0$. Therefore, these filters can be implemented with half the number of multiplications than arbitrary choices of filter designs. They are appropriate for sampling rate conversion ratios of 2:1 and are useful for higher rate stages of multirate decimators or interpolators where conversion ratios of 2 occur in each stage. These designs can also be used in the baseband if aliasing (or imaging) is allowed in the final transition band of the system.

Transition band aliasing (or imaging) is a very useful and important technique that leads to a large savings in filter complexity, especially in the last stage of the decimator (or first stage of the interpolator). Transition band aliasing is typically employed in commercial audio band and voiceband data converters. As an example, we will reconsider the multistage audio band filter example previously described in Section 13.4. The reader will

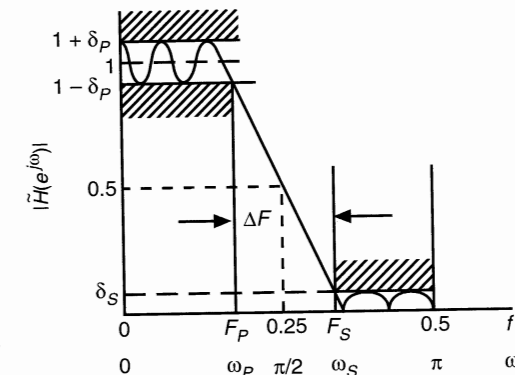


Figure 13.7 Design criteria for each stage half-band filter.

recall that the two-stage design resulted in a filter order $N_2 = 205$ in the last stage, which is derived from specifications that do not allow transition band aliasing; that is, the stopband of the final filter stage was 22.05 kHz, which is half the final sample rate of 44.1 kHz. If transition band aliasing is allowed instead, then the transition band will extend from 20 to 24.1 kHz. After the final 2:1 decimation, energy in the frequency band of 22.05–24.1 kHz will alias back into the band from 20 to 22.05 kHz. This doubling of the transition bandwidth results in a filter of length $N_2 = 103$, which is half that of the original.

13.5.3 Ternary-Encoded FIR Filters

A third type of FIR digital filter that has been applied to decimators and interpolators is a class in which all coefficient values are ternary [13–16], that is, $\{+1, 0, -1\}$. In this approach, similar to sinc^K filters, the objective is to achieve a design that requires neither a hardware multiplier nor a RAM for storing intermediate state variables. An additional objective is to significantly reduce the number of bits needed in coefficient storage. The convolution entails only additions (+1) or subtractions (−1).

Two types of design procedures have been reported to realize such filters. The first entails passing the desired filter's impulse response through a noise-shaping coder, such as a $\Delta\Sigma$ modulator, having an internal quantizer with three levels, $\{+1, 0, -1\}$. The main idea is to noise shape the ideal coefficients such that the passband response and the early part of the stopband response remain sufficiently accurate within the specification. The stopband attenuation then gradually rises (degrades) with increasing frequency according to the characteristics of the noise-shaping coder that was used for encoding the impulse response. When this technique is applied to $\Delta\Sigma$ D/A converters, the output of the ternary-encoded interpolator is fed into the *real-time* $\Delta\Sigma$ modulator, followed by an analog low-pass filter. The total output noise prior to the analog low-pass filter is therefore the superposition of the noise-shaped frequency response of the interpolator and the real-time $\Delta\Sigma$ quantization noise. The analog low-pass filter not only removes out-of-band quantization noise but corrects the upward tilt of the out-of-band frequency response due to the encoded interpolator. An example of a 1:64 interpolator for an audio band D/A converter is given in [14]. The filter length reported here is 1792 taps, while the analog low-pass filter is a third-order Butterworth. This results in a combined system frequency response having a passband ripple of 0.1 dB and stopband rejection of 74 dB.

There are several obvious problems associated with this design procedure. First, the noise-shaping coder has an infinite impulse response by its very nature, because it employs feedback. As a result, when the finite-length desired filter is passed through it, the ternary output sequence from the coder will generally not decay to zero as the tail of the filter decays to zero. Nearly any sort of ad hoc procedure that forces the noise-shaping coder to decay, such as integrator leakage, may still produce a result that is less than optimum, because the coder itself is forced to be less than optimum. Second, it is not always acceptable to produce a filter that requires additional low-pass filtering at the output to correct the stopband.

A more general and optimal design procedure [15, 16] for producing a ternary-coded filter entails a recursive algorithm that finds the time-domain minimum mean-square error objective function defined as [15]

$$E = \sum_{n=-\infty}^{\infty} [g(n\tau) - \delta h(n)]^2 \quad (13.12)$$

where $g(\cdot)$ is the target impulse response, $\tau = T/K$ is the sampling period, δ is a scalar quantity used in the optimization, K is the oversampling factor, and $\{h(n)\}$ is the approximation sequence equal to the impulse response of the ternary FIR filter. In this procedure, the $\{h(n)\}$ is convolved with multiple accumulators to produce a composite approximation of the desired filter response. Unlike the noise-shaping procedure, the algorithm is capable of generating filters that have relatively flat stopbands and low oversampling ratios. This makes such filters more generally useful and allows for multistage implementation. Using this technique, the resulting filter order will generally be much shorter than that designed using noise-shaped encoding. In addition, since the out-of-band filter characteristic is not noise shaped, such filters can be used for decimation of an over-sampled A/D converter (since there is no postfilter to attenuate the out-of-band energy as with the D/A converter).

13.5.4 Combining sinc^K Filters with FIR and IIR Filters

Perhaps the most popular filter architecture for $\Delta\Sigma$ A/D and D/A conversion entails the combination of a sinc^K filter at the high sampling rate and an FIR or IIR filter operating at the intermediate and low sampling rates. In many applications, such as high-quality digital audio, linear phase is important; hence, FIR filtering is used exclusively [4–5, 9]. In voiceband telephony applications, however, linear phase is not required. In addition, the antialiasing stopband attenuation is only about 45 dB. This leads to the use of an IIR filter operating typically between two and five times the lowest sampling rate [17–19]. Infinite impulse response filters typically require higher internal word lengths for state variables in order to avoid undesirable effects of limit cycles. Even so, this architecture is a reasonable and efficient choice when the IIR filter order is relatively low, such as sixth or seventh order [18, 19]. A major advantage with this architecture is that the filter coefficients can be quantized down to a few bits, so that the filter can be efficiently implemented with bit-serial cascaded biquadratic structures [17] or possibly wave digital filters [20]. Efficient implementation of this architecture depends heavily upon how short the coefficient word lengths can be quantized, since the number of 1-bit adders is determined by the number of 1's in the coefficients. The objective here is to quantize the ideal coefficients such that the

resulting nonideal pole and zero positions from one stage can be compensated for by the quantized pole and zero positions of the other stages. In the end, this may entail adding additional poles or zeros. When the filter order is relatively low, this method can be done systematically by analyzing the frequency responses of a large number of different combinations of quantized coefficients from all the stages. It will be found that some coefficients are very sensitive, while others are not. The sensitive ones will ultimately require longer word lengths.

We previously discussed the concept of transition band aliasing and showed its advantages in reducing filter complexity. We showed that allowing transition band aliasing in an FIR filter resulted in about a factor of 2 reduction in the complexity of the filter. The reduction is far more modest for an IIR implementation. For example, assume that a voiceband filter has a passband edge at 3.4 kHz, a passband ripple requirement of 0.2 dB, a stopband ripple of 45 dB, and a final sample rate of 8 kHz. Without transition band aliasing, the transition band would extend from 3.4 kHz to the stopband of 4 kHz. With transition band aliasing, however, the transition bandwidth is doubled so that the stopband is 4.6 kHz. If transition band aliasing is not allowed, a seventh-order elliptic filter sampling at 16 kHz is required in the final stage before decimation to 8 kHz. When transition band aliasing is allowed, the filter is sixth order, saving only one order.

13.5.5 Minimum-Phase FIR Filters

While some applications require linear phase to avoid phase dispersion, other applications require a minimum average delay instead, especially applications involving digital echo cancellation. This leads to the use of minimum-phase filters. Minimum phase, per se, only means that the filter does not contain poles or zeros outside the unit circle in the z -plane. This automatically makes most classical IIR filters minimum-phase filters as well, but some types have lower passband delay than others. An *optimal* minimum-phase filter is one in which the worst-case passband delay is the lowest possible value for a given set of frequency constraints. Such a minimum delay can be achieved with an FIR filter designed according to the *minimum-phase alternation theorem* of Chen and Parks [21]. The resulting filter has a group delay characteristic somewhat similar to that of a Chebyshev-II IIR filter, but the worst-case passband delay is typically lower than that of any classical-type IIR filter that meets the same set of frequency constraints. Although the impulse response of a minimum-phase FIR filter is not necessarily symmetrical, the worst-case passband delay is significantly lower than that of a linear-phase FIR filter having the same frequency constraints. A minimum-phase FIR filter has additional advantages: The overall order of the design is less than that of a linear-phase FIR and the coefficients are less sensitive to coefficient quantization. Minimum-phase FIR filters can readily be used in single-stage or multistage decimators and interpolators. The design procedure for optimal minimum-phase FIR filters is facilitated by use of a linear programming algorithm such as METEOR [22].

13.5.6 Compensation Techniques

Previously, we illustrated that for a $\Delta\Sigma$ D/A converter design the design of the interpolator must account for the design of the analog low-pass filter. There are other issues in the analog low-pass filter besides stopband image rejection that also affect the interpolator

design. First, the analog low-pass filter corner frequency is typically placed as close to the passband edge as possible. This causes passband droop if the analog low-pass filter does not have an equiripple response. In addition, it also causes phase nonlinearity near the passband edge. These can both be precompensated in the design of the interpolation filter characteristics. This requires that the desired objective function be applied into an FIR filter design procedure. This can be done by either modifying the popular Parks-McClellan algorithm [10] or using a linear programming technique [22]. The same technique used to compensate for the passband droop of a sinc^K filter can also be applied to compensate for the nonidealities of the analog filter.

13.6 DIGITAL FILTER STRUCTURES

Until now, the networks we have been considering have been represented primarily by block diagrams of sampling rate conversion systems but have not considered the more detailed structures within these blocks. In some cases, the manner in which a filter block is

implemented is implied by the equations; however, there are often many alternatives in how the hardware or software within the block is organized. In this section, we will consider these issues and trade-offs in more detail through the use of signal flow graphs. Signal flow graphs are often useful to more explicitly describe the manner in which a system of equations is defined within a block and can also be used to more easily define manipulations to these equations in ways that are more intuitive [2].

To illustrate this point, Figure 13.8 shows a signal flow graph of a direct-form FIR digital filter $h(n)$ with an input $x(n)$ and an output $y(n)$. This figure is seen to be a direct interpretation of the convolution equation

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k]$$

(13.13)

A less obvious realization of this same exact filter function is shown in Figure 13.9. This structure is the dual or transpose of the structure in Figure 13.8, and it is obtained by reversing the direction of all branches in the network, exchanging roles of nodes and

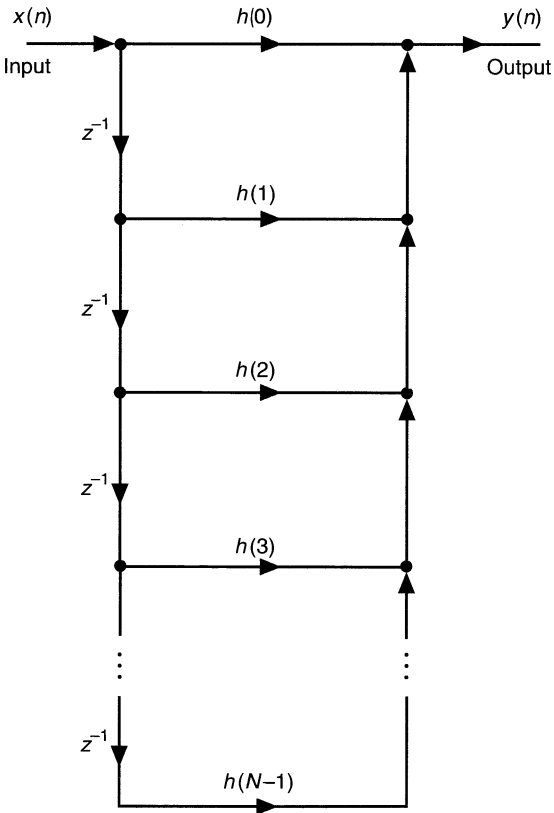


Figure 13.8 Direct-form structure for an FIR digital filter.

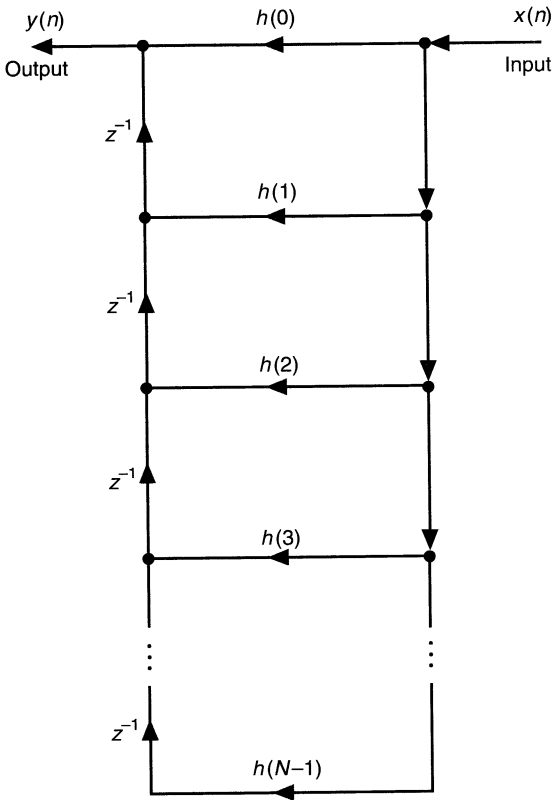


Figure 13.9 Transposed direct-form FIR filter structure.

branch points, and interchanging inputs and outputs. This operation can be applied to all linear shift-invariant digital filter structures with the same result; that is, the input-to-output relation remains unchanged with transposition. This concept, as will be seen shortly, is extremely useful in the construction of efficient structures for decimators and interpolators.

13.6.1 Direct-Form and Transpose Direct-Form Decimators and Interpolators

By applying the above direct-form filter structure in the block diagram of Figure 13.2(a), a structure for a direct-form digital decimator can be defined. This structure, shown in Figure 13.10(b), suggests that filter outputs be computed and then $M - 1$ out of M of them be discarded, leaving the desired decimated samples $y(m)$. By recognizing that

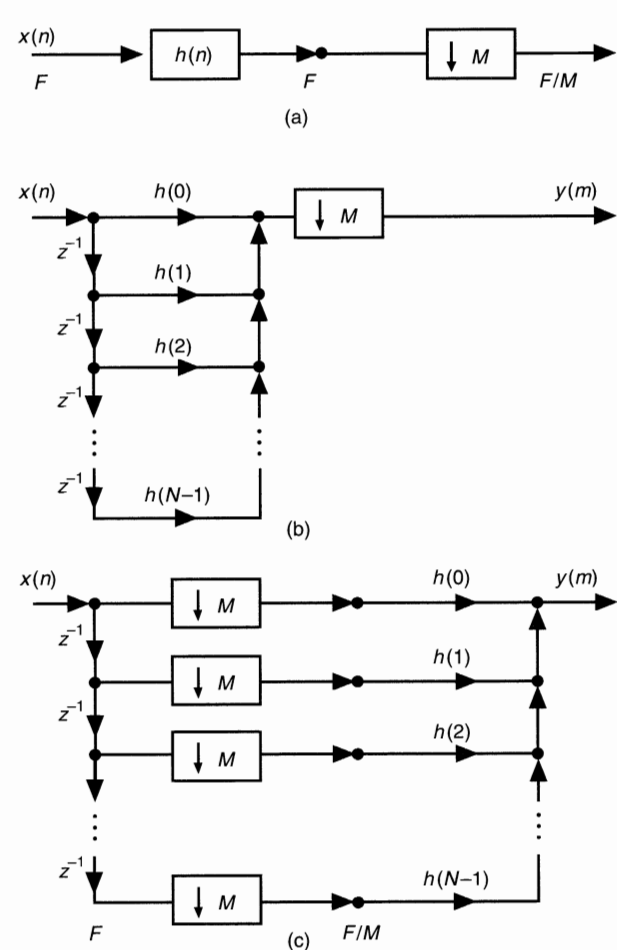


Figure 13.10 Generation of an efficient direct-form structure of an M -to-1 decimator.

the sampling rate reduction operation and the multiplication operations can be interchanged, the structure in Figure 13.10(c) results. This structure suggests that a factor of M in computation is saved by performing the filter functions at the low sampling rate side of the structure rather than the high sampling rate side. It is a direct interpretation of Eq. (13.1).

Figure 13.11 shows a similar set of operations for defining a structure for a 1-to- L interpolator. In this case, however, it is desirable to apply the transpose direct-form structure for the filter so that the multiply operations are on the same side of the structure as the 1-to- L expander. In this way, the operations of sampling rate expanding and multiplication can be interchanged so that the filter operations can again be translated to the low sampling rate side of the structure (i.e., multiplications by zero-valued samples are avoided to achieve a more efficient implementation). If the direct-form structure were used, it would be considerably more difficult to perform this interchange in the structure.

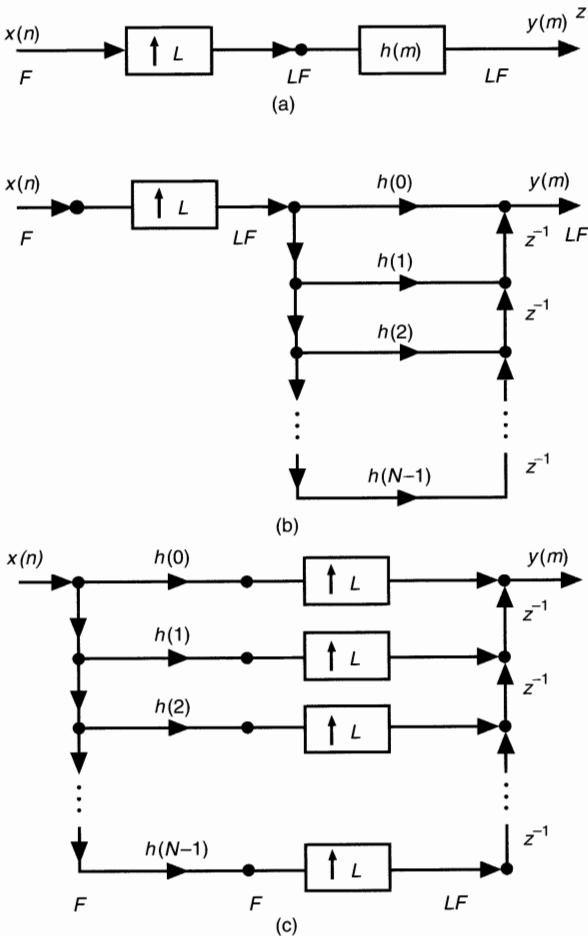


Figure 13.11 Steps in the generation of an efficient structure of a 1-to- L interpolator.

Another way to generate the interpolator structure of Figure 13.11(c) is to apply the concepts of duality and transposition [2] and directly transpose the decimator structure of Figure 13.10(c), that is, by recognizing that decimators and interpolators are duals. In this process, sampling rate compressors are replaced with sampling rate expanders, the directions of branches are reversed, nodes and branch points are interchanged, and inputs and outputs are interchanged.

13.6.2 Polyphase Architectures for Decimators and Interpolators

A second class of architectures for implementing decimators and interpolators is based on the concept of polyphase structures. An M -to-1 polyphase decimator structure can be derived by applying the change of variables

$$k = rM + \rho \quad (13.14)$$

to Eq. (13.1), where ρ takes on values $\rho = 0, 1, 2, \dots, M-1$. Then Eq. (13.1) can be rewritten in the form

$$y(m) = \sum_{r=-\infty}^{\infty} \sum_{\rho=0}^{M-1} h(rM + \rho) x(mM - rM - \rho) \quad (13.15)$$

By defining the subsampled impulse response and signals as

$$p_{\rho}(r) = h(rM + \rho) \quad \text{for } \rho = 1, 2, \dots, M-1 \quad \text{for all } r \quad (13.16)$$

and

$$x_{\rho}(r) = x(rM - \rho) \quad \text{for } \rho = 1, 2, \dots, M-1 \quad \text{for all } r \quad (13.17)$$

the form of Eq. (13.15) can be written as

$$y(m) = \sum_{\rho=0}^{M-1} \sum_{r=-\infty}^{\infty} p_{\rho}(r) x_{\rho}(m-r) \quad (13.18)$$

or as

$$y(m) = \sum_{\rho=0}^{M-1} p_{\rho}(m) * x_{\rho}(m) \quad (13.19)$$

where $*$ denotes convolution. The signal $y(m)$ in this case is seen to be the sum of M signals that are obtained by convolutions of the subsampled impulse responses convolved with the respective subsampled input signals. By careful interpretation of this equation, the polyphase decimator structure for an M -to-1 decimator can be derived, as shown in Figure 13.12. Like the above direct-form structure, it is seen that all of the computations in this structure are performed at the low sampling rate. An alternate interpretation of this structure is given by the counterclockwise commutator model in Figure 13.13. This form

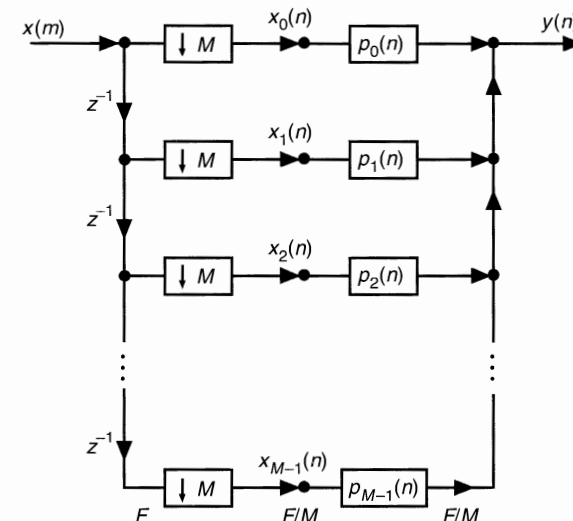


Figure 13.12 Polyphase structure for an M -to-1 decimator.

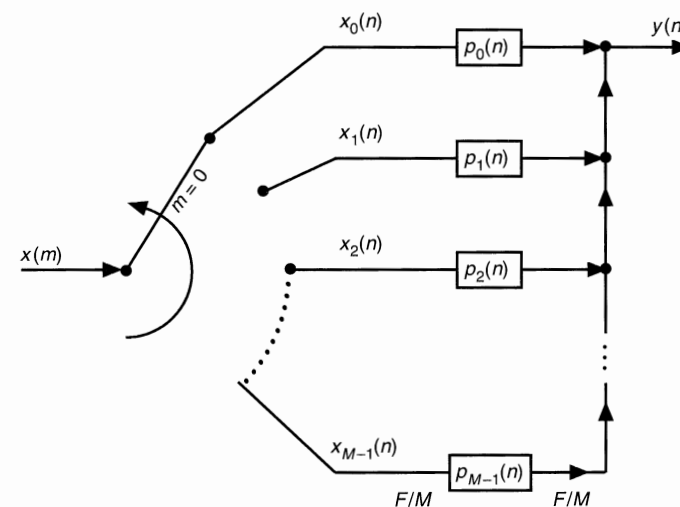


Figure 13.13 Counterclockwise commutator model for an M -to-1 decimator.

of the structure shows that the input delay line is really not needed in a practical implementation.

By appealing to the concepts of duality and transposition, a similar set of polyphase structures can be derived for integer interpolators where L is replaced by M in this formulation. This leads to the structures in Figures 13.14 and 13.15, respectively.

So far, no stipulation has been made as to the form of the filters in Figures 13.12–13.15. In theory, these subblocks or filters can be implemented by any filter structure that

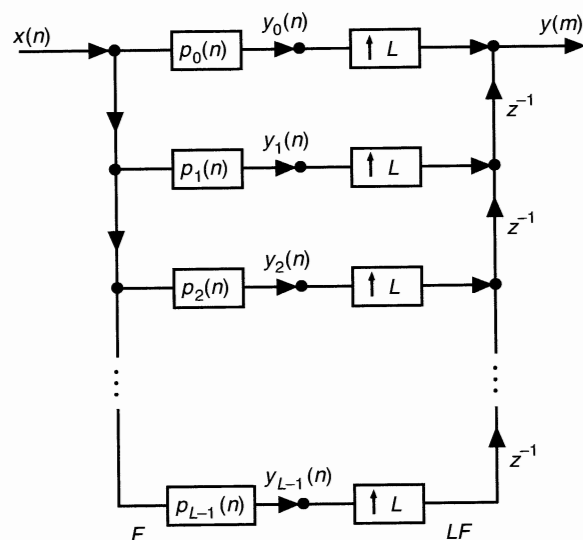


Figure 13.14 Polyphase structure for a 1-to- L interpolator.

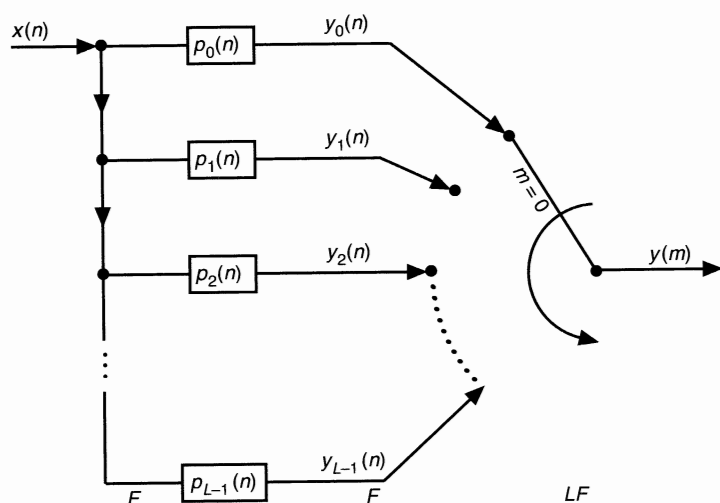


Figure 13.15 Counterclockwise commutator model for a 1-to- L interpolator.

can realize the specified impulse response. In practice, if a careful choice is made using the direct-form or transpose direct-form structures for this implementation, a savings of M (or L) in required memory locations can be achieved in the overall decimator or interpolator. For example, if the direct-form structure of Figure 13.8 is applied in the implementation of each of the polyphase filter blocks of the interpolator in Figure 13.15, then it can be seen that all of the polyphase filters can share the same tapped delay line, saving a factor of L in memory as well as in the computation. This resulting structure is illustrated in

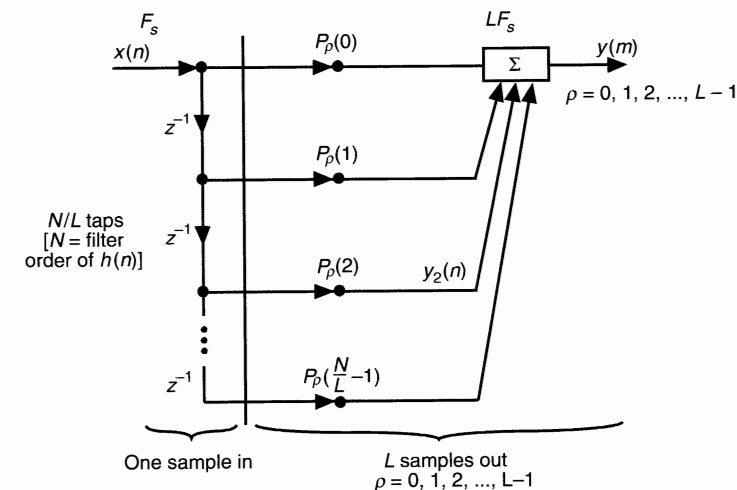


Figure 13.16 Polyphase 1-to- L interpolator structure with FIR polyphase filters that share the same tapped delay line.

Figure 13.16. The input signal $x(n)$ feeds a tapped delay line of N/L taps, where N is the total number of taps in the overall low-pass filter $h(n)$. Then L output samples $y(m)$ are computed from this filter using the respective polyphase filters defined in Eq. (13.16), where each polyphase filter is N/L taps long. In fact, the same filter structure is applied L times, only the coefficients change for each respective output sample (i.e., each step of the commutator of Fig. 13.15). The resulting structure is seen to be a two-rate structure with time-varying coefficients.

A similar memory-efficient structure can be defined for the polyphase decimator if the transpose direct-form filter of Figure 13.9 is applied to each of the polyphase filters in the decimator structure of Figure 13.13. In this case, all of the output-tapped delay lines of the filters can be shared. The resulting structure is shown in Figure 13.17, and it consists of a two-rate structure in which M input samples are weighted by the polyphase coefficients and accumulated to produce one output $y(n)$. This structure is slightly more complicated to visualize because each delay element must receive the accumulated products from all of the polyphase filters before they are applied to the tapped delay line. This set of operations is seen to be the transpose of the interpolator in Figure 13.16, where branching points and summing points are interchanged in the structure. The extra accumulator in each polyphase branch is the transpose operation to branching or reuse of the tapped delay line signals in the interpolator. The structure requires N/M taps in the delay line versus N taps for a nonshared architecture. This savings of M in memory must be traded for a somewhat more complicated control structure and accumulator in the polyphase implementation.

13.6.3 Multistage Architectures

The above architectures apply to single-stage decimators or interpolators with integer sampling rate conversion ratios. They also apply to the implementation of individual

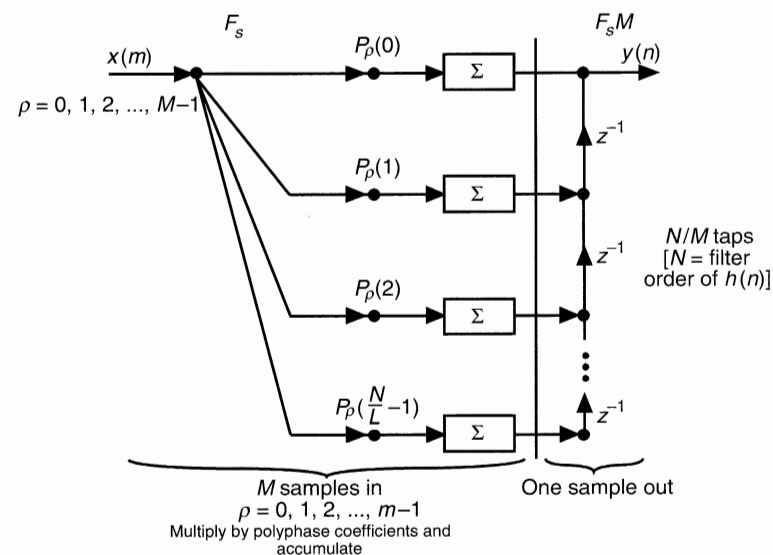


Figure 13.17 Polyphase M -to-1 decimator with FIR polyphase filters that share the same output tapped delay line.

stages in multistage designs. When used in multistage architectures, additional considerations must be given to the control structure necessary to deal with the intermediate sampling rates and the data flow problems associated with controlling and synchronizing the inputs and outputs of the different stages. This control can be linked to a multiple clocking structure in hardware or to a multiple loop control structure in a software implementation.

To illustrate this control process, Figure 13.18(a) shows an example of a data flow graph for the implementation of a three-stage decimator using direct-form filter structures for each stage. This structure could be realized in hardware or software, although it is illustrated as a software flow graph. In the process of computing one decimated output signal, a block of M input signals must be entered in the decimator, where M is the product of the conversion ratios of the three stages, M_1 , M_2 , and M_3 , respectively. Three state-variable buffers S_1 , S_2 , and S_3 are used to hold internal data for the filters whose tap lengths are N_1 , N_2 , and N_3 , respectively, for the three stages. Figure 13.18(b) shows the control structure for controlling the flow of data from stage to stage. This control process consists of three loops, one within the other, to maintain the proper timing and flow of data in the data structure. Inputs to the decimator are shifted into the first stage in blocks of M_1 samples and one output is computed and shifted into stage 2. After M_2 outputs are computed in stage 1 and shifted into stage 2, one output is computed from stage 2 and shifted into stage 3. The above process is repeated, as indicated by the control chart, until M_3 samples are computed and shifted into stage 3. At this point one output from the final stage of the decimator can be computed and M input samples to the decimator have been used up. Then the whole process can be repeated again for the next block of M input samples.

The above structure can be modified to accommodate any of the structures described earlier for the individual stages. By reversing the direction of flow, a similar data flow and control structure can also be defined for multistage interpolators [2].

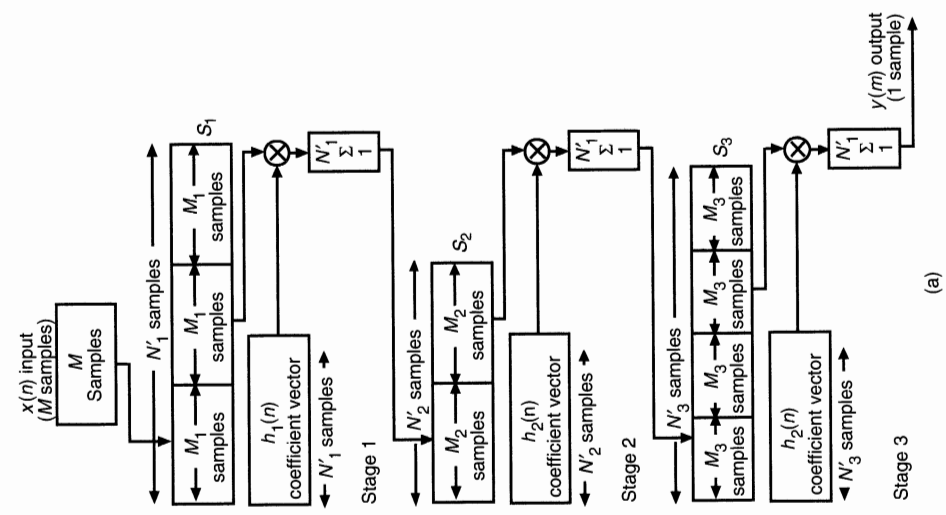
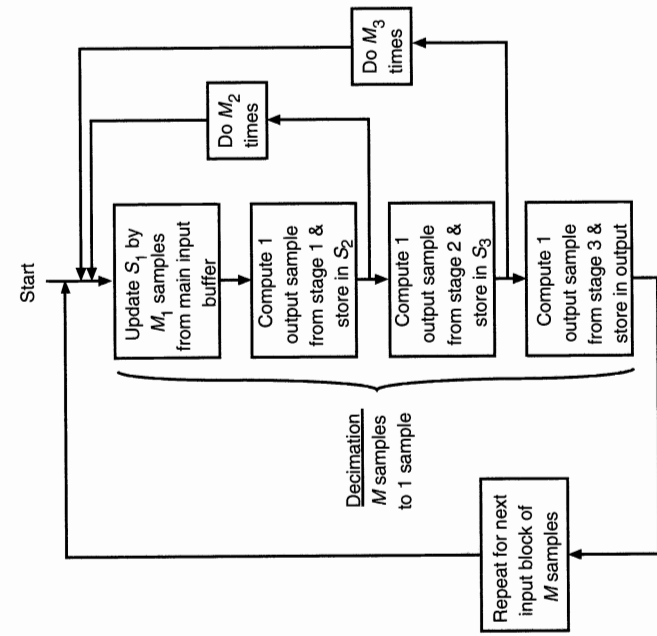


Figure 13.18 (a) Data flow structure and (b) control structure for a three-stage FIR decimator.

13.7 HARDWARE IMPLEMENTATION—ARCHITECTURAL ISSUES

In the last section, we examined various algorithms and architectures for efficient decimators and interpolators. We also discussed basic considerations in their implementation. We now turn our attention specifically to the issues regarding hardware architecture, that is, items in the second column of Table 13.1.

13.7.1 Historical Background

Rapid advances in integrated circuit technology since 1980 have caused rapid changes in digital filter hardware implementation over the same period. The result: What was an efficient hardware architecture *yesterday* may no longer be attractive *today*; that is, the design trade-offs outlined in Table 13.1 are constantly evolving with technology. In the 1970s, for example, the development of a general-purpose single-chip digital signal processor was just beginning. Digital filters, at that time, were implemented with discrete building blocks at the board level. The primary issues and concerns on the subject of hardware realization of digital filters during this era are seen in the earliest classic papers on the subject [17, 23–27], which spanned the time from the late 1960s to the early 1980s. These papers were concerned with two major issues: (1) how to avoid the use of parallel multipliers and (2) how to reduce the memory sizes for the state variables and coefficients. The earliest hardware filter designs were more concerned with the realization of efficient IIR filters rather than FIR filters, since FIR filters generally require a large number of coefficients and state variables. Large memory chips at that time were not available, so memory was often implemented with flip-flops.

In the 1980s, commercial signal processing chips, such as echo cancellers and modems for telephony, were commonly implemented with dedicated special-purpose architectures [28, 29], because single-chip digital signal processors (DSPs) and data converters were too expensive. Digital multipliers either consumed too much chip area or were too slow for the application, so multiplierless digital filter architectures became important for successful implementation in many of these chips.

In the 1990s, however, it became routine to put more than one million transistors on a chip. Submicron CMOS technology began to dramatically change the scope of the problem. For example, in a typical 0.75- μm CMOS process, a 16-by-16 parallel multiplier would have a cycle time of about 25 ns and a chip area of about 1–2 mm^2 . As for memory, a typical 2048 $\times$ 16 static RAM in the same technology would consume about 3–4 mm^2 . To put these figures in perspective, it would not have been uncommon for a relatively large chip to consume at least 100 mm^2 in this technology. Single-chip general-purpose DSPs began to include on-chip high-resolution $\Delta\Sigma$ data converters [30, 31].

The degree of complexity and computational speed offered in present-day technologies makes the issues found in the classic papers less relevant to current interests and concerns on the subject. We will therefore direct our attention only to discussing the architectures relevant to VLSI implementation based on the technologies available at the time of this writing.

13.7.2 Architectural Features and Styles

Digital filtering is perhaps the most basic function performed by a digital signal processor. All but the most trivial digital filtering hardware have certain basic features and

characteristics: They involve intensive amounts of data; they require multiple use of the same data; they involve intensive intermediate data manipulation and address generation; and they often require fast multiplication and addition. The basic building blocks for digital signal processing hardware include multipliers, adders, registers, ROM, RAM, high-bandwidth buses, address generators, instruction decoders, clock and timing generators, and input/output units including serial, parallel, and analog.

There are three general styles of digital filter architectures: *hard wired*, *reconfigurable*, and *programmable*. Hard-wired filters have an architecture so specialized that the logic and interconnections are specific to the filter characteristics. They are typically used with $\Delta\Sigma$ converters for one or more of the following reasons:

1. The filters implement dedicated functions or standards that, in all likelihood, will never change, so that the filter design will likely never change.
2. The application is extremely cost sensitive, and the low-cost benefits outweigh the relatively long design cycle required for a custom hard-wired design.
3. The application requires the lowest possible power dissipation.
4. The sampling rate is very high relative to the speed of the technology, requiring extensive use of parallelism and pipelining, so that a programmable architecture would be too slow.
5. A programmable DSP is not available in the same system as the $\Delta\Sigma$ converter.
6. A programmable DSP is available, but the filtering and data I/O operations consume too many MIPS from the DSP.

Reconfigurable filters are ones whose filter characteristics are easily changed without changing the logic and interconnections of the chip. This is often accomplished by simply changing the ROM mask of the chip, where the coefficients and possibly the program instructions are stored. In the case of programmable filters, a general-purpose DSP performs the last stages of decimation or initial stages of interpolation. Such implementations are more likely to be used when the $\Delta\Sigma$ converter is used in a system that already includes a programmable DSP.

13.7.3 Arithmetic Processing Issues

One of the most important trade-offs in determining the optimum filter architecture for a given application is the choice of the arithmetic architecture. A useful benchmark for determining this choice is the speed of a parallel multiplier relative to the required sampling rate and filter length. Figure 13.19 shows a family of curves for multiplication speeds varying from 10 to 100 ns. At one end of the curves are applications such as seismology and control, where a fast multiplier would be able to generate nearly 100,000 multiply/accumulate operations (MACs) per sample. Such multipliers applied to telephony and speech applications could still generate nearly 10,000 MACs per sample. Similarly, for audio and ISDN applications, nearly 1,000 MACs per sample would be possible. For video applications, one such multiplier would only be able to produce a few MACs per sample.

13.7.3.1 Bit-Serial and Digit-Serial Arithmetic. If the sampling rate is low relative to the speed of a single multiplier, then we can consider bit-serial arithmetic

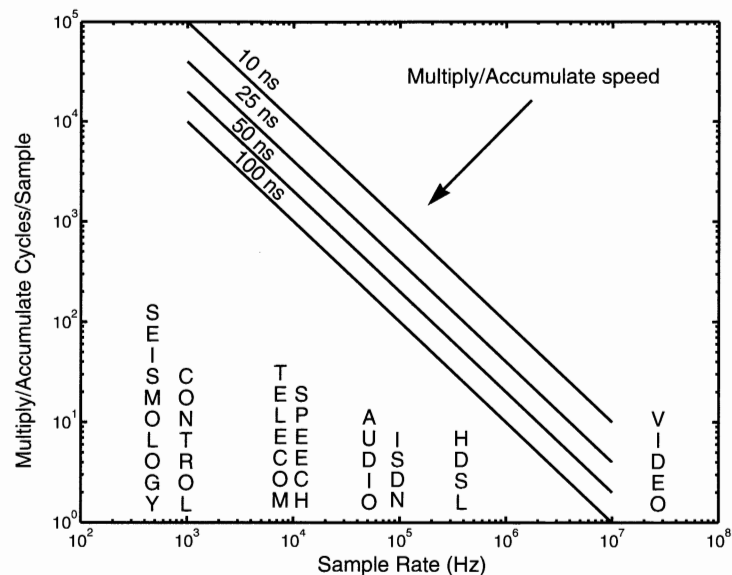


Figure 13.19 Multiply/accumulate cycles per sample vs. sampling rate multiply/accumulate speed.

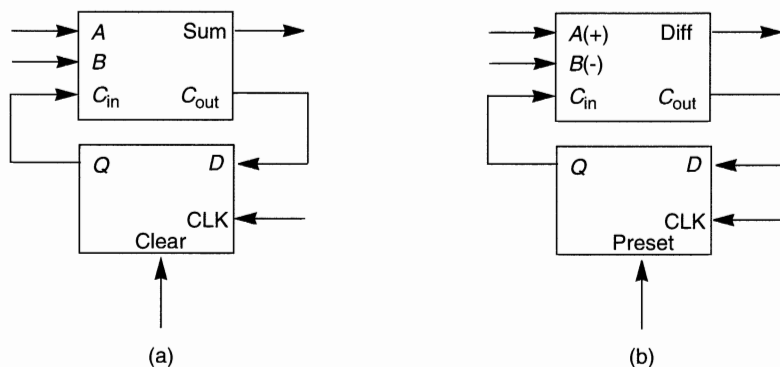


Figure 13.20 Bit-serial adder (a) or subtractor (b) consisting of 1-bit full adder (subtracter) and 1-bit register.

[17, 25, 28, 32]. A bit-serial adder is nothing more than a 1-bit full adder with a delay register between the carry-out and carry-in bits, as shown in Figure 13.20. Two N -bit operands can be multiplied by simply providing a bit-serial adder per coefficient 1 bit with the appropriate number of delay shifts, followed by a summation of outputs from the bit-serial adders. It normally takes N clock cycles to multiply two N -bit operands; therefore the speed of multiplication will be F/N , where F is the clock rate. If the clock rate is still below the maximum allowable speed of the technology, then each bit-serial multiplier can be multiplexed among K other pairs of operands [28] in order to make better use of the

chip area. The clock rate will therefore be K times faster. The objective is to make the best use of the processing elements within the limits of the technology. Underutilizing the speed of the processing elements may be an indication that the implementation is more costly than it ought to be. In any distributed arithmetic scheme, the total number of processing elements can be minimized by multiplexing. However, such multiplexing requires a more complicated control structure. This trade-off must be carefully analyzed before a final decision is made on the choice of architecture.

Bit-serial architectures reduce the interprocessor communication down to 1 bit. Generally, the number of processors is very large, but because each processor is so small, the overall economy is high. Bit-serial architectures are usually most effective for filters having only a few state variables, such as IIR filters and wave digital filters [17, 20, 25, 26]. For this reason, bit-serial techniques are less frequently applied to FIR filter structures, especially where the filter lengths are relatively long. An interesting bit-serial FIR filter can be seen in [4], where substantial effort was put on minimizing the filter lengths and coefficient 1 bits involved. Bit-serial techniques are also applicable to relatively large parallel structures requiring intensive interprocessor communication.

Higher throughput rates can be achieved with *digit*-serial arithmetic [33–35]. In a digit-serial architecture, the data and coefficients are multiplied in sections of a few bits (rather than 1 bit) per word per clock cycle.

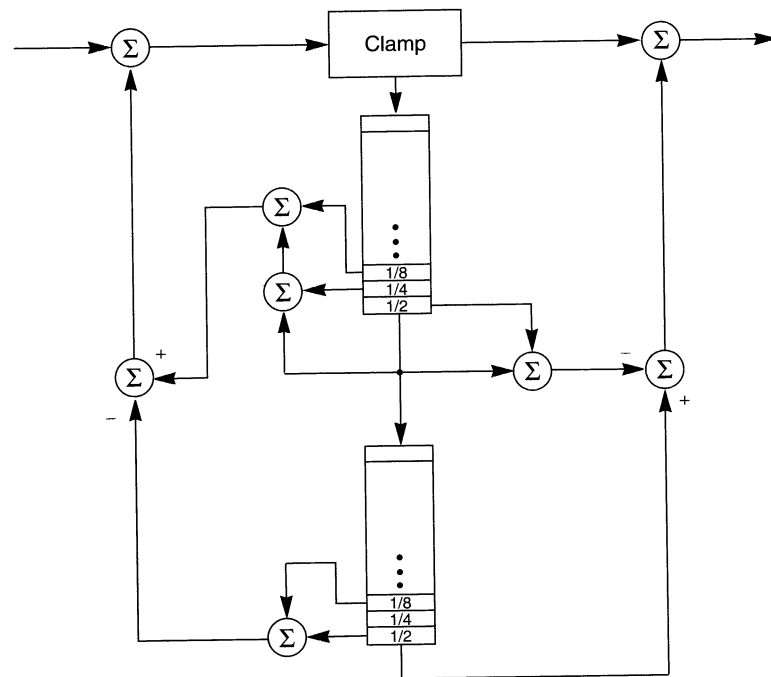
Bit-serial and digit-serial architectures are usually very complicated to design. This can be alleviated somewhat by using a special-purpose silicon compiler [36–38]. Another major disadvantage with bit-serial and digit-serial architectures is that the filter is not easily reconfigurable for other frequency responses. Architectures which use bit-serial or digit-serial arithmetic are therefore considered *hard-wired* structures. Even the control logic is dependent upon the coefficients.

When using a bit-serial architecture, great effort is usually put into minimizing the number of coefficients, their word lengths, and 1 bits. An example of a bit-serial biquad section and its z -transfer function is shown in Figure 13.21. In this example, a total of only eight bit-serial adders is needed to implement the arithmetic for the entire biquad.

13.7.3.2 Parallel Multiplication. Another choice is to use parallel multiplication. By this, we mean the multiplication of two binary N -bit operands within a single clock cycle.¹ For low-to-moderate sampling rates, a single parallel multiplier can be multiplexed among all the filtering operations, much the same as a commercial programmable digital signal processor [40, 41]. For higher sampling rates or cases where the filter lengths are very long, the multiplications may need to be distributed among several parallel multipliers. The extreme limit for high sampling rates is one dedicated parallel multiplier per filter tap.

13.7.3.3 Combined Bit-Serial and Bit-Parallel Architectures. Efficient implementations of multistage multirate filters have been proposed and constructed that combine bit-serial and bit-parallel techniques [4, 17]. The bit-parallel structures are reserved

1. The design of parallel multipliers is a mature subject and beyond the scope of this text. The reader is referred to [39].



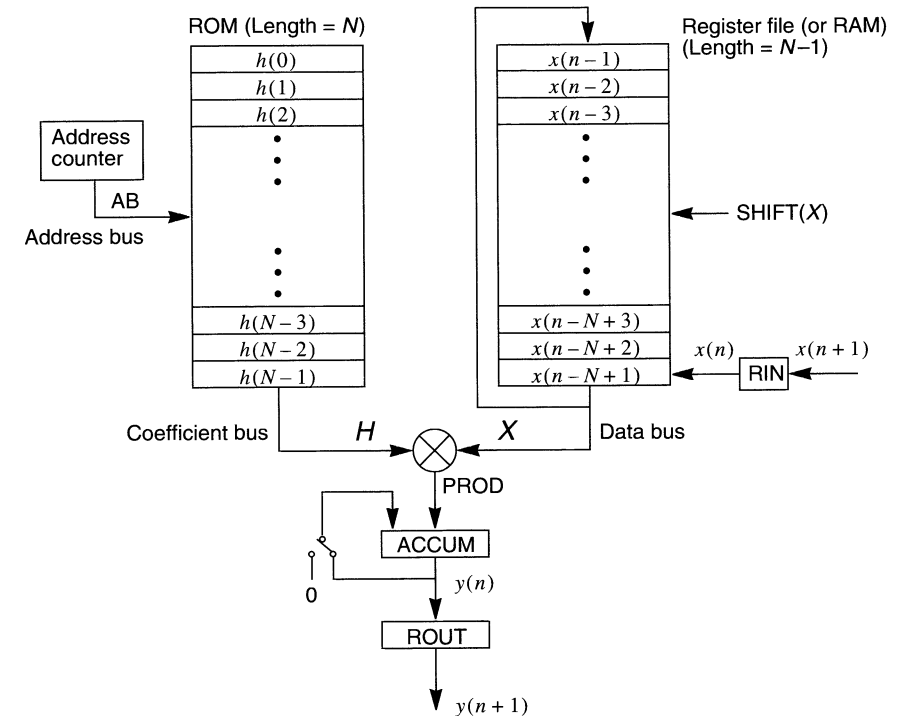
$$H(z) = \frac{1 - (3/2)z^{-1} + z^{-2}}{1 - (11/8)z^{-1} + (5/8)z^{-2}}$$

Figure 13.21 Bit-serial biquad section example that requires eight bit-serial adders.

for the higher sampling rates, while the bit-serial structures are used for the lower sampling rates. In [4], the low-rate filters are FIR, while in [17], they are IIR. In each case, the designs have short-length coefficients in order to avoid explicit hardware multipliers.

13.7.3.4 Single-Multiplier Architectures. In Section 13.6, we described the structures of various direct-form and polyphase FIR filters. We now provide two examples that illustrate implementations on a customized processor architecture.

Figure 13.22 illustrates an implementation of a direct-form FIR filter and its corresponding microinstruction sequence. The coefficients are stored sequentially in a ROM. The ROM is accessed sequentially through a simple address counter. The state variables are stored either in a register file or a RAM. There are trade-offs between the use of a register file versus a RAM. If a register file is used [42], every data sample is shifted each clock cycle. This may be disadvantageous if the number of registers is large, causing excessive power dissipation. In addition, a large register file does not lay out efficiently compared with a RAM cell. However, the layout area of a small RAM will be dominated by its support circuitry, namely, the address generation, address decoding, sense amplification, and so on. Regarding the operation of the architecture, the coefficients and state variables are multiplied and accumulated together according to Eq. (13.13), forming one



Micro-Instruction Sequence

INIT:	AB = N-1;	/* Initialize address counter every N */
	CLEAR(ACCUM);	/* Clear the accumulator */
FIRST:	PROD = H * X;	/* First multiply is h(N-1) * x(n-N+1) */
	ACCUM += PROD;	/* Multiply/accumulate */
	AB --;	/* Decrement address counter */
	LOAD(RIN);	/* Load new x(n) and overwrite x(n-N+1) */
	SHIFT(X);	/* Shift the register file */
LOOP(N-1):	FOR(n=1; n < N-1; n++) {	/* Loop N-1 times */
	PROD = H * X;	/* Next multiplies are h(N-2) * x(n-N+2); N-1 times until h(0) * x(n) */
	ACCUM += PROD;	/* Multiply/accumulate */
	AB --;	/* Decrement address counter */
	SHIFT(X);	/* Shift the register file */
	}	/* End loop */
OUT:	LOAD(ROUT);	/* New output y(n) */
RETURN:	GOTO(INIT);	/* Return to top; next sample */

Figure 13.22 Customized arithmetic logic unit (ALU) based direct-form FIR filter with micro-instruction sequence

new output sample $y(n)$ for N multiply/accumulate operations. The multiplication, accumulation, and address increment/decrement (and/or data shifting) will typically be executed in parallel within a single instruction cycle (i.e., all the instructions inside the FOR loop indicated in the example microinstruction code listing). A description of an actual implementation of a direct-form FIR filter on a commercial DSP chip, along with its corresponding microinstruction sequence, is given in [40].

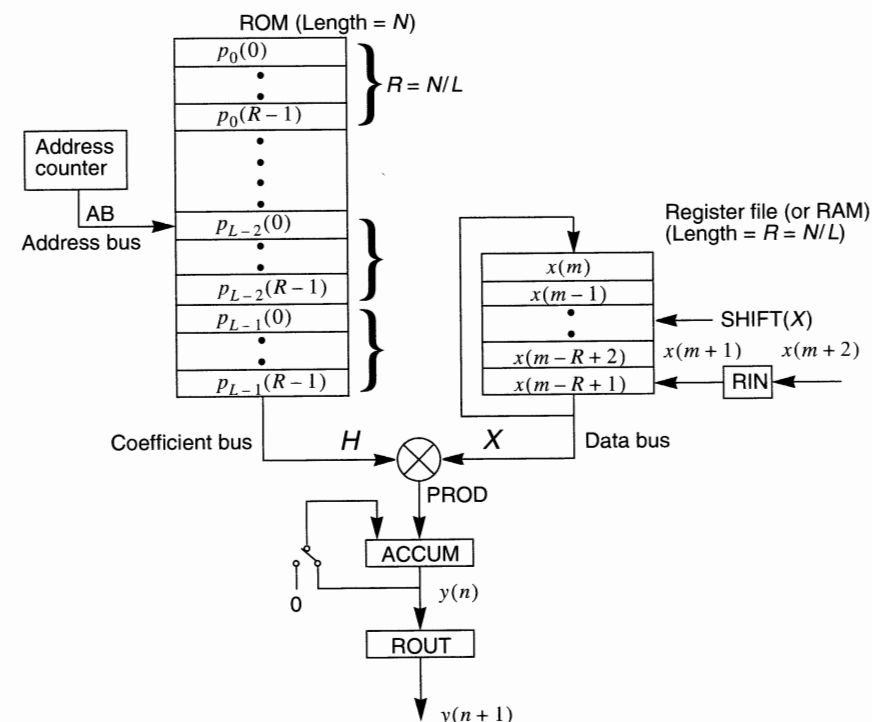
The architecture in this example does not indicate any pipelining in the data path. An example of such an architecture used in a commercial $\Delta\Sigma$ converter chip is given in [43]. For higher speed operation, this architecture can be modified to incorporate pipelining, much like that seen in the commercial DSP in [40]. This may simply entail adding delay registers for H and X at the inputs to the multiplier and adding a delay register at the output of the multiplier that feeds the accumulator. This would allow propagation delays of one entire clock period for each of the following operations: the memory access time, the multiplication time, and the accumulation time. However, it would also add a latency of two additional clock periods to the output. It will also require a different control structure and a different instruction set architecture. Deeper levels of pipelining may even be employed by adding delay registers within the multiplier in order to speed up the multiplier itself. This technique is employed in some commercial floating-point DSP chips [41].

While a direct-form FIR filter structure is straightforward and simple, it is inefficient for decimation or interpolation, as discussed in Section 13.6. Figure 13.23 illustrates an efficient polyphase FIR interpolation filter realization of Figure 13.16. While its physical structure and interconnections look much the same as that of Figure 13.22, there are many important differences. First, the register file or RAM storing the state variables is of length R , which is a factor of L shorter than before. Also, the coefficients are stored in the order of polyphase groupings, in accordance with Eq. (13.16). However, the control is more complicated than for the direct-form FIR filter. For each new input data sample $x(m)$, there are L new output samples, where each output is the result of one inner loop of R multiply/accumulates from a group of R polyphase coefficients. From the above discussion, we can see that the control structure gets even more complicated if the same physical hardware is used for implementing more than one stage of a multistage polyphase interpolator or decimator.

13.7.4 DSP and Programmable Implementations

An ever-increasing number of commercial and consumer electronics systems incorporate both programmable digital signal processors and $\Delta\Sigma$ data converters, either as two separate chips or combined as a single chip [30, 31, 44]. As the speed and capability of these DSPs continue to improve, it makes more sense to incorporate the decimation and interpolation filters of the $\Delta\Sigma$ data converters in the DSP rather than build separate and redundant dedicated hardware on the $\Delta\Sigma$ converter [44]. The limits of this approach, however, will be reached when the sampling rate of the application is so high that the filtering and data I/O operations consume too many MIPS from the DSP. This will be alleviated somewhat as DSPs incorporate multiple arithmetic units on the same chip.

Many systems must run more than one application, each requiring a unique sampling rate and filtering characteristics from the $\Delta\Sigma$ data converters. Hard-wired and reconfigurable architectures offer only limited flexibility. However, the use of a programmable DSP enables these characteristics to be changed in *real* time.



Micro-Instruction Sequence

INIT_CL:	CLEAR(ACCUM);	/* The first clearing of the accumulator */
INIT:	AB = N-1;	/* Initialize address counter every N */
OUTER_LOOP(L):	FOR(n = 0; n < L-1; n++) {	/* Do outer loop L times */
INNER_LOOP(R):	FOR(r = 0; r < R-1; r++) {	/* Do inner loop R = N/L times */
	PROD = H * X;	/* First multiply is $p_{L-1}(R-1) * x(m-R+1)$ */
	ACCUM += PROD;	/* Multiply/accumulate */
	AB --;	/* Decrement address counter */
	SHIFT(X);	/* Shift the register file */
	}	/* End inner loop */
	LOAD(ROUT);	/* New output y(n) */
	CLEAR(ACCUM);	/* Clear the accumulator */
	}	/* End outer loop */
	LOAD(RIN);	/* Load new x(m) and overwrite x(m-R+1) */
	SHIFT(X);	/* Shift new x(m) to top of stack */
RETURN:	GOTO(INIT);	/* Return to top; compute next L outputs */

Figure 13.23 Customized ALU-based polyphase FIR interpolation filter with microinstruction sequence.

Hard wiring the decimation and interpolation filters also puts a limit on the usefulness of the noise-shaping property. Using the $\Delta\Sigma$ modulation technique, circuit speed can be traded off for resolution. For example, a second-order modulator will produce approximately 15 dB better resolution for each octave increase in the oversampling ratio, until the analog noise power dominates the baseband quantization error. Similarly, for example, a third-order modulator will produce a 21-dB/octave improvement. It should be possible to use the same device as either a 16-bit converter at a sampling rate F or a 13-bit converter at a rate $2F$ or a 10-bit converter at a rate $4F$. This is in contrast with only a 3.01 dB ($\frac{1}{2}$ -bit) per octave improvement obtainable with a data converter that does not employ noise shaping. However, without programmable control of the decimation and interpolation process, the characteristics of a $\Delta\Sigma$ data converter appear (to the user) little different from those found in other types of data converters.

While the use of a programmable DSP in the implementation of multistage decimation and interpolation offers many degrees of freedom, there are many complex issues that must be addressed. In the Sections 13.7.4.1–13.7.4.3 we will discuss some of these considerations.

13.7.4.1 Multirate Filtering Efficiency on DSPs. Reference [2] discusses the merits of various structures used for decimation and interpolation filtering. There is a fundamental trade-off between computational efficiency (multiplies per second) and the complexity of the control structure. Filters that are more computationally efficient usually have a more complicated control structure. When a programmable DSP is used for executing the final-stage decimation or initial-stage interpolation filter, it will be found that some types of filter structures that have a more complicated control structure cannot be implemented efficiently on a DSP chip. Of course, this is highly dependent on the features of the particular DSP chip. For example, if the DSP has an instruction cache [40] that allows for low-overhead looping, then longer filters will execute out of the cache with greater efficiency, because of the overhead of setting up the pipeline before getting into the cache and then clearing out the pipeline after exiting the cache. For this reason, some polyphase implementations might not work as efficiently as others, since each filter in the polyphase bank is relatively short when compared with its single-phase direct-form equivalent. If the DSP has only one set of pointers supporting the beginning and ending locations of a circular buffer, then a polyphase structure may not work very efficiently. This is because each filter in the polyphase filter bank needs its own circular buffer pointers, requiring DSP instruction overhead for fetching and saving the address of the pointers each time the program switches to the next filter in the filter bank. A filter structure that works efficiently on most programmable DSPs is the time-varying coefficient implementation of an interpolator, shown in Figure 13.16 and described by the example in Section 13.7.3.4 and Figure 13.23. In this structure, the coefficients are ordered into N/L groups or phases. The state variables are shared among the phases of each polyphase group, so that only N/L RAM locations for the state variables are required and only one set of circular buffer pointers is needed, since the structure behaves as a single filter. The dual of this structure for a decimator, shown in Figure 13.17, will generally be less efficient to implement than its interpolator counterpart. This is due to the fact that the intermediate accumulations before each delay register will require more manipulation of data to and from memory, since most programmable DSPs have only a few *physical* accumulators.

13.7.4.2 Multistage Implementation Including sinc^K Filters. In Section 13.4, we showed that decimation and interpolation are usually broken up into multiple stages when the oversampling ratio is large. Many $\Delta\Sigma$ A/D converters break the decimation process into a sinc^K filter stage that decimates by a large factor M_1 (i.e., typically 8–64), followed by an FIR (or IIR) narrow-band filtering stage that decimates by a smaller factor M_2 (i.e., typically 2–8). Similarly, a $\Delta\Sigma$ D/A converter may include a narrow-band filter that interpolates the input data by a small factor L_2 , followed by a sinc^K filter that interpolates the data by a larger factor L_1 . Filtering flexibility can be accomplished by simply using a programmable DSP to do the narrow-band filtering and decimation or interpolation, that is, the M_2 or L_2 stages. An additional degree of freedom can be obtained by designing the sinc^K filter stages having a programmable decimation or interpolation factor M_1 or L_1 .

In Section 13.5.1, we covered decimation and interpolation using sinc^K filters. These were also covered in Section 1.4.3, which also included material related to hardware implementation. It has been shown that hardware structures that implement sinc^K filters can have a very simple architecture composed of adders and registers [7, 8]. Such structures are very efficient; that is, they consume relatively little chip area. They can be made to have a programmable decimation or interpolation factor by using a programmable clock divider to generate the sample clocks. The output word size is larger than the input by a factor of $K \log_2 M$ bits. For this reason, a programmable shift register may be incorporated for shifting the output word so that the sign bit is correctly aligned.

13.7.4.3 Data Transfers and Buffering Between the Stages. In Section 13.6.3, we showed that data buffers are generally needed between stages in a multistage design. We will now clearly show the importance of this concept with regard to a multistage decimation and interpolation architecture incorporating a programmable DSP [33]. Figure 13.24 illustrates an example of such an architecture. First, let us consider the case without the data buffers. For the A/D path, the DSP receives output samples of the sinc^K decimator at an intermediate sample rate M_2F . Assuming the DSP does not incorporate DMA (direct memory access), an interrupt may be generated every time a new data sample is available for the DSP. (Alternatively, *polling* may be used in lieu of interrupts. We will cover issues regarding polling later.) There is an overhead penalty for servicing these interrupts, since the DSP must first save the contents of its pointers and registers prior to branching out of the routine it was running at the time the interrupt occurred. The DSP must then fetch the data sample, write it to a memory location, branch out of the interrupt service routine, and return. After the DSP has been interrupted M_2 times (in order to fetch M_2 data samples), it can then filter and decimate this data block into one decimated output sample. The interrupt overhead penalty can be reduced by using a data buffer between the two stages of decimation. In this case, the data buffer resides between the sinc^K decimator and the DSP. This reduction in interrupt overhead is proportional to the depth of the buffer. For example, if the buffer depth is M_2 , then the interrupt overhead is reduced by a factor of M_2 . A flag then interrupts the DSP only when the buffer is full, or after M_2 samples have been written to the buffer.

By duality, a similar problem exists on the D/A side that is alleviated by use of a data buffer. The PCM input data at a sampling rate F are interpolated by a narrow-band filter program in the DSP having an interpolation factor L_2 . The DSP writes output samples

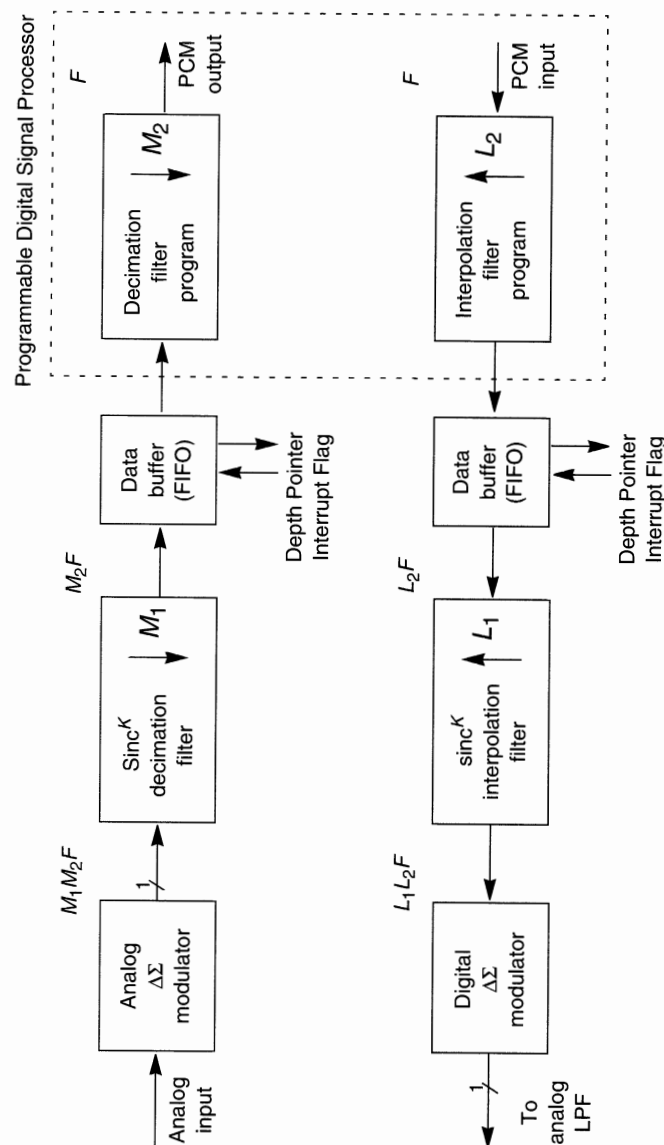


Figure 13.24 Multistage decimator and interpolator incorporating a programmable DSP with data buffering between stages.

from the interpolation program into a buffer, where the depth of the buffer is, for example, L_2 . The buffer output feeds a sinc^K interpolator having an interpolation factor L_1 . When the buffer is empty, the DSP is interrupted so the first-stage interpolator program writes a block of L_2 samples into the buffer. Therefore, the interrupt rate is just simply F . Without either a data buffer or DMA, the sinc^K interpolator would be interrupting the DSP at a rate L_2F .

We should point out that most DSPs incorporate data transfer by *polling* as well as by interrupts. When polling is used, the DSP typically waits within a loop of instructions that check the status flags of an input or output register until that register is ready to be read or written. Polling alleviates the problem of having to save the pointers and registers to service an interrupt but causes additional DSP instructions to be wasted while waiting for the I/O status flags to change accordingly. (In some rare cases, however, it is possible to time the overall DSP instructions within the application so that the data transfers are ready the first time the status flags are checked per data transfer, so that very few instructions are wasted.) The rate at which polling must occur would be M_2 or L_2 greater if the data buffers were not present, causing the number of wasted polling instructions to be M_2 or L_2 times greater.

13.7.5 Mixed Analog and Digital Implementations

It is possible to use an analog FIR filter at the output of a $\Delta\Sigma$ D/A converter [45–47]. It is also possible to include decimation in such an analog filter [47], which is analogous to digitally filtering and decimating the output of a $\Delta\Sigma$ A/D modulator. This would entail using a sampled-data analog method for implementing the filter. Two such techniques are *switched capacitor* and *switched current*. The fundamental challenge associated with applying either of these techniques is coefficient inaccuracy due to capacitor or current-mirror mismatches.

13.8 CONCLUSION

This chapter has focused primarily on the principles of sampling rate conversion and how they apply to state-of-the-art techniques for $\Delta\Sigma$ conversion. As can be seen by this discussion, sampling rate conversion plays a predominant role in the signal processing aspects of these designs and has a strong determination on the overall complexity and cost of the design of efficient A/D and D/A $\Delta\Sigma$ converters. This discussion was motivated by the very broad design perspective illustrated in Table 13.1, where the design problem was divided into four major categories: algorithmic, architectural, VLSI layout, and VLSI process. This chapter has focused primarily on the first two categories. As pointed out in the discussion, the choice of design is strongly governed by the current state of the art in VLSI design and layout, which is a strongly varying function of time compared to the theoretical and architectural issues; that is, design choices that were of interest in the 1970s and 1980s no longer lead to the most effective implementations in the 1990s. Things that were hard to do then, such as multipliers, are no longer the overriding issues in the 1990s; however the underlying theoretical principles have changed more slowly. Basic principles such as direct-form and polyphase algorithm architectures and filter design issues such as sinc,

half-band, and ternary coefficient filter designs have been discussed. Algorithm architectures, based on signal flow graph principles and multistage concepts and their trade-offs, have been discussed. Finally hardware architectures covering the range from fully dedicated hardware designs to fully programmable general-purpose DSP implementations and the trade-offs associated with implementing different stages of a multistage design with different techniques have been briefly covered.

These trade-offs are also seen to change depending on the requirements of the design; that is, the trade-offs differ depending on whether the designer is building a high-quality audio (CD grade) converter or a telephone band voiceband only converter where bandwidth and dynamic range are not nearly so severe. The choice of design technique and the trade-offs, as outlined in Table 13.1, differ with application as well as tools, availability, and evolution of the state of the art in layout and process technology. We hope that this discussion has helped to clarify these design issues and is useful in guiding design engineers in choosing the right combination of trade-offs and design choices that are well matched to current state-of-the-art technologies.

ACKNOWLEDGMENTS

The authors especially thank Mohit Prasad and other anonymous reviewers for their careful review of this chapter.

REFERENCES

- [1] J. C. Candy and G. C. Temes, eds., *Oversampling Delta-Sigma Data Converters*, Piscataway, NJ, IEEE Press, 1992.
- [2] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Englewood Cliffs, NJ, Prentice-Hall, Englewood Cliffs, NJ, 1983.
- [3] C. W. Farrow, "A continuously variable digital delay element," *IEEE Proc. ICASSP'88*, pp. 2641–2645, April 1988.
- [4] T. Saramaki and H. Tenhunen, "Efficient VLSI-realizable decimators for sigma-delta analog-to-digital converters," *IEEE Proc. ISCAS'88*, pp. 1525–1528, June 1988.
- [5] D. R. Welland et al., "Stereo 16-bit delta-sigma A/D converter for digital audio," *J. Audio Eng. Soc.*, vol. 37, pp. 476–486, June 1989.
- [6] J. C. Candy, "Decimation for sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-34, pp. 72–76, Jan. 1986.
- [7] E. B. Hogenaar, "An economical class of digital filters for decimation and interpolation," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-29, pp. 155–162, April 1981.
- [8] E. Dijkstra, O. Nys, C. Piguat, and M. Degrauwe, "On the use of modulo arithmetic comb filters in sigma delta modulators," *IEEE Proc. ICASSP'88*, pp. 2001–2004, April 1988.
- [9] S. Park, "Multistage decimation filter design technique for high-resolution sigma-delta A/D converters," *IEEE Trans. Instr. Measurement*, vol. 41, no. 6, pp. 868–873, Dec. 1992.

- [10] *QEDesign 1000 Digital Filter Design and Analysis System*, Momentum Data Systems.
- [11] M. G. Bellanger, "Computation rate and storage estimation in multirate digital filtering with half-band filters," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-25, no. 4, pp. 344–346, Aug. 1977.
- [12] D. J. Goodman and M. J. Carey, "Nine digital filters for decimation and interpolation," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-25, no. 2, pp. 121–126, April 1977.
- [13] P. W. Wong and R. M. Gray, "FIR filters with sigma-delta modulation encoding," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-38, pp. 979–990, June 1990.
- [14] J. W. Scott, "Multiplier-free interpolation for oversampled digital-to-analog conversion," presented at the 92nd Convention of the Audio Engineering Society, Vienna, March 1992, preprint 3317.
- [15] N. Benvenuto, L. E. Franks, and F. S. Hill, "Dynamic programming methods for designing FIR filters using coefficients -1 , 0 , and $+1$," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-34, pp. 785–792, Aug. 1986.
- [16] S. Ghanekar, S. Tantaratana, and L. E. Franks, "Design and architecture of multiplier-free FIR filters using periodically time-varying ternary coefficients," *IEEE Trans. Circuits. Syst. I: Fund. Theory Appl.*, vol. 40, no. 5, pp. 364–370, May 1993.
- [17] J. C. Candy, B. A. Wolley, and O. J. Benjamin, "A voiceband codec with digital filtering," *IEEE Trans. Commun.*, vol. COM-29, pp. 815–830, June 1981.
- [18] V. Friedman et al., "A bit-slice architecture for sigma-delta analog-to-digital converters," *IEEE Trans. Selected Areas Commun.*, vol. 6, no. 3, Apr. 1988.
- [19] AT&T CSP1027 codec data sheet, available through AT&T Microelectronics.
- [20] E. Dijkstra et al., "Wave digital decimation filters in oversampled A/D converters," *IEEE Proc. ISCAS'88*, pp. 2327–2330, June 1988.
- [21] T. W. Parks and C. S. Burrus, *Digital Filter Design*, John Wiley & Sons, New York, 1987.
- [22] K. Steiglitz, T. W. Parks, and J. F. Kaiser, "METEOR: a constraint-based FIR filter design program," *IEEE Trans. Signal Proc.*, vol. 40, no. 8, pp. 1901–1909, Aug. 1992.
- [23] L. B. Jackson, J. F. Kaiser, and H. S. McDonald, "An approach to the implementation of digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, Sept. 1968.
- [24] A. Peled and B. Liu, "A new hardware realization of digital filters," *IEEE Trans. Acoust. Speech Signal Proc.*, vol. ASSP-22, Dec. 1974.
- [25] C. Caraiscos and B. Liu, "Bit-serial VLSI implementations of FIR and IIR digital filters," *IEEE Proc. ISCAS*, May 1983.
- [26] A. Fettweis and J. A. Nossek, "Sampling rate increase and decrease in wave digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-29, Dec. 1982.
- [27] A. Peled and B. Liu, "A new approach to the realization of nonrecursive digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 477–484, Dec. 1973.
- [28] O. E. Agazzi et al., "A digital signal processor for an ANSI standard ISDN transceiver," *IEEE J. Solid-State Circuits*, vol. 24, pp. 1605–1613, Dec. 1989.
- [29] D. L. Dutweiler and Y. S. Chen, "A single-chip VLSI echo canceller," *Bell Syst. Tech. J.*, no. 59, pp. 149–160, 1980.

- [30] AT&T DSP16C data sheet, AT&T Microelectronics.
- [31] Motorola DSP56156 and DSP56166 data sheets, Motorola Semiconductor.
- [32] P. Denyer and D. Renshaw, *VLSI Signal Processing, A Bit-Serial Approach*, Addison-Wesley, Reading, MA, 1985.
- [33] R. Hartley and P. Corbett, "Digit-serial processing techniques," *IEEE Trans. Circuits Syst.*, vol. 37, no. 6, June 1990.
- [34] R. Hartley and P. Corbett, "Use of digit-serial computation in systolic arrays," *IEEE Trans. Circuits Syst. II*, vol. 39, no. 1, pp. 62–65, Jan. 1992.
- [35] P. T. R. M. Owens, and M. J. Irwin, "Digit serial multipliers," *J. Parallel Distrib. Comput.*, vol. 11, no. 2, pp. 156–162, 1991.
- [36] N. Bergmann, "A case study of the F.I.R.S.T. silicon compiler," in R. Bryant, ed., *Third CALTEC Conference on Very Large Scale Integration*, Springer, 1983.
- [37] R. Hartley and P. Corbett, "A digit-serial silicon compiler," *Proc. 25th IEEE Design Automation Conf.*, pp. 646–649, Aug. 1988.
- [38] H. De Man et al., "CATHEDRAL-II—a computer-aided synthesis system for digital signal processing VLSI systems," *Computer-Aided Eng. J.*, vol. 5, no. 2, p. 55–66, April 1988.
- [39] J. F. Cavanagh, *Digital Computer Arithmetic*, McGraw-Hill, New York, 1984.
- [40] F. Ferro and K. Ulery, "The architecture and programming of the DSP16 digital signal processor," AT&T Microelectronics application note.
- [41] R. N. Kershaw et al., "A programmable digital signal processor with 32b floating point arithmetic," *ISSCC Dig. Tech. Pap.*, pp. 92–93, Feb. 1985.
- [42] Y. Matsuya, K. Uchimura, A. Iwata, et al., "A 16-bit oversampling A-to-D conversion technology using triple-integration noise shaping," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 921–929, Dec. 1987.
- [43] T. Okamoto, Y. Maruyama, and K. Hinooka, "A 16 b oversampling codec with filtering DSP," *ISSCC Dig. Tech. Pap.*, pp. 74–75, Feb. 1991.
- [44] S. R. Norsworthy, L. E. Bays, and J. Fischer, "A programmable CODEC signal processor," *ISSCC Dig. Tech. Pap.*, Feb. 1996.
- [45] J. W. Scott and T. R. Viswanathan, "Integral switched capacitor FIR filter/digital-to-analog converter for sigma-delta encoded digital audio," U. S. Patent 5,012,245, assigned to AT&T Bell Laboratories, issued April 1991.
- [46] P. J. Hurst and J. E. C. Brown, "Finite impulse response switched-capacitor filters for the delta-sigma modulator D/A interface," *IEEE Trans. Circuits Syst.*, vol. 38, no. 11, Nov. 1991.
- [47] H. S. Fetterman, "Sigma delta digital-to-analog converter frequency smoothing using a filtering switched-capacitor-3-level converter," Masters Thesis, Lehigh University, Sept. 1990.

CAD for the Analysis and Design of $\Delta\Sigma$ Converters

Christopher Wolff
John G. Kenney
L. Richard Carley

14.1 INTRODUCTION

In $\Delta\Sigma$ modulator design, the need for computer support is acute. Since few modulator topologies have been analyzed exactly, analytical expressions for modulator behavior and performance are generally not available. Some approximate analytical methods do exist, but simulation is still needed to verify the predictions of the analytical methods.

This chapter describes the use of CAD in both modulator design and simulation. To open the discussion, Section 14.2 shows how the design of a multibit modulator can be formulated as an optimization problem that is then solved numerically. The remainder of the chapter deals with the issue of modulator simulation. Section 14.3 describes the use of difference-equation simulation, Section 14.4 shows the utility of simulation based on the quantizer transfer function, and Section 14.5 outlines other approaches to modulator simulation.

14.2 MULTIBIT CONVERTER DESIGN

In the context of $\Delta\Sigma$ modulation, multibit quantization offers the following advantages over single-bit quantization:

1. The quantization noise is lower (by 6 dB for every additional bit; more if optimal NTFs are used).
2. The white-noise model is more accurate (tonal behavior is less likely).
3. The loop is easier to stabilize (quantizer overload can be completely avoided).
4. For $\Delta\Sigma$ DACs, the analog postfilter has less out-of-band noise to remove and need not deal with signals that change as rapidly as in those in single-bit designs.

The principal disadvantage of multibit quantizers is that a simple m -level DAC is not inherently linear unless $m = 2$. Such nonlinearity errors are not attenuated by the loop filter and contribute directly to modulator nonlinearity. The techniques described in Chapter 8, such as digital correction [1] and dynamic element matching [2, 3], can be used to mitigate the nonlinearity.

In this section, the design of high-order multibit noise-shaping coders will be developed. The specific problem that is addressed is how to place the poles of the NTF so that the SNR is maximized while guaranteeing that the system does not overload. In multibit systems, it is possible to avoid overloading the quantizer by ensuring that the input signal plus the accumulation of quantization errors does not exceed the full-scale range of the quantizer. An overload criterion based on the l_1 norm of the NTF will be described. This methodology, known as CLANS (Closed-Loop Analysis of Noise Shapers), places the closed-loop poles of the noise-shaping coder by minimizing the in-band quantization error, while guaranteeing that the closed-loop poles are placed inside a user-defined radius in the z -plane and that the l_1 norm of the NTF does not exceed a user-defined specification.

14.2.1 Accumulation of Quantization Error

The modulator model that will be used in this section is the universal model of Figure 4.2, where, for the sake of simplicity, the STF $G(z)$ is assumed to be 1. The output of the quantizer is then $V(z) = U(z) + H(z)E(z)$, and its input is

$$Y(z) = V(z) - E(z) = U(z) + (H(z) - 1)E(z) \quad (14.1)$$

Since $h(0) = 1$, the time-domain version of Eq. (14.1) is

$$y(n) = u(n) + \sum_{i=1}^{\infty} h(i)e(n-i) \quad (14.2)$$

If the spacing of the quantization levels is assumed to be $\Delta = 2$ and the quantizer was not overloaded at any time before time n ,

$$|e(n-i)| \leq \Delta/2 = 1 \quad (14.3)$$

Thus,

$$\begin{aligned} |y(n)| &\leq |u(n)| + \left| \sum_{i=1}^{\infty} h(i)e(n-i) \right| \\ &\leq |u(n)| + \sum_{i=1}^{\infty} |h(i)e(n-i)| \\ &\leq |u(n)| + \sum_{i=1}^{\infty} |h(i)||e(n-i)| \\ &\leq |u(n)| + \sum_{i=1}^{\infty} |h(i)| \end{aligned} \quad (14.4)$$

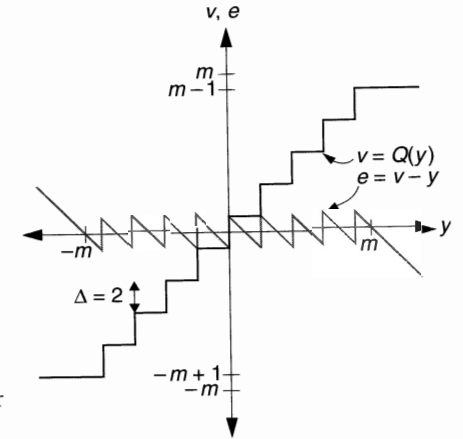


Figure 14.1 An ideal m -level quantizer with a level spacing $\Delta = 2$.

Since the quantizer will not overload at time n if $|y(n)| \leq m$ (see Figure 14.1), the following conditions are sufficient to guarantee that the quantizer will never overload:

1. The initial conditions of the modulator are consistent with the assumption that the modulator was not overloaded before time 0.
2. The input signal is bounded: $\|u\|_{\infty} \equiv \max(|u(n)|) < \infty$.
3. The modulator NTF is such that

$$\sum_{i=1}^{\infty} |h(i)| \leq m - \|u\|_{\infty} \quad (14.5)$$

This equation is equivalent to Eq. (4.24) if the outer quantization levels are scaled from their present value of $m - 1$ to 1.

By virtue of the fact that $h(0) = 1$,

$$\sum_{i=1}^{\infty} |h(i)| = \|h\|_1 - 1$$

where $\|h\|_1$ is the l_1 norm (the sum of the absolute values) of the impulse response of the NTF. Together with Eq. (14.5), this shows that the quantity $\|h\|_1 - 1$ determines the fraction of the m -level DAC's dynamic range that is consumed by the accumulation of quantization errors; the remainder is free to be consumed by the signal. In the optimization problem that will be described next, the maximum accumulation of quantization errors will be used to guarantee that the modulator never overloads, thereby ensuring its stability.

14.2.2 Formulation and Solution of the Optimization Problem

The objective of this section is to present a methodology for determining the closed-loop poles in the NTF of a high-order multibit $\Delta\Sigma$ modulator. The key element in this

methodology is the formulation of an optimization problem. We have successfully designed high-order loops using the nonlinear optimizer in MATLAB [4] and have considerable experience with the nonlinear optimization package NPSOL [5]. An example of the methodology implemented in MATLAB can be found in Appendix A of a previously published work [6]. Readers may choose to tailor the optimization problem to their specific requirements by including such features as the sensitivity of the NTF poles to the precision of the loop filter coefficients.

There are four inputs used by CLANS. They are as follows:

1. The order of the loop filter, N
2. The maximum radius of the NTF poles, $r_{\max}$
3. The oversampling ratio, R
4. The maximum accumulation of the quantization errors, $Q = \|h\|_1 - 1$.

14.2.2.1 Quantization Noise. The objective function for the minimization process is the quantization noise power. Assuming that the quantization noise $e(n)$ is white and uniformly distributed on $[-1, 1]$ and that the modulator is followed with a low-pass filter with a frequency response $F(\omega)$, the quantization noise power is

$$P_q = \frac{1}{3\pi} \int_0^\pi |H(e^{j\omega})F(\omega)|^2 d\omega \quad (14.6)$$

This equation is impractical to implement in an optimization methodology since it involves integrating over a continuous variable while the transfer functions of $H(z)$ and $F(\omega)$ are not fixed. However, it does suggest an approach for optimizing high-order noise-shaping coders.

The first step toward determining a practical definition of the objective function is to specify $F(\omega)$, which heavily depends on whether the application is an ADC or a DAC. For example, in a DAC, $F(\omega)$ could be the combination of a switched-capacitor filter followed by a low-complexity continuous-time filter or it could be a high-order (e.g., fifth-order) classical continuous-time filter. The low-pass filter in an ADC can also take a variety of forms [7]. Two simple strategies that we have successfully employed are to define $F(\omega)$ as a low-pass brick-wall filter with a cutoff at π/R [8] or to define $F(\omega)$ as a high-order IIR filter such as a Butterworth filter [6]. Once the coefficients and topology of the postfilter are defined, the frequency response $F(\omega)$ can be entirely determined.

Due to the complexity of both $F(\omega)$ and $H(z)$, a closed-form solution to Eq. (14.6) is not practical. Instead of evaluating this integral, we choose to sample the integrand along an evenly spaced grid of frequency points between 0 and π and approximate the integral with a sum. Since this operation involves the sampling of a frequency response, it is important that the number of points is large enough that aliasing of the impulse response is avoided. If $F(\omega)$ is assumed to be a brick-wall filter, good designs can be achieved by minimizing $H(e^{j\omega})$ at $\omega = \pi/R$ [8].

14.2.2.2 Constraints. The first and possibly most important constraint is the maximum accumulation of quantization errors, Q . Since the largest input that is guaranteed to yield stable behavior is $\|u\|_\infty = m - Q$, a low value of Q implies a large input

range. On the other hand, a high value for Q generally leads to a larger peak SNR because it allows the use of more aggressive NTFs. In fact, were it not for the presence of electronic noise, the value of Q that maximizes the peak SNR would be very close to m .

Assuming that the electronics noise has a power P_e and that it combines additively with the quantization noise whose power is P_q , the SNR for a sine wave of amplitude $\|u\|_\infty$ is

$$\text{SNR} = \frac{\|u\|_\infty^2}{2(P_q + P_e)} = \frac{(m - Q)^2}{2(P_q + P_e)} \quad (14.7)$$

If the reciprocal of SNR is used as the objective function, Q would not need to be specified in advance—it would be determined automatically. However, it has been our experience that this objective function does not converge well to a final solution. For this reason, Q is left as a design parameter.

14.2.2.3 Representation of Closed-Loop Poles for the Optimizer. The output of CLANS is the z -domain poles specified using polar notation; that is, the two variables describing a pole are its radius r_i and its angle θ_i . We have found that this is not the most desirable representation of the poles for the optimization problem. One of the inputs to CLANS is a maximum radius $r_{\max}$ at which the poles can be placed. The z -transform specification of a biquad is $A(z) = 1 + a_1 z^{-1} + a_2 z^{-2}$. When the poles are in complex conjugate pairs, the stability margin can be tested by verifying that the coefficient a_2 is less than $r_{\max}^2$. However, a simple test for the stability margin cannot be applied if $A(z)$ factors into two real poles.

To make the $r_{\max}$ constraint easy to apply, CLANS computes the poles using a bilinear transformation:

$$z = r_{\max} \frac{1 + s}{1 - s} \quad (14.8)$$

This transformation maps the $j\Omega$ axis in the s -domain onto a circle of radius $r_{\max}$ in the z -domain; left-half-plane poles map inside a circle whose radius is $r_{\max}$. To make it possible for the optimizer to make a smooth transition between complex conjugate poles and two real-axis poles, the poles in the s -domain are represented in terms of natural frequencies and damping ratios. That is, they are the roots of the polynomial $s^2 + 2\zeta\omega_n s + \omega_n^2$. It has been our experience that optimizing the pole placement with this representation leads to more robust convergence of the optimizer.

14.2.3 Example Results

Figure 14.2 plots the poles and zeros of a pair of fifth-order NTFs whose poles were placed using the CLANS methodology and whose zeros were the optimized NTF zeros of Section 4.4.2. The optimality of the NTF zeros is predicated on the assumption that the magnitude of the denominator of $H(z)$ is constant in the band of interest. Since the oversampling ratio is not large and the NTF poles are not subject to any special constraints, this assumption needs to be verified once the NTF has been found. Simultaneous optimization of the poles and zeros may be required if the magnitude of the denominator varies significantly across the band of interest.

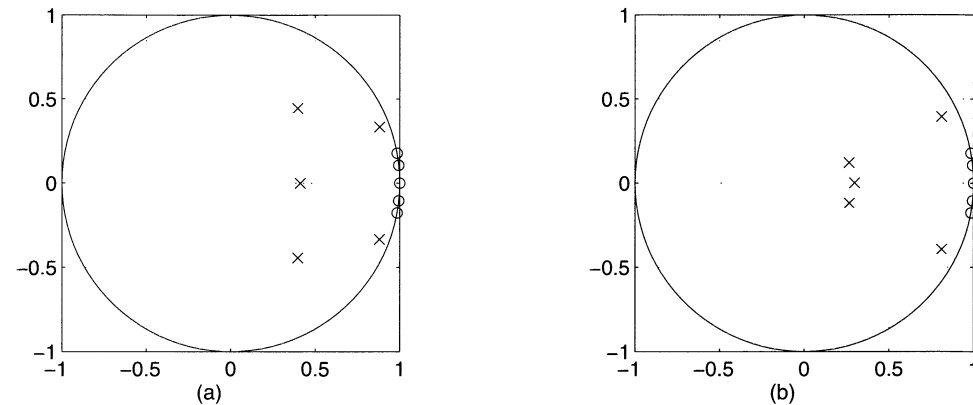


Figure 14.2 Poles and zeros of a pair of fifth-order NTFs optimized for an oversampling ratio of 16: (a) $Q = 4$; (b) $Q = 5$.

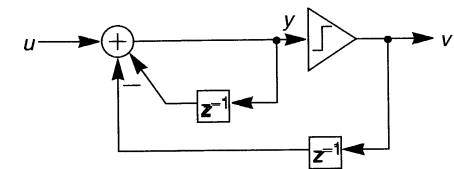
The NTF illustrated in Figure 14.2(a) used a maximum of $Q = 4$ levels for the accumulation of quantizer errors and achieved an in-band noise power of $P_q = -86$ dB; the NTF in Figure 14.2(b) used $Q = 5$ and achieved $P_q = -92$ dB. Assuming both employ an $m = 16$ -level DAC and that electronic noise is negligible, the peak SNRs are guaranteed to be at least 104 and 109 dB, respectively, for the two modulators. However, if the electronic noise is -110 dB below the internal DAC's full-scale value ($10 \log_{10}(m-1) = 24$ dB), the SNR of both systems will be limited to about 102 dB. Lower Q values will increase the allowable input magnitude, but this gain will be swamped by a large increase in the quantization noise power ($P_q = -75$ dB for $Q = 3$). Although higher Q values decrease P_q , the loss in input dynamic range, coupled with the fact that P_e dominates P_q for large Q values, likewise prevents the attainment of high SNR values. Thus, the two designs of Figure 14.2 represent the best possible configurations for this system. Clearly, the CLANS methodology facilitates the design of high-performance, multibit $\Delta\Sigma$ modulators.

14.3 SIMULATION BASED ON DIFFERENCE EQUATIONS

Once an initial modulator design has been found, the designer must use simulation to verify the modulator's performance. The most direct method is to simulate the modulator in terms of its difference equations. The output spectrum of the modulator can then be calculated and the SNR, or other performance measures, evaluated.

For ideal modulators, the difference equation method is very easy to implement. For example, Figure 14.3 shows a 10-line C++ program that simulates the time-domain behavior of a first-order modulator. Higher order modulators are likewise easily simulated, requiring as little as one extra line of code for each increment in the order. See Figure 14.4 for code that simulates a fifth-order modulator.

To incorporate such nonideal effects as finite (and even nonlinear) op-amp gain, offset, comparator hysteresis, and so on, more detailed code is needed. Since the difference



```
#include <stream.h>

main(){
    double u, y=0, v=0;
    while( cin >> u ){
        y += u-v;
        v = (y>=0)?1:-1;
        cout << v << endl;
    }
}
```

Figure 14.3 An ideal first-order $\Delta\Sigma$ modulator and C++ code that simulates it.

equations must be derived afresh for every new topology and/or nonideal effect, and since such derivations are usually manual, the difference equation approach is cumbersome and error prone for all but the most simple systems.

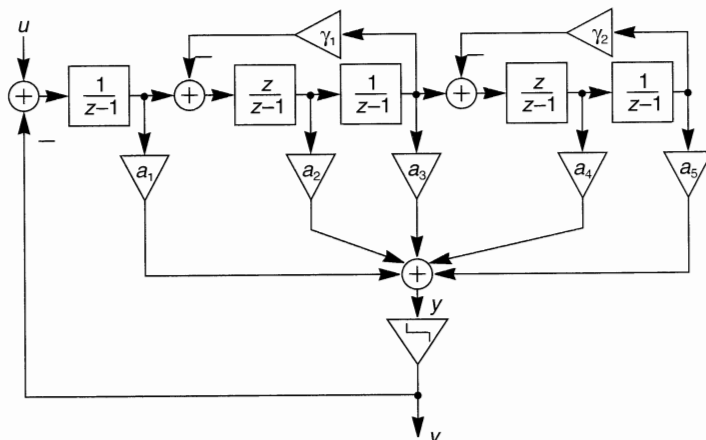
Computer-assisted, analytical methods have been developed to estimate four of the most common parameters of $\Delta\Sigma$ modulators, namely SNR, distortion, swing range, and stability. In the next section we will present one method for modeling the $\Delta\Sigma$ modulator using a 1-bit quantizer and discuss how the above parameters are efficiently computed using our approximate model.

14.4 SIMULATION BASED ON THE QUANTIZER TRANSFER FUNCTION

Many studies of single-bit $\Delta\Sigma$ modulators have been made using both analytical and empirical techniques [9–12]. Unfortunately, the analytical work often suffers from unrealistic assumptions or only works for a specific topology such as an ideal first-order loop. In this section, a statistical average transfer function model for the quantizer is presented and used to determine the performance of a $\Delta\Sigma$ modulator. This model can be used to predict the performance (SNR), the distortion, and the stability of the loop as a function of the input signal. In addition, this model provides substantial insight into the behavior of the modulator and can help to guide design improvement.

14.4.1 Statistical Average Quantizer Transfer Function

On any given clock cycle, the output from a quantizer takes on one of a few possible values (two values for the 1-bit case); therefore, it is extremely nonlinear. However, if we were to observe the loop in operation over a large number of clock cycles, we could measure the average input and average output of the quantizer. By varying a dc input and repeating this experiment, we can define a statistical average transfer function for the



```
#include <stream.h>

main(){
    const double a1=.56, a2=.25, a3=.054, a4=.0084, a5=55e-5,
               g1=7e-4, g2=2e-3;
    double x1=0, x2=0, x3=0, x4=0, x5=0;
    double u, v=0, y;
    while( cin >> u ){
        // x1,x3,x5 are at time n; v,y,x2,x4 are at time n-1.
        x2 += x1-g1*x3;
        x4 += x3-g2*x5;
        y = a1*x1 + a2*x2 + a3*x3 + a4*x4 + a5*x5;
        v = (y>=0)?1:-1;
        x1 += u-v;
        x3 += x2;
        x5 += x4;
        // x1,x3,x5 are at time n+1; v,y,x2,x4 are at time n.
        cout << v << endl;
    }
}
```

Figure 14.4 A fifth-order modulator (Figure 5.7) and C++ code that simulates it.

quantizer in a $\Delta\Sigma$ modulator. In this section we discuss the ways in which this quantizer transfer function (QTF) model can be used.

The QTF can be used for a number of purposes, including the prediction of the modulator's distortion. To understand the assumptions used in the QTF model, consider that oversampled converters have, by nature, a clock that operates at a much greater frequency than the signal frequencies of interest because the oversampling ratio is often large (e.g., 64 times or more). For first-order $\Delta\Sigma$ modulator loops, a comparison of dc and sinusoid inputs using exact analysis [13] has shown that this quasi-static approximation is valid for large oversampling ratios. Thus, one useful approximation of the $\Delta\Sigma$ loop is that the input is a dc signal at the time scale of the clock. If the modulator's oscillations are treated as

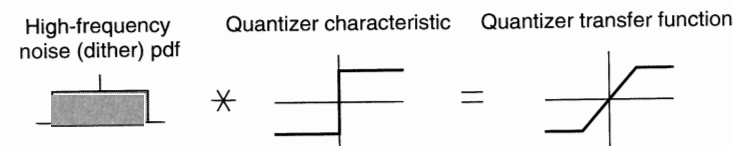


Figure 14.5 Low-frequency model of the quantizer assuming the input is the sum of high-frequency noise and a low-frequency signal.

high-frequency noise, a QTF can be computed. The QTF can be found without assuming that the pdf of the high-frequency noise is independent of the input or that it is white. The QTF can be estimated with a simulator or calculated directly using a Markov model of the $\Delta\Sigma$ loop [15, 17, 23].

From control theory, a memoryless nonlinearity whose input is the sum of a low-frequency input signal and a high-frequency noise can be modeled as a memoryless nonlinearity given by the convolution of the original nonlinearity with the pdf of the high-frequency noise [14] (see Figure 14.5). In a $\Delta\Sigma$ modulator, the high-frequency oscillations smooth the dc transfer function of the quantizer from its ideal shape (i.e., a step for a single-bit quantizer) in a similar fashion. However, in the case of a $\Delta\Sigma$ modulator the pdf of the loop noise depends on the dc input level; therefore, we must directly determine the QTF rather than determine the pdf of the noise and convolve it with the stepwise nonlinearity that represents the quantizer.

As covered in detail in Chapter 3, the appropriate addition of dither can destroy periodic oscillations and whiten the quantization noise. Therefore, it is possible to consider the effective dithered quantizer transfer function to be a smoothed, sigmoidal nonlinearity [16, 17], independent of the initial conditions of the $\Delta\Sigma$ loop.

As illustrated in Figure 14.6, the QTF can be determined by running a set of simulations that have different dc input values and measuring the mean quantizer input signal in each case. The average quantizer output will equal the dc input because the loop normally has nearly infinite gain at dc.

The shape of the QTF, that of the pdf of the high-frequency noise components, and the range of stability for $\Delta\Sigma$ loops have previously been demonstrated using simulation studies and experimental measurements [15, 17]. Starting a $\Delta\Sigma$ simulation with different initial conditions on the state variables in the loop filter can produce a different set of limit cycles [18], but this does not affect the calculation of the QTF [17] as long as some dither (random) signal is included and the average is taken over sufficiently many cycles. One simple method for determining if sufficient averaging is being performed is to start the modulator in different states and compare the resulting QTFs—they should be the same. The response of the modulator at low frequencies can be observed by replacing the quantizer and the high-frequency loop noise with the QTF smoothed nonlinearity [27] as shown in Figure 14.7.

14.4.2 Distortion

Distortion is caused by nonlinearities in the $\Delta\Sigma$ loop, such as op-amp gain nonlinearity, slew rate limiting, capacitor voltage dependence, and quantizer nonlinearity. Circuit simulation of component blocks can be used to predict the nonlinearity of the loop filter's

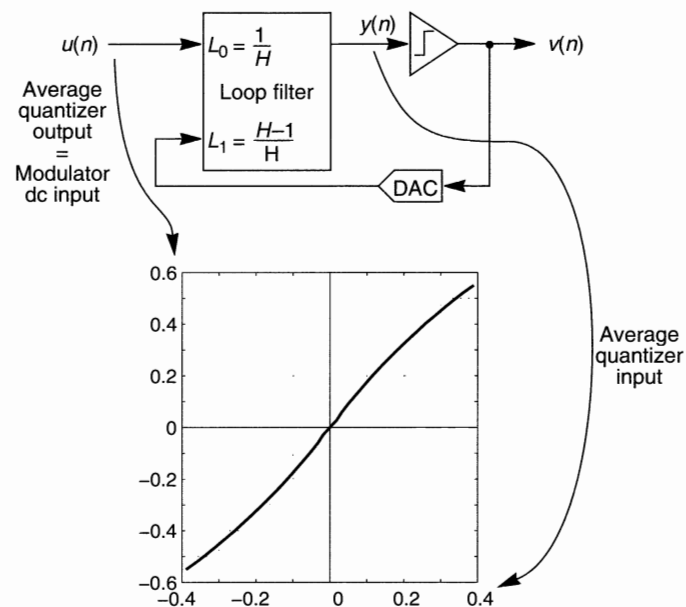


Figure 14.6 Determination of the QTF. The QTF of the sixth-order example modulator of Chapter 4 is shown as an example.

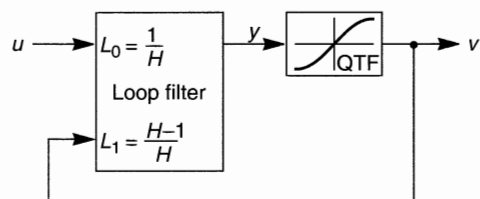


Figure 14.7 A low-frequency model of the modulator. The low-frequency gain of the STF is assumed to be unity.

components. Several authors have presented methods that predict harmonic distortion for switched-capacitor filters as caused by nonideal components [19, 20]. These methods have been used to derive analytical models for harmonic distortion in $\Delta\Sigma$ modulators that use switched-capacitor loop filters [9, 21]. Like the SNR, the harmonic distortion of $\Delta\Sigma$ modulators is often estimated by difference equation simulators or switched-capacitor simulators [22–24]. Other switched-capacitor simulators [25, 26] cannot be used because of the aperiodic nature of $\Delta\Sigma$ modulators.

Even if the loop filter's components are ideal, the modulator will exhibit distortion because the QTF is nonlinear. If the nonlinearity is weak, it can be modeled with a low-order polynomial, and the coefficients of the polynomial can be used to predict the modulator's distortion [27, 28]. Specifically, if the QTF is approximated by a cubic nonlinearity

$$v = k_1 y + k_3 y^3 \quad (14.9)$$

then the low-frequency behavior of the modulator may be approximated by the systems shown in Figure 14.8.

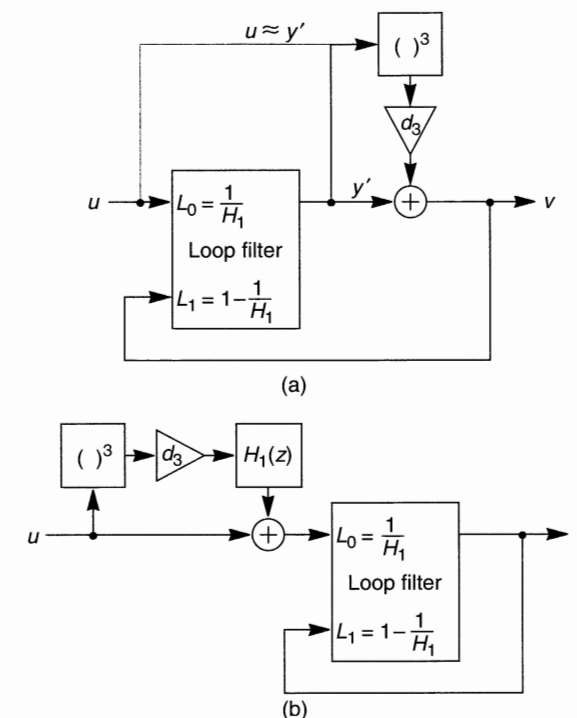


Figure 14.8 Derivation of a low-frequency model of the modulator suitable for distortion calculations.

In both systems of Figure 14.8, the linear portion of the quantizer gain has been absorbed into the loop filter, so that the new NTF is (cf. Section 4.2)

$$H_1 = \frac{H}{k_1 + (1 - k_1)H} \quad (14.10)$$

and the new quantizer input is $y' = y/k_1$. The low-frequency STF is still assumed to be unity. Thus, for low frequencies and as a first-order approximation, $y' \approx v \approx u$. By Eq. (14.9)

$$v = y' + d_3(y')^3 \approx y' + d_3 u^3 \quad \text{where } d_3 = k_3/k_1^3 \quad (14.11)$$

If the distortion term is reflected to the modulator input as in Figure 14.8(b), it becomes readily apparent that quantizer nonlinearity results in noise-shaped distortion.

Since the amplitude of the third harmonic produced by a cubic nonlinearity with a sinusoidal input of amplitude A is $A^3/4$, the signal-to-third-harmonic-distortion ratio for the system shown in Figure 14.8(b) is

$$20 \log_{10} \left| \frac{A}{d_3 A^3 H_1(z)/4} \right| = -20 \log_{10} \left| \frac{d_3 A^2 H_1(z)}{4} \right| \quad (14.12)$$

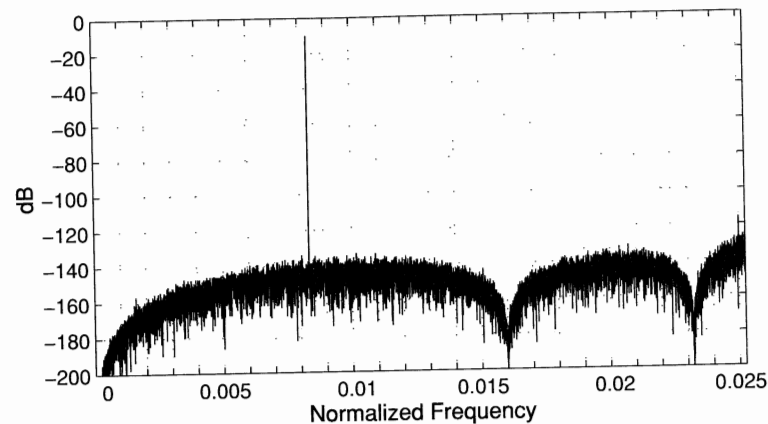


Figure 14.9 In-band spectrum of the example modulator with a -20 -dB input.

where $z = \exp(j\pi f)$ and f is the normalized frequency ($f = 1$ corresponds to $f_s/2$) of the input sinusoid. Since the maximum NTF magnitude typically occurs at the passband edge, a signal whose third harmonic lies at the passband edge will result in the most distortion.

For the example sixth-order modulator considered in Chapter 4, a polynomial fit to the QTF shown in Figure 14.6 yields a quantizer gain of $k_1 = 1.8$ and a d_3 distortion coefficient of approximately $d_3 \approx -0.6$. The gain of H_1 at the passband edge is -75 dB, and for -10 dB input the worst-case signal-to-distortion ratio estimated using Eq. (14.12) is 111 dB. Figure 14.9 plots the in-band spectrum of the plot and is 106 dB below the signal. In this manner, the QTF can be used to predict the distortion of a $\Delta\Sigma$ modulator with a sine wave input. The QTF model can also be used to estimate intermodulation distortion.

14.5 SIMULATION APPROACHES

14.5.1 Overview

At some point in the design of an analog $\Delta\Sigma$ modulator, simulation is necessary. Analytical models can provide good starting points in the design of an analog $\Delta\Sigma$ modulator, but simulation is needed to determine the effects of analog nonidealities on system performance. Performance measures such as SNR versus input amplitude require many simulation runs, one for each point on the curve. Each simulation requires 10,000–100,000 clock cycles depending on the oversampling ratio and the desired accuracy of the SNR estimate. For example, if a DFT/FFT is used for postprocessing, a $\Delta\Sigma$ modulator with $64\times$ oversampling and 1024 in-band FFT bins would require 131,072 clock cycles plus the initial cycle. Alternately, if an IIR or FIR output filter is used, many data points are required to allow for filter settling. The standard CAD tools for most analog integrated circuit simulation tasks is the circuit simulator (e.g., SPICE). To achieve an accuracy on the order of 90 dB, circuit simulators typically require 100–1000 time steps per clock cycle. Thus, over a million time steps are needed just to acquire a single data point on the SNR-versus-input-amplitude plot. Such simulations can take more than a day even on today's fastest work-

stations. Worse yet, rounding and truncation errors within the circuit simulation algorithms typically set an upper limit on the measurable SNR on the order of 90 dB. Alternative simulation approaches are needed for analog $\Delta\Sigma$ modulators that are both efficient and allow accurate prediction of SNR.

14.5.2 Model Comparison

When deciding the best simulation strategy for a particular problem, it is important to make a distinction between the modeling technique and the simulation technique. The same modeling technique might yield very different performance depending on the complexity and features of the simulator in which it is implemented. Important characteristics of a modeling technique are the following: speed of simulation, modeling capabilities, and reusability of an implemented model. Ideally, the best modeling technique would be the quickest to operate and develop, be capable of modeling any desired circuit features, and be reusable for other projects without any modification. Realistically, the goal of this comparison is to determine which model features can be combined to produce the best balance for simulating analog $\Delta\Sigma$ modulators at various stages of the design process.

The following modeling approaches are compared: device models, circuit macromodels, time-domain macromodels, finite-difference equations, table-lookup models, harmonic balance methods, and behavioral models. Because there is no standard for analog modeling terminology, a brief explanation of each is necessary for clarification. Device models are small and large signal models of active devices (e.g., MOSFETS, diodes, etc.) used in a circuit simulation such as SPICE [29]. Circuit macromodels, such as the Boyle model of an op-amp [30], are parameterized models of circuits made up of several devices and passive components. The macromodel is an equivalent circuit made from components available in a circuit simulator that is less complex than the original circuit and uses circuit specifications as model parameters. For example, the op-amp-macromodel might have parameters for slew rate or gain-bandwidth product.

Time-domain macromodels, such as the Chuang or Lin op-amp models [31, 32], are designed strictly for transient analysis and are generally not used with a circuit simulator. These models are based on a set of time-domain equations derived for a specific circuit. As is done for circuit macromodels, the time-domain equations use circuit specifications as control parameters.

Finite-difference equations are based on the z -transform of the transfer function of sampled-data circuits. For example, an ideal switched-capacitor integrator has the z -transform and difference equation descriptions

$$\frac{Y(z)}{X(z)} = \frac{z^{-1}}{1 - z^{-1}} \Rightarrow y(n) = y(n-1) + x(n-1) \quad (14.13)$$

Using difference equations of this sort results in small and efficient simulation programs. There are many examples of simulators based on finite-difference equations such as MIDAS [33] for oversampled converters and SWITCAP2 [22] for switched-capacitor circuits.

Table-lookup models use a two-step approach to modeling. The first step is to extract a table of input and output points for the original circuit with the use of a high-accuracy simulator, such as a circuit simulator. The second step is to use the stored table of points

TABLE 14.1 COMPARISON OF SEVERAL MODELING TECHNIQUES FOR DSM SYSTEMS

Model	Advantages	Disadvantages
Device models	Based on device physics	Too slow for discrete-time systems
Circuit-based macromodels	Can add model features that approach device model accuracy	Only a factor of 10 speed improvement over device models [56]
Time-domain macromodels	Produces quick simulations; can model dynamic errors	Not reusable because equations depend on load and feedback configuration
Difference equations	Simulates the quickest of all	Custom implementations not directly reusable; cannot model dynamic errors
Table-lookup models	Acceptable speedup over device models; can achieve accurate modeling of static errors	Unclear how to optimize table building; only model static errors; size of table increases with the square of the circuit states; tables are not reusable

instead of the original circuit for further transient simulations. In other words, this method compiles the circuit into a table. An implementation of table lookup for oversampled converters, ZSIM [34], has a predicted 0.5-mV error/cycle that appears to limit the simulated performance of oversampled converters to 80 dB.

Envelope following, or the harmonic balance method [35], simulates clocked circuits in which the waveform is similar from cycle to cycle. However, the states of an oversampled converter can change significantly from cycle to cycle and the states are not periodic. Hence, the simulation of oversampled converters is an inappropriate target for the harmonic balance method. This method will not be considered further.

While there may be several reasons for choosing one type of model over another, the main criteria for $\Delta\Sigma$ modulators are speed of simulation, modeling capabilities, and reusability. The amount of time it takes to run a simulation is dependent on both the type of model and the way the simulator was implemented. A hidden cost in many models is the time it takes the user to construct models for new circuits. Reusability refers to being able to quickly modify an existing model, perhaps from a library of models, for a new circuit. For example, a circuit-based macromodel of an op-amp can be reused for many kinds of op-amps by only changing the parameters of the circuit, whereas table-lookup models must be rebuilt for any change in the circuit.

From Table 14.1, it appears that a good modeling approach for $\Delta\Sigma$ modulators would be to take advantage of the modeling features and reusability of circuit-based macromodels and the speed of finite-difference equations and time-domain macromodels. The behavioral modeling approach discussed in the next section is a combination of these techniques.

14.5.3 Efficient Macromodel Simulation

In this section we describe the capabilities of a compiled behavioral modeling approach that was developed specifically for $\Delta\Sigma$ modulators. The heart of the method is to split a circuit with multiple nonlinearities into a set of linear circuits that are solved explicitly before simulation (i.e., compiled). During simulation, the program determines the correct linear circuit to use and applies the presolved equation to calculate the next piece of the transient response.

Time-domain macromodels are essentially a set of equations derived for a particular circuit and its loading conditions. Unfortunately, this means that the macromodel equations must be derived every time the loading conditions change. In the behavioral approach, circuit macromodels can be mixed arbitrarily with other components since the state-variable equations of the network are extracted automatically during precompilation. In this way, the op-amp circuit macromodel does not have to be changed when the feedback network around it is altered. Also, any existing circuit macromodels or templates can be used directly in the behavioral model. In addition, difference equations can be mixed as desired with the other parts of the model. Another important aspect of behavioral models is that there are few fundamental limits to the model. Unlike using difference equations alone, there are fewer restrictions on what can be modeled. Thus, behavioral modeling can combine time-domain macromodels, circuit macromodels, and finite-difference equations.

There are three steps in compiled behavioral modeling: circuit partitioning, circuit compilation, and macromodeling. First, the circuit is partitioned into smaller pieces with a few piecewise-linear (PWL) blocks and any number of linear circuit elements. Second, the circuit is compiled by solving for the state-variable equations, significantly reducing the transient simulation time. Finally, the state-variable equations are called as functions at each time step during the transient simulation. If the circuit or function to be modeled contains nonlinearities, they must be piecewise linear. The idea behind PWL simulation is that if a circuit is restricted to any particular segment of the PWL function, the circuit can be solved as a linear circuit [36]. During simulation, the program determines the correct linear circuit to use and applies the presolved equation to calculate the next piece of the transient response.

As an example of the compiled behavioral modeling technique, an op-amp model will be discussed. Circuit macromodels for op-amps are plentiful in the literature and provide a good base for writing behavioral models [30, 37–39]. A single circuit macromodel is unlikely to model every possible technology and configuration. Instead, different circuit macromodels can be used, such as for different technologies (e.g., CMOS, bipolar or GaAs) and topologies (e.g., one-stage, two stage, or folded cascode). An example op-amp macromodel is shown in Figure 14.10.

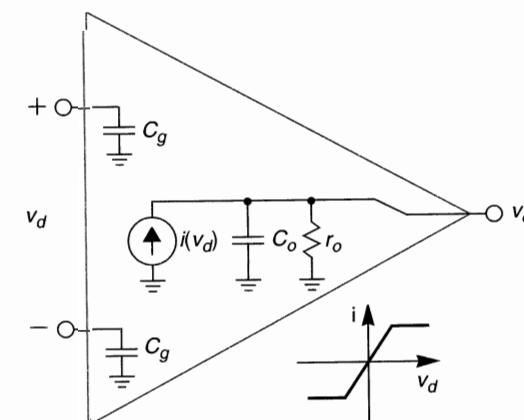


Figure 14.10 An operational transconductance amplifier (OTA) macromodel.

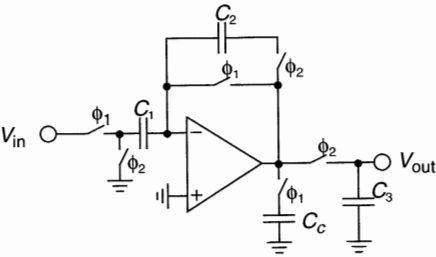


Figure 14.11 An offset-canceling switched-capacitor integrator.

To demonstrate how the modeling is done, consider an integrator designed for use in a high-speed $\Delta\Sigma$ modulator (see Figure 14.11). The integrator consists of a single-stage op-amp and a feedback network that includes offset cancellation. In this case, there are two clock phases and a PWL block with three segments, resulting in six equivalent circuits for the behavioral model. Figure 14.12 shows an equivalent circuit during phase 1, when the current source is in the linear mode, and Figure 14.13 depicts the qualitative time-domain behavior that results from such models.

To illustrate the differences between the various modeling techniques, consider a first-order switched-capacitor $\Delta\Sigma$ modulator. In order to distinguish the CPU time of the simulation method from such factors as parse time and compile time, the difference equation and the behavioral simulators differed only in the description and processing of the $\Delta\Sigma$ modulator loop itself (a program was developed that could run both methods). Default values for HSPICE accuracy tolerance (e.g., RELTOL) were used. The $\Delta\Sigma$ modulator was run for 40,960 cycles for the difference equation and behavioral methods on a DECstation

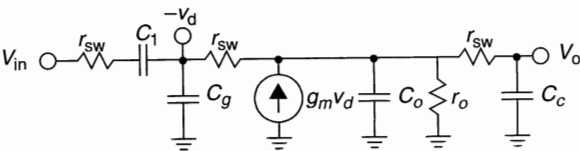


Figure 14.12 Equivalent circuit during phase 1 for the circuit in Figure 14.11, assuming the controlled current source is operating in the linear mode.

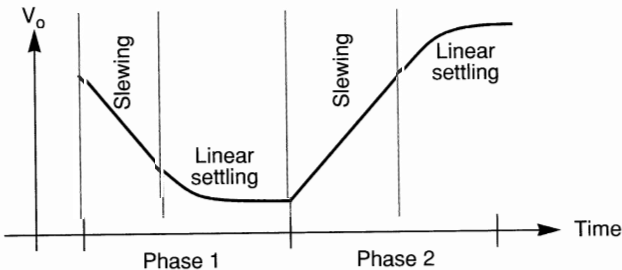


Figure 14.13 Transitions from one PWL segment to the other, as seen in the time domain.

TABLE 14.2 COMPARISON OF CPU TIME FOR THREE SIMULATORS

Simulator	Difference Equations	Behavioral	HSPICE
Extract and compile	3 sec	5 min	None
Transient simulation	14 sec	1 min, 42 sec	191 hr
Change and resimulate	17 sec	1 min, 45 sec	191 hr

3100. Because of the prohibitive CPU time required for HSPICE, only 34 clock cycles were simulated, which took 568.4 sec, roughly 16.7 sec/cycle. The estimate for a complete HSPICE run is (40,960 cycles)(16.7 sec/cycle) = 684,750 sec, or roughly 8 days. Table 14.2 compares the simulation time for the three different simulation methods.

For behavioral models, extract-and-compile time refers to how long it takes to extract the state-variable equations and compile the results. For difference equations this section refers to the compilation time of the custom C program. Because the behavioral simulator is capable of using either behavioral models or difference equations independently, the results show the direct effect of a change in models without including extraneous information such as postprocessing time and implementation efficiency. Finally, the incremental-compile section illustrates that behavioral models do not need to be reextracted for small changes to the system or for changing the input conditions.

Overall, the main advantage of using the behavioral method is that simulations take on the order of minutes instead of days for HSPICE and are not a great deal slower than difference equations. Behavioral models can include dynamic effects and are reusable. The transient noise simulation algorithm noise floor is very low (SNR at 110 dB has been simulated successfully). The main limitations are that good macromodels are time-consuming to create and no method for partitioning has been demonstrated yet despite the rapid slowing of the simulation for large numbers of PWL blocks. Also, PWL functions are only approximations to the real nonlinearities. Altogether the behavioral method provides a good simulation technique between the high-speed, low-accuracy difference equation approach and the high accuracy and low speed of circuit simulation.

Another circuit nonideality that has been addressed using a similar behavioral approach is voltage-dependent charge dump from MOS switches. This is one of the important distortion mechanisms in high-resolution ADCs implemented using switched-capacitor circuits. Trihy and Rohrer [40] have developed a general-purpose switched-capacitor circuit simulator (including $\Delta\Sigma$ modulators), called AWESwit, that can efficiently model many important circuit nonidealities such as voltage-dependent charge dump from MOS switches, finite op-amp gain, op-amp slew rate limitations, and failure to fully settle before the end of the clock period.

14.6 CONCLUSION

The modulator design process needs rapid evaluation. The use of approximate models can help reduce the number of iterations in the $\Delta\Sigma$ modulator design cycle by providing good estimates for stability, SNR, and distortion. Use of a variety of simulation tools that are

based on difference equations, behavioral models, or table-based models work well as performance predictors, leaving device-level simulation to the final verification of a $\Delta\Sigma$ design.

REFERENCES

- [1] A. Cataltepe, A. R. Kramer, L. E. Larsen, G. C. Temes, and R. H. Walden, "Digitally corrected $\Sigma\Delta$ multi-bit data converters," *Internat. Symp. on Circuits and Systems*, vol. 1, pp. 647–650, June 1989.
- [2] R. Carley and J. Kenney, "A 16-bit 4th order noise-shaping D/A converter," *IEEE Proc. Custom Integrated Circuits Conf.*, pp. 21.7.1–21.7.4, 1988.
- [3] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 267–273, April 1989.
- [4] *User's Guide for Pro-MATLAB*, MathWorks Inc., South Natick, MA, 1991.
- [5] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "User's guide for NPSOL," Department of Operations Research, Stanford Univ., 1986.
- [6] J. G. Kenney and L. R. Carley, "Design of multibit noise-shaping data converters," *Analog Integrated Circuits and Signal Processing*, Kluwer Academic Publishers, Boston, MA, May 1993.
- [7] R. E. Crochiere and L. R. Rabiner, "Interpolation and decimation of signals—a tutorial review," *Proc. IEEE*, vol. 69, pp. 300–331, March 1981.
- [8] J. G. Kenney and L. R. Carley, "CLANS: a high-level synthesis tool for high resolution data converters," *Internat. Conf. on Computer-Aided Design*, vol. 1, Nov. 1988.
- [9] M. W. Hauser and R. W. Broderon, "Circuit and technology considerations for MOS delta-sigma A/D converters," *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 3, pp. 1310–1315, 1986.
- [10] S. H. Ardalan and J. J. Paulos, "An analysis of nonlinear behavior in delta-sigma modulators," *IEEE Trans. Circuits Syst.*, vol. CAS-34, no. 6, pp. 593–603, June 1987.
- [11] V. Friedman, "The structure of limit cycles in sigma delta modulation," *IEEE Trans. Commun.*, vol. COM-36, no. 8, pp. 972–979, Aug. 1988.
- [12] R. M. Gray, W. Chou, and P. W. Wong, "Quantization noise in a single-loop sigma-delta modulator with sinusoidal inputs," *IEEE Trans. Commun.*, vol. 37, no. 9, pp. 956–968, Sept. 1989.
- [13] P. W. Wong, "Two stage sigma-delta modulation," *IEEE Trans. ASSP*, vol. 38, no. 11, pp. 1937–1952, Nov. 1990.
- [14] G. Zames and N. A. Schneidtor, "Dither in nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. AC-21, no. 5, pp. 660–667, Oct. 1976.
- [15] C. M. Wolff, "Stability analysis of delta-sigma modulators," M.S. Thesis, Carnegie Mellon University, May 1987.
- [16] D. Anastassiou, "Error diffusion coding for A/D conversion," *IEEE Trans. Circuits Syst.*, vol. 36, no. 9, pp. 1175–1186, Sept. 1989.
- [17] C. M. Wolff and L. R. Carley, "Modeling the quantizer in higher-order delta-sigma modulators," *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 3, pp. 2335–2339, 1988.
- [18] N. Bridgett and C. P. Lewis, "Effect of initial conditions on limit cycle performance of second order sampled data sigma delta modulator," *Electron. Lett.*, vol. 26, no. 12, pp. 81–83, June 1990.

- [19] K.-L. Lee and R. G. Meyer, "Low-distortion switched-capacitor filter design techniques," *IEEE J. Solid-State Circuits*, vol. SC-20, no. 6, pp. 1103–1113, Dec. 1985.
- [20] D. M. Freeman, "Slewing distortion in digital-to-analog conversion," *J. Audio Eng. Soc.*, vol. 25, no. 4, pp. 178–183, April 1977.
- [21] B. E. Boser and B. A. Wooley, "The design of sigma-delta modulation analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1298–1308, Dec. 1988.
- [22] K. Suyama and Y. Tsvividis, "Simulation of mixed switched-capacitor/digital networks with signal-driven switches," *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1403–1413, Dec. 1990.
- [23] C. M. Wolff and L. R. Carley, "Simulation of delta-sigma modulators using behavioral models," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 376–379, May 1990.
- [24] C. M. Wolff, "The analysis and simulation of delta-sigma modulators," Research Report No. CMUCAD-91-13, Ph.D. Dissertation, Carnegie Mellon University, Feb. 1991.
- [25] J. VandeWalle, J. Rabaey, W. Vercruysse, and H. De Man, "Computer-aided distortion analysis of switched capacitor filters in the frequency domain," *IEEE J. Solid-State Circuits*, vol. SC-18, no. 3, pp. 324–333, June 1983.
- [26] K. Kundert, J. White, and A. Sangiovanni-Vincentelli, "A mixed frequency-time approach for finding the steady-state solution of clocked analog circuits," *IEEE Custom Int. Circuits Conf.*, pp. 6.2.1–6.2.4, 1988.
- [27] J. K. Roberge, *Operational Amplifiers: Theory and Practice*, Wiley, New York, 1975, pp. 217–230.
- [28] A. Gelb and W. E. Vander Velde, *Multiple-Input Describing Functions and Nonlinear System Design*, McGraw-Hill, New York, 1968, pp. 72–73.
- [29] L. W. Nagel, "SPICE: A computer program to simulate semiconductor circuits," Tech. report ERL-M520, University of California—Berkeley, May 1975.
- [30] G. R. Boyle, B. M. Cohn, D. O. Peterson, and J. E. Solomon, "Macromodeling of integrated circuit operational amplifiers," *IEEE J. Solid-State Circuits*, vol. SC-9, no. 6, pp. 353–364, Dec. 1974.
- [31] C. T. Chuang, "Analysis of the settling behavior of an operational amplifier," *IEEE J. Solid-State Circuits*, vol. SC-17, pp. 74–80, Feb. 1982.
- [32] J. C. Lin and J. H. Nevin, "A modified time-domain model for nonlinear analysis of an operational amplifier," *IEEE J. Solid-State Circuits*, vol. SC-21, no. 3, pp. 478–483, June 1986.
- [33] L. Williams and B. Wooley, "MIDAS—A functional simulator for mixed digital and analog sampled data systems," *Proc. IEEE 1992 Int. Symp. Circuits Syst.*, vol. 5, pp. 2148–2151, May 1992.
- [34] G. Brauns, M. Steer, S. Ardalan, and J. Paulos, "Table-based modeling of delta-sigma modulators using ZSIM," *IEEE Trans. CAD for ICs*, vol. 9, no. 2, pp. 142–150, Feb. 1990.
- [35] K. Kundert, J. White, and A. Sangiovanni-Vincentelli, "A mixed frequency-time approach for finding the steady-state solution of clocked analog circuits," *Proc. IEEE Custom Integrated Circuits Conf.*, San Diego, CA, pp. 8.7.1–8.7.4, May 1991.

- [36] J. T. J. van Eijndhoven and W. M. G. van Bokhoven, "Piecewise linear modeling in the simulation of electronic networks," *IEEE Int. Symp. Circuits Syst.*, pp. 1206–1209, vol. 2, 1982.
- [37] B. Epler, "SPICE2 application notes for dependent sources," *IEEE Circuits Devices Mag.*, vol. 3, no. 5, pp. 36–44, Sept. 1987.
- [38] B. Perez-Verdu, J. L. Huertas, and A. Rodriguez-Vasquez, "A new nonlinear time-domain op-amp macromodel using threshold functions and digitally controlled network functions," *IEEE J. Solid-State Circuits*, vol. 23, no. 4, pp. 959–971, Aug. 1988.
- [39] C. J. Hage and R. A. Rohrer, "Efficient op amp analysis with manufacturer specified macromodel parameters," *IEEE Trans. CAD for ICs*, vol. TCAD-1, no. 3, pp. 105–112, 1982.
- [40] R. J. Trihy and R. A. Rohrer, "Simulating sigma-delta modulators in AWEswit," *ICCAD*, Nov. 1993.

FOR FURTHER READING

- [1] B. P. Agrawal and K. Shenoi, "Design methodology for sigma-delta modulators," *IEEE Trans. Commun.*, vol. COM-31, no. 3, pp. 360–369, March 1983.
- [2] B. E. Boser and B. A. Wooley, "Quantization error spectrum of sigma-delta modulators," *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 3, pp. 2331–2334, 1988.
- [3] R. M. Gray, "Spectral analysis of quantization noise in a single-loop sigma-delta modulator with DC inputs," *IEEE Trans., Commun.*, vol. 37, no. 6, pp. 588–599, June 1989.
- [4] L. Longo and M. Copeland, "A 13 bit ISDN-band oversampled ADC using two-stage third-order noise shaping," *IEEE Custom Int. Circuits Conf.*, pp. 21.2.1–21.2.4, 1988.
- [5] R. M. Gray, "Oversampled sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-35, no. 5, pp. 481–489, May 1987.
- [6] O. Feely and L. Chua, "The effect of integrator leak in $\Sigma\text{--}\Delta$ modulation," *IEEE Trans. Circuits Syst.*, vol. 38, no. 11, pp. 1293–1305, Nov. 1991.
- [7] J. C. Candy and O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. Commun.*, vol. COM-29, no. 9, pp. 1316–1323, Sept. 1981.
- [8] M. H. H. Höfelft, "On stability of a 1 bit quantized feedback system," *IEEE Int. Conf. ASSP*, pp. 844–848, 1979.
- [9] L. R. Carley, "A oversampling analog-to-digital converter topology for high-resolution signal acquisition systems," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 83–90, Jan. 1987.
- [10] M. W. Hauser, P. J. Hurst, and R. W. Broderson, "MOS ADC-filter combination that does not require precision analog components," *IEEE ISSCC Dig. Tech. Papers*, vol. 28, pp. 80–81, 1985.
- [11] E. F. Stikvoort, "Some remarks on the stability and performance of the noise shaper or sigma-delta modulator," *IEEE Trans. Commun.*, vol. 36, no. 10, pp. 1157–1162, Oct. 1988.
- [12] T. Ritoneimi, T. Karema, and H. Tenhunen, "Design of stable high order 1-bit sigma-delta modulators," *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 4, pp. 3267–3270, 1990.

- [13] K. S. Kundert, *Sparse 1.2: A Sparse Equation Solver Tailored for Node Admittance Matrices*, Univ. of California, Berkeley, CA, 1985.
- [14] B. E. Boser, K.-P. Karmann, H. Martin, and B. A. Wooley, "Simulating and testing oversampled analog-to-digital converters," *IEEE Trans. CAD for ICs*, vol. 7, no. 6, pp. 668–674, June 1988.
- [15] A. W. Drake, *Fundamentals of Applied Probability Theory*, McGraw-Hill, New York, 1967.
- [16] J. C. Candy, "A use of double integration in sigma-delta modulation," *IEEE Trans. Commun.*, vol. 33, no. 3, pp. 249–258, March 1985.
- [17] Texas Instruments, *Operational Amplifier Macromodels Data Manual*, Texas Instruments, Dallas, TX, 1990.
- [18] C. M. Wolff and L. R. Carley, "Calculating the stability range and SNR of delta-sigma modulators," in *Proc. IEEE Symp. Circuits Syst.*, vol. 2, pp. 1423–1426, 1989.

A

Accumulate-and-dump circuit, 33

Accumulator, 312

ADCs,

bandpass, 282

continuous-time design, 297–300

example, 288, 289

design, 292, 293

performance, 289, 290

reported implementations, 301–304

simulations, 290, 291

switched-capacitor N-path, 294–296

transfer function design, 284

cascaded, 193

analytical linearized modeling, 196, 197

circuit topologies, 204–208

continuous-time, 213–217

experimental results, 211, 212, 239–241

high speed, 219

multibit, 222–224, 233–239

simulations, 197, 198

single-loop design, 195

high-order single-bit, 165–190

choices for the NTF, 172

comparison of loop topologies, 174

example: loop filter design, 181–183

preventing idle tones, 185, 186

stability, 170, 183

switched-capacitor loop filter design, 186

Architectures, 199, 203, 309

cascade (MASH), 315

cascade-of-resonators, 293

chain of integrators, 175–180

comparison of, 166

dual-quantizer, 273

dual-truncation MASH, 318

error feedback, 313, 314

improved DAC linearity, 247

layout influences, 399

Leslie-Singh, 273–275

multibit noise-shaping, 245

multistage, 429–431

Autocorrelation, 82, 83

of quantization error, 86, 87, 93

with added dither, 88

Autozeroed integrator, *see* Integrator

B

Bandpass $\Delta\Sigma$ ADCs, 282

band location, 285

Bandpass $\Delta\Sigma$ ADCs (*continued*)
 continuous time design, 297
 design, 292
 example modulator, 288
 linear model, 285
 linear model predictions, 289
 reported implementations, 301
 SNR, 291
 transfer function design, 284–293

Bang-bang control, 44

Barrel shifter, 257

Bennett's theory, 47–51, 63
 conditions, 48, 51, 52

C

Capacitors,
 averaging, 316
 mismatch in, 210, 230, 350

Cascaded modulators, 24, 194–197, 206, 373
 analytical linearized modeling, 196
 circuit topologies, 204–208
 continuous time, 213–217
 designing, 25
 designing ADCs, 193
 experimental results, 211–212
 idle-tone performance, 196
 sources of error, 209
 third order system diagram, 199, 203
 using digital correction, 269

Cascaded multibit modulators, 222–244
 block diagram, 222, 227
 design of, 233
 with digital correction, 269
 with dual-feedback single-path, 276
 with dual quantization, 275
 experimental results, 239–241
 harmonic distortion, 224, 226
 implementation, 229
 interstage coupling, 225
 linear model, 223

Chaos, 65, 131

Characteristic function method, 53

Charge injection, 341, 342, 351, 368

Circuit topologies, 175–181, 302
 internal DAC, 247

Clock generation, 361

Clock jitter, 363, 401, 402

Clock noise, 96

Comb filter, *see* Decimator

Comparator, 360
 with regenerative feedback, 372

Complex modulation, 300

Continuous-time modulators, 158–162, 297
 (non)return-to-zero waveforms, 160, 298

D

DACs,
 architectures, 309
 cascade structure, 315, 316
 design example, 321, 324, 325
 dual-truncation MASH, 318–320
 error feedback structure, 313

Dead zones, 10, 20

Decimation, 410, 411
 analog filters, 390
 bandpass modulators, 300
 brick-wall, 198, 199
 comb filter, 197–199, 391, 392, 416–418
 design of, 29, 34
 efficiency, 413–416
 efficient direct-form M-to-1, 424, 425
 filter, 210
 implementing sinc, 32
 low-pass filter, 35
 multistage architectures, 28, 429–431
 overview, 33, 34
 polyphase architectures, 426–429
 sinc, *see* comb

Delta modulation, 26–28

Delta-sigma modulation, 5
 architecture, *see* Architecture
 basic structures and terminology, 78–80
 C++ code simulating, 453, 454
 continuous-time design, 158–162, 297
 (non)return-to-zero waveforms, 160, 298
 dead zones, 10
 illustration, 21
 dynamic range limits, 151, 155
 example, stereo 18-bit $\Delta\Sigma$ ADC IC, 186
 idle tones, *see* Idle tones
 at low oversampling ratios, 220
 with multibit internal converters, 224
 pattern noise, 8, 10
 practical design methodology, 152–154
 design trade-offs, 156
 example, sixth-order NTF, 154
 with ramp input, 6

second order, 13, 66, 85
 general block diagram, 149
 single-loop, 59, 194
 third order, 21

Demodulator, 36
 cascaded, 40
 circuit design, 41

Describing function analysis, 52, 145

Design examples,
 ADC, 288, 289
 bandpass, 288
 DAC, 321, 324, 325
 decimation, 28
 loop filter, 181
 noise transfer function, 321
 sixth-order NTF, 154
 stereo 18-bit ADC IC, 186

Differential circuits,
 1-bit DAC, 359
 cascaded multibit $\Delta\Sigma$ modulator, 235
 comparator, 237, 360, 361
 folded-cascade op-amp, 236, 344–347
 SC integrator, 337–340
 second-order $\Delta\Sigma$ modulator, 369
 sixth-order bandpass modulator, 295

Digital correction techniques, 264, 316, 317
 calibration of, 265–269

Distortion, 455–458
 intermodulation, 97

Dither, 52, 85
 in A/D modulators, 121
 dynamic, theory, 124
 implementation considerations of, 127
 effect on tones near $f_s/2$, 119
 generation, 121
 nonsubtractive, 104
 subtractive, 105, noise-shaped, 123
 noise-shaped, 76
 empirical studies, 112–120
 in PCM quantizers, 104
 topologies, 107
 circuit, *see* Circuit topologies
 for multistage modulators, 109
 for single-stage modulators, 107

Dithered PCM, *see* PCM

Duality, 411, 412

Dual quantizer ADC, 273–277

Dynamic range, 200, 220, 231
 considerations, 361
 scaling, 362

E

Element matching,
 dynamic matching, 251
 individual level averaging, 259
 noise-shaped, 260
 dynamic randomization, 253
 effect on SNR, 263
 trimming methods, 249

Error cancellation logic, 223, 232

Error feedback, 23, 180

Experimental results,
 bandpass, 301
 cascade, 211, 212, 239–241

F

Filter transformations, 286
 low-pass to band-pass, 287

FIR filters, 409, 413–415
 direct-form structure, 422
 transposed, 423
 half-band, 418
 implementation, 437, 439
 minimum phase, 421
 ternary-encoded, 419

First order modulator, 6
 C++ code simulating, 453
 implementation, 169
 stability, 147–152

Flicker noise, 240, 353–356
 modeled in a MOSFET, 355

Folded-cascode amplifier, 191, 236, 344
 biasing schemes, 344
 common-mode feedback scheme, 347
 input-referred noise, 346

Fractional rate changing, 413

H

Half-band filters, 418

Harmonic distortion, 211, 225, 234, 240, 342, 351

Harmonic power, 225

High order $\Delta\Sigma$ modulators, 14, 15
 C++ code simulating a fifth order, 454
 choices for the NTF, 172
 comparison of architectures, 166

High order $\Delta\Sigma$ modulators (*continued*)
 comparison of loop topologies, 174
 chain of integrators, 176–180
 error feedback, 180, 181
 example loop filter design, 181–183
 comparison of SC and RC realizations, 168
 design choices, 167
 preventing idle tones, 185
 root locus of, 144
 stability, 141

I

Idle tones, 23, 62, 80–83, 100, 135, 384
 in third order modulators, 20, 22
 in multistage modulators, 98
 in single stage modulators, 84
 observability of, 80–81
 practical measures for preventing,
 185
 commercial designs, 185
 IIR filters, 420
 Impact ionization, 190
 Input impedance, 363
 Integrator,
 analog, 204
 autozeroed, 204, 206
 effects of component nonidealities,
 348
 finite op-amp gain in, 210
 input, 336
 leakage in, 13, 20, 69, 205, 232, 233
 nonlinear effects, 350–353
 nonlinear settling, 348, 349
 pole error, 350
 switched-capacitor, 204, 205
 Interfacing to a $\Delta\Sigma$ ADC, 367, 368
 Intermodulation distortion, *see* distortion
 Interpolation, 411
 efficiency, 413–416
 multistage architectures, 429–431
 polyphase architectures, 426–429
 sinc^k , 37, 416, 417
 Interstage coupling, 225
 Intrinsic noise, 353, 397
 Invariant set, 150

J

Johnson noise, 354

L

Layout considerations, 365–367, 398, 399
 Limit cycles, *see* Idle tones
 Loop filter, 78, 142, 175
 continuous time, 298
 design example, 181
 switched-capacitor design, 186

M

Macromodel, OTA, 461
 MASH modulators, 64, 80, 194, 271, 403
 see also Cascaded modulators
 MATLAB, 89, 92, 161, 297, 450
 delta-sigma toolbox, 156
 Multibit converter design, 447
 Multibit internal quantizer, 264
 Multibit quantizer loops, 316
 Multistage conversion, *see* Cascaded
 modulators

N

Noise,
 capacitor and $1/f$, 296
 clock, 96
 Noise transfer function (NTF), 79, 143
 Butterworth alignment, 173
 choices for, 172
 design example, 321
 N-path design, 294
 sixth order example, 154–157
 with multibit quantizer, 447–449
 Non-idealities,
 capacitor mismatch, 210, 230, 350
 charge injection, 341, 342, 351, 368
 finite op-amp gain, 210, 294
 impact ionization, 190
 integrator leakage, 13, 20, 69, 205, 232, 233
 intrinsic noise, 353, 397
 thermal noise, *see* Thermal noise
 Nyquist rate, 78
 Nyquist theorem, 408–410

O

Op-amp, 343
 class AB, 371
 finite gain, 210, 294

folded-cascode, 191, 236, 344
 biasing schemes, 344
 common-mode feedback scheme, 347
 input-referred noise, 346
 speed considerations, 297
 Optimization, 449
 least-squares, 287
 Oversampling, 1
 review of, 1–3

P

Pattern noise, *see* Idle tones
 PCM, 3, 36, 55, 61, 62
 dithered, 58
 multilevel, 26
 Polyphase filter, 426, 429
 PWM, 403, 404

Q

Quantization,
 basic properties, 3, 4
 error, 3–5, 24
 statistical properties, 3
 two-level, 11
 with error feedback, 38
 Quantization errors, 75
 periodic structure of, 76
 Quantization noise, 46, 210, 450
 cancellation, 200, 201
 Quantizer, 329
 correlation of error and input, 50
 error, 45–50, 59
 accumulation of, 448
 gain, 12, 145–148
 μ -law, 272
 overload, 66–68
 resolution, 220
 white noise approximation, 5, 46,
 146
 Quasi-stationary, 54

R

Reference voltage, 284, 386
 Resolution,
 calculation of, 7
 Root locus,
 of a high-order modulator, 144

S

Sampling, 410–411
 Signal transfer function, 79, 143
 Simulation approaches, 458–463
 C++ code for a fifth order modulator, 454
 C++ code for a first order modulator, 453
 comparisons, 460
 HSPICE, 463
 MATLAB toolbox, 156
 OTA macromodel, 461
 SWITCAP2, 459
 SNR, 28, 153–157
 Sripad and Snyder's conditions, 50
 Stability, 141, 170
 describing function method, 145
 linear analysis, 142
 rigorous theoretical results,
 first and second order, 147–152
 with dither test, 130
 with uncontrolled input signal, 170
 Stabilization,
 nonlinear techniques, 162
 nonlinear global techniques, 183
 state variable method, 183–184
 Switched-capacitor circuits, 195, 206–208,
 234–238, 295, 303, 369, 387–389, 396, 400
 1-bit DAC, 359
 3-bit differential DAC, 238
 3-bit flash ADC, 238
 charge injection, 341, 342, 351, 368
 cascaded multibit CMOS $\Delta\Sigma$ modulator,
 235
 clock jitter, 363
 dynamic range scaling, 362
 finite op-amp gain, 294
 folded-cascode operational amplifier with
 common-mode feedback, 237
 fourth-order Bessel low-pass filter, 388,
 390
 gain error, 230
 gain insensitive integrator, 205
 incomplete settling, 232, 234
 integrator, 205, 236, 337–340
 loop filter, 186
 regenerative feedback comparator, 237
 second-order $\Delta\Sigma$ modulator, 369
 single-ended integrator, 205
 sixth-order $\Delta\Sigma$ modulator, 295
 Switched-capacitor design, 292–297

T

Thermal noise, 210, 240, 309, 353–356
Tones, *see* Idle tones
Topology, *see* Architecture

U

Uniform quantization, 45
Unit elements, 247

V

Voltage references, 384–386

W

Weyl’s theorem, 56, 62, 65
White noise approximation, 5, 47, 52, 146

About the Editors

Steven R. Norsworthy is currently worldwide Operations Manager for the Wireless Telecom and Video Business Units of Motorola Semiconductor (Analog Division) in Tempe, Arizona. Prior to that, from 1985–1995, he worked at AT&T Bell Laboratories, where his last position was manager of the advanced mixed-signal technology group. He has published numerous papers and holds 18 patents in the area of delta–sigma converters. He received BSEE and MSEE degrees from the University of South Florida and the University of Florida, respectively. At Pennsylvania State University, he has taught graduate courses in digital signal processing. He also holds two degrees in music and has played for the Florida Orchestra, the New Hampshire Symphony, the Lehigh Valley Chamber Orchestra, and the Pennsylvania Symphonia Orchestra. He was a faculty member in the Department of Music at the University of New Hampshire from 1978–1990.

Richard Schreier received B.A.Sc. and M.A.Sc. degrees in Electrical Engineering from the University of Toronto in 1983 and 1985. From 1986 to 1988 he worked as a member of the scientific staff in the CMOS cell library group of Bell-Northern Research, Ottawa, Canada. He then returned to the University of Toronto to pursue his Ph.D. degree. Upon completion of his Ph.D. in 1991, he joined Oregon State University, where he is currently an associate professor. His interests in the mathematical and computer-aided design aspects of circuit design have been rewardingly indulged by the many opportunities afforded by noise-shaping circuits.

Gabor C. Temes received his undergraduate education in Hungary and his Ph.D. in Electrical Engineering from the University of Ottawa in 1961. He has held industrial positions

with Bell-Northern Research and Ampex Corporation and academic ones at the Technical University of Budapest, Stanford University, and UCLA. He is currently a professor at Oregon State University. He has written about 250 publications and several books. He is a Fellow of the IEEE and has received numerous IEEE awards, including the Centennial Medal and the Technical Achievement Award as well as the Education Award of the Circuits and Systems Society. His research deals with CMOS analog integrated circuits.